Understanding Machine Perception: How Do Neural Networks Represent the World?

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Roland Simon Zimmermann
aus Recklinghausen

Tübingen
2024

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	07.07.2025
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Dr. Wieland Brendel
2. Berichterstatter:	Prof. Dr. Seong Joon Oh
3. Berichterstatter:	Prof. Dr. Wojciech Samek

Abstract

In recent years, neural networks have become omnipresent in computer vision. However, many open questions remain about their functionality and reliability: How do these networks perceive the world? Under what assumptions do they learn to correctly recognize and represent the world? And when do they fail?

With Artificial Intelligence (AI) systems being integrated into ever more everyday applications in the real world, there is a pressing need for a trustworthy foundation upon which to base them. This thesis explores possible routes to higher trustworthiness through an enhanced understanding of neural networks. For this, it pursues both a bottom-up and a top-down approach:

The bottom-up approach focuses on rendering the internal information processing of specific neural networks understandable to humans. As long as black-box networks' (internal) information processing remains obscure, skepticism about their behavior will persist. I first introduce experimental paradigms to precisely quantify how well humans can interpret the features internally used by neural networks. I continue investigating the (un)reliability of existing interpretability tools. Next, I compare the interpretability of various vision models and show a need for explicitly optimizing for it. Finally, I present work on finding fully automated interpretability measures that alleviate the need for manual human evaluations. Such automated measures have the potential to enable interpretability optimization, resulting in more interpretable models.

The top-down approach investigates general machine learning algorithms and how to obtain theoretical guarantees for their learned representations. My strategy is twofold: I begin by working on a better understanding of when and why contrastive learning, a common form of representation learning, works. By connecting contrastive learning with identifiability research, I show that under certain assumptions, contrastive learning inverts the data-generating process. Next, I present work proposing a theoretical framework for analyzing object-centric learning to allow stronger guarantees on the generalization capability in multi-object scenes. This leads to the first method that provably learns structured, object-centric representations. I expect both theoretical contributions to inspire new research on reliable and scalable learning algorithms.

In summary, this thesis extends our understanding of neural representation learning algorithms and illuminates paths to make them more trustworthy. While the theoretical results are expected to inspire research on more theoretically grounded representation learning algorithms, the practical tools shed more light on how neural networks work and enable ways to improve their interpretability and traceability.

Zusammenfassung

In den letzten Jahren sind neuronale Netze im maschinellen Sehen allgegenwärtig geworden. Es bleiben jedoch viele offene Fragen zu ihrer Funktionalität und Zuverlässigkeit: Wie nehmen diese Netze die Welt wahr? Unter welchen Voraussetzungen stimmt ihre Wahrnehmung mit der des Menschen überein? Und wann versagen sie?

Da Künstliche Intelligent in immer mehr alltägliche Anwendungen integriert wird, besteht ein dringender Bedarf an einer vertrauenswürdigen Grundlage, auf die sie sich stützen kann. In dieser Arbeit werden mögliche Wege zum Erreichen von Vertrauenswürdigkeit untersucht, indem sowohl ein Bottom-up- als auch ein Top-down-Ansatz verfolgt wird:

Der Bottom-up-Ansatz konzentriert sich darauf, die interne Informationsverarbeitung bestimmter Netze für den Menschen verständlich zu machen. Solange die (interne) Informationsverarbeitung von neuronalen Black-Box-Netzen ein Rätsel bleibt, wird die Skepsis über ihr Verhalten fortbestehen. Ich führe zunächst experimentelle Paradigmen ein, um genau zu quantifizieren, wie gut Menschen die von neuronalen Netzen intern verwendeten Merkmale interpretieren können. Anschließend untersuche ich die (Un-)Zuverlässigkeit bestehender Interpretationswerkzeuge. Als nächstes vergleiche ich die Interpretierbarkeit verschiedener Netzwerke und zeige die Notwendigkeit einer expliziten Optimierung der Interpretierbarkeit auf. Schließlich stelle ich Arbeiten zur Entwicklung vollautomatischer Interpretierbarkeitsmaße vor, die die bisherige Abhängigkeit von manuellen Bewertungen durch den Menschen überwindet. Solche automatisierten Maße haben das Potenzial, eine Optimierung der Interpretierbarkeit zu ermöglichen, was zu besser interpretierbaren Modellen führt.

Der Top-Down-Ansatz untersucht Algorithmen des maschinellen Lernens und wie man theoretische Garantien für ihre erlernten Darstellungen erhält. Meine Strategie ist zweiteilig: Ich beginne damit besser zu verstehen, wann und warum kontrastives Lernen, eine gängige Form des Repräsentationslernens, funktioniert. Indem ich das kontrastive Lernen mit Forschung zur Identifizierbarkeit verbinde, zeige ich, dass das kontrastive Lernen unter bestimmten Annahmen den Prozess der Datengenerierung invertiert. Des Weiteren stelle ich Arbeit vor, die einen theoretischen Rahmen für die Analyse des objektzentrierten Lernens vorschlägt, um Garantien für die Generalisierungsfähigkeit in Szenen mit mehreren Objekten zu ermöglichen. Dies führt zu der ersten Methode, die nachweislich strukturierte, objektzentrierte Repräsentationen lernt. Ich erwarte, dass beide theoretischen Beiträge neue Forschungen zu zuverlässigen und skalierbaren Lernalgorithmen inspirieren werden.

Zusammengefast erweitert diese Arbeit unser Verständnis von Algorithmen zum Erlernen neuronaler Repräsentationen und zeigt Wege auf, um sie vertrauenswürdiger zu machen. Während die theoretischen Ergebnisse die Forschung zu neuen, theoretisch fundierteren Algorithmen anregen sollen, werfen die praktischen Werkzeuge mehr Licht auf die Funktionsweise neuronaler Netze und ermöglichen Wege zur Verbesserung ihrer Interpretierbar- und Vertrauenswürdigkeit.

Acknowledgments

This thesis concludes a big chapter of my life. A chapter characterized by change, various global crises, but also, most importantly, all the great people in my life.

First of all, I'd like to express my gratitude to our society for enabling me and other young people to get interested in science and finance our research. This gratitude also extends to everyone responsible for establishing research facilities, organizing research conferences or contributing to the general scientific process.

I am deeply thankful for all the researchers I was fortunate to collaborate with in the past few years. You taught me new things, and some even became close friends. First, I'd like to thank Ulrich Parlitz for introducing me to the exciting world of research. As research is a craft of its own, I am very thankful to my PhD supervisor, Wieland Brendel, for teaching me his craft, giving me great advice and being an amazing mentor.

Next, I want to thank my fellow lab members who experienced the ups and downs of research with me and with whom I had many insightful discussions. Most importantly, I want to thank in alphabetical order: Attila Juhos, for the exciting discussions about machine learning theory; David Klindt, for initiating exciting discussions and projects; Evgenia Rusak, for being very inclusive and generally a great collaborator and scientist; Judy Borowski, for improving my communication and organization skills; Robert Geirhos, for being supportive and offer great advice; Prasanna Mayilvahanan, for inspiring me to look for light even in the face of bad news; and Thomas Klein, for all the great discussions we had about interpretability. I want to thank Evgenia, Prasanna, Attila, and Thaddäus for being wonderful friends, the time we spent together, and the memories we made.

I am also thankful for the chances given by Nicholas Carlini and Florian Tramèr, and by Klaus Greff and Thomas Kipf when they hosted me during my internships at Google Brain. I learned much from all of you about the scientific process.

Furthermore, I also want to thank all of my friends from outside the lab for being such great people, for encouraging and supporting me, and for making life more joyful. Most importantly, this applies to Laurenz Hemmen and Julien Siems.

A special thank you goes to my loving family for all their support: Thank you, Mama, for being caring and giving great advice; thank you, Papa, for getting me excited about science; and thank you, Felix, for igniting my passion for computer science. Finally, I want to thank Michelle Pantis for being a truly wonderful partner and for helping me notice and cherish the positive aspects of everything and everyone.

In summary, I am immensely grateful to everyone who contributed to this chapter of my life and helped me become the son, brother, friend, partner, colleague, and researcher I am today.

Contents

1 Introduction **11**
1.1 Understanding the Internal Information Processing of Representation Learners 13
1.1.1 Background . . . 14
1.1.2 Research Questions . . . 16
1.2 Understanding Representation Learning Algorithms Through Theoretical Guarantees . . . 19
1.2.1 Background . . . 20
1.2.2 Research Questions . . . 22
1.3 Publications . . . 25

2 Understanding the Internal Information Processing of Neural Networks **27**
2.1 How Useful are (Feature) Visualizations for Understanding Units? . . . 27
2.1.1 Do Visualizations Enable Humans to Predict Neural Activations? . . . 28
2.1.2 Do Visualizations Support Causal Understanding of Neural Activations? 31
2.2 How Reliable Can Any Feature Visualization Method Be in General? . . . 34
2.3 Do Model and Training Design Choices Influence the Per-Unit Interpretability? . . . 37
2.4 Can We Automate the (Human-Centric) Quantification of the Per-Unit Interpretability? . . . 41

3 Understanding Neural Networks Through Theoretical Guarantees **45**
3.1 How Can the Empirical Success of Contrastive Learning be Explained? . . . 45
3.2 Is there a Theory for Object-Centric Learning Providing Performance Guarantees for Models? . . . 49

4 Conclusion **53**

5 Outlook **57**
5.1 Improving the Interpretability of Individual Units . . . 57
5.2 Transferring the Insights on Interpretability from Vision to Language Models . 58
5.3 Scaling Interpretability from Single Units to Entire Models . . . 58
5.4 Benchmarking Future Progress on Interpretability . . . 59
5.5 Understanding Models: Moving from Interpretability to Behavioral Analysis . 60

Bibliography **63**

Appendix **85**
A.1 Exemplary Natural Images Explain CNN Activations Better than State-of-the-Art Feature Visualization . . . 86

A.2 How Well Do Feature Visualizations Support Causal Understanding of CNN Activations? 128
A.3 Scale Alone Does not Improve Mechanistic Interpretability in Vision Model . . 160
A.4 Don't Trust Your Eyes: On The Unreliability of Feature Visualizations 194
A.5 Measuring Interpretability at Scale Without Humans 232
A.6 Contrastive Learning Inverts the Data Generating Process 270
A.7 Provably Learning Object-Centric Representations 292

1 Introduction

Over the course of hundreds of thousands of years, evolution produced a marvelous feat of engineering: the human visual system. We humans are able to perceive the world around us, easily recognize objects and patterns, reason about them, and generalize to completely new environments in a short time. For a long time, although arguably much shorter than evolution needed, scientists and engineers dreamed of building artificial systems replicating human capabilities to advance automation and ease the lives of humans. In the last eighty years, much has happened in the quest of building artificial intelligence (AI): starting with mechanisms that could represent logical expressions [1], we now have access to complex neural networks that can achieve or even surpass human performance in certain tasks [2–5]. However, the task of replicating human vision is not yet complete, as our understanding of these artificially created systems is still fairly limited: How do they perceive the world? Under which assumptions can they be trained to correctly recognize the world? And when and why do they fail?

While the goal of building intelligent or human-like machines had been mystical for a long time [6], it is now an established scientific discipline called machine learning [6–8]. Generally speaking, machine learning describes the work of letting machines — so-called machine learning models — implement non-trivial behavior by learning from data. Compared to the approach of symbolic artificial intelligence, the model's behavior is not hard-coded by humans but determined by its training data [9]. There are various paradigms of teaching models how to map their input to output, such as supervised, self-supervised, and unsupervised training, as well as reinforcement learning, each having different benefits and drawbacks. Furthermore, various approaches to building models exist. The choice of which learning paradigm and model class to use highly depends on the task and the available data [9, 10]. Nowadays, most approaches for building artificial systems that rival humans in their (visual) perception are powered by deep neural networks (DNNs) [10]. DNNs are a concatenation of multiple parameterized linear and non-linear operations (so-called layers), whose parameters are determined during training. Although they have been originally inspired by biological neural networks [11], they are now a means of engineering with an ever-looser biological inspiration [10]. DNNs especially shine for data-rich tasks. However, due to their many building blocks and parameters, DNNs are often black boxes whose decisions and information processing remain a mystery to humans [12, 13]. A common step when building intelligent systems is to transform high-dimensional complex data, such as images or text, into low-dimensional and arguably more structured representations. Such representations are valuable for computing similarities of data samples, e.g., for data retrieval [e.g., 14], or building specialized models for data-scarce tasks [15]. Finding algorithms that learn such representations, so-called representation learners, has been of interest for a long time, especially since the inception of DNNs [15]. As they are used as backbones in various machine learning models and applications, understanding the representation learners is a crucial aspect of understanding the full downstream application. However, as modern representation learners are based on neural networks and are, thus, mainly still black boxes, fully understanding how they process information remains an open problem.

Gaining a better understanding of how neural networks learn, process information, or, more abstractly speaking, perceive the world is an important research goal. The reasons why this

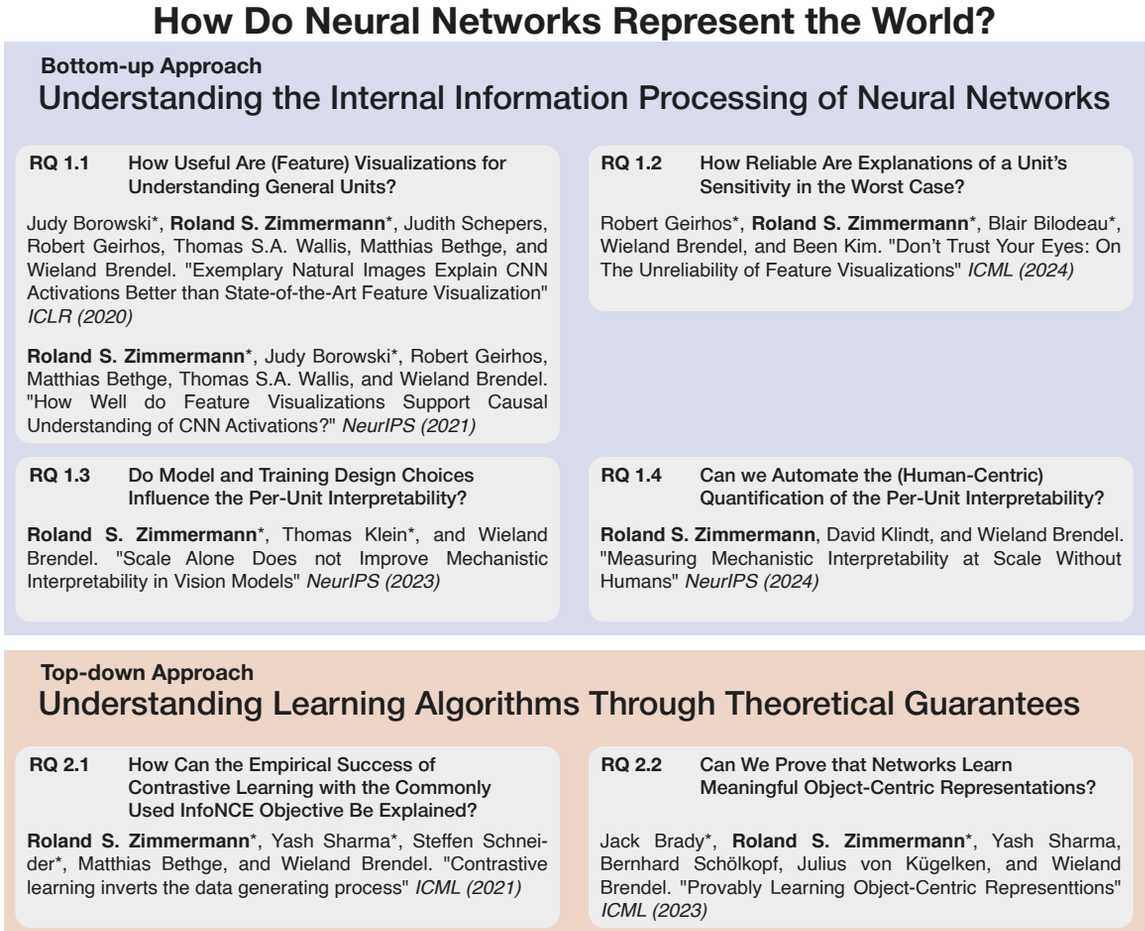


Figure 1.1: **Structure of the Thesis.** This thesis presents two approaches for improving our understanding of how neural networks represent the world: a bottom-up and a top-down approach (see Figure 1.2). The bottom-up approach focuses on understanding *specific* neural networks by deciphering their internal information processing. The top-down approach explores how to understand *families* of neural networks by understanding learning algorithms through theoretical investigations and guarantees.

is important are manifold, amongst which the following two intrigue me the most:
First, by understanding how neural networks process information, one can make their decision-making process more transparent and understandable for humans. With an economically driven interest in deploying neural network-based AI systems in everyday scenarios comes the societal responsibility of ensuring their trustworthiness and reliability. The bigger the consequences of an AI's action are, the more critical it is to verify and ensure their correctness. Making their decisions transparent to humans simplifies said verification process. This observation is also expressed in modern laws such as the EU's "General Data Protection Regulation" [16] and its "Artificial Intelligence Act" [17] demanding accountability and traceability for algorithmic decisions [18]. Moreover, besides looking into deployed networks and explaining their decisions, one might leverage the insights into a network to identify unintended biases or potential failure cases before deployment.
Second, by better understanding the influence of learning algorithms on the resulting neural networks, one can hope to increase their general reliability. Specifically, comprehending why certain learning algorithms work especially well while others fail, and what hidden properties they engrave into a network is helpful for describing their limits. This knowledge is important

for determining the suitability of a learning algorithm for a specific task. Additionally, it can lead to more efficient learning algorithms that produce more reliable models.

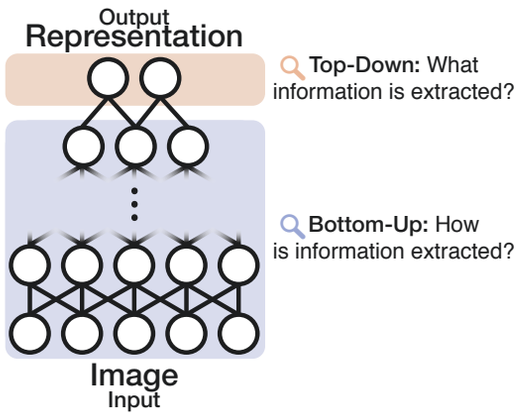


Figure 1.2: **Approaches for Understanding Neural Networks.** This thesis uses two approaches for understanding neural networks. The bottom-up approach aims to understand them by comprehending *how* they extract information (i.e., understanding all of their building blocks individually). Contrarily, the top-down approach aims to understand them by comprehending *what* information they extract by investigating their learned (output) representations.

In this thesis, I present two lines of work toward a better understanding of how neural vision models work and how they process input (see Figure 1.1). Common nomenclature in deep learning says that information is processed bottom-up, similar to how trees grow. Using this nomenclature, this thesis' lines of work can be seen as a bottom-up and top-down approach (see Figure 1.2).

In the thesis' first part, I focus on understanding models mechanistically via a bottom-up approach by investigating their internal information processing (see Section 1.1 and Chapter 2). Specifically, as neural networks are comprised of different units (e.g., layers or neurons), such an investigation aims to understand the input sensitivity of individual units and how they contribute to the behavior of other units. Starting with simple units of the network and covering bigger and bigger parts of the network, one aims to understand networks in a bottom-up fashion. This research is motivated by the hope that once one understands (almost) all units in a network, the network's entire behavior is demystified and can be verified by humans — which will result in higher trust in these networks.

Besides this bottom-up approach to understanding a model's overall behavior by comprehending its building blocks, this thesis presents a top-down approach (see Section 1.2 and Chapter 3). Here, the goal is no longer to understand a neural network by understanding its individual units, but by understanding its overall behavior in the form of its final representations. Specifically, I investigate what information different neural networks extract when using specific representation learning algorithms. This investigation is implemented through theoretical work posing assumptions on the training data of networks to guarantee informative and generally useful representations. Such results can eventually increase the overall trust put in models resulting from said learning algorithms.

1.1 Understanding the Internal Information Processing of Representation Learners

Understanding our mind has been a long-lasting desire of humans [19, 20]. Scientifically, this desire has been acted out first by ancient philosophers [21] who reasoned about the mind on a philosophical or meta-physical level. Psychologists, as well as cognitive and behavioral scientists, analyzed human behavior and reasoned about the mind through their observations [20]. After establishing a tight link between a person's mind and their brain through accidental observations, scientists began to zoom in and analyze the mind on a neurological level [19]. While psychology and cognitive science still aim to understand the

human mind and its behavior through macro observations, neuroscience takes a different perspective: Starting with microscopical investigations of individual neurons, one aims first to understand individual neurons, then bigger and bigger neural circuits, before explaining more and more parts of the brain and mind [22, 23].

With the advent of machines implementing intelligent and partially human-like behavior, this ancient desire extended from understanding human minds to machine behavior. The increasingly more complex behavior of machines sparked the interest of scientists originally interested in understanding the human mind as well as machine learning scientists [24]. The former group's interest often originates in the idea that understanding machine behavior will eventually help explain human minds: For one, machines allow faster and more controlled experiments, whose insights might transfer to biological systems. For example, machine learning models are nowadays a popular tool for modeling the behavior of (parts of) biological brains in neuro- or visionscience [25–28]. For another, methodologies and tools developed for understanding machines might also be helpful for understanding human brains [29]. The latter group's motivation is mostly three-fold: Firstly, the curiosity about how simple mathematical principles lead to complex behavior [30]; second, the desire to obtain insights into the inner workings of machines to improve them [30, 31]; third, the hope that explaining the behavior of machines will make them more reliable and trustworthy [32].

1.1.1 Background

Research on explainable AI (XAI) aims to grant humans insights into machines' decisions and inner workings on various levels of detail [33, 34]. Due to this thesis' focus on vision, this section's upcoming introduction to XAI will primarily focus on said modality. While the terminology and boundaries of XAI are partially blurry [33, 35], it can be seen as an overarching term for two sub-fields:

First, there is interest in making AI systems' decisions understandable for human users [13, 33]. This is particularly interesting for practical applications with ethical or safety concerns [12, 32] in high-stakes scenarios such as medical diagnoses. One potential approach to achieving this is through self-explaining systems or post-hoc explanations [34]. The former refers to special systems that can perform introspection and explain how the final decision comes together in a human-understandable format. With the development of large language models (LLMs) [36] and vision language models (VLMs) [37] that are powerful problem-solving machines and allow users to communicate with them via natural language, hope came that those systems automatically would become self-explaining [38]. However, researchers realized that the introspection produced by such an LLM might be deceiving and not truthfully reflect its decision process [39]. The latter, post-hoc explanations, are model-agnostic explanations that explain a model's prediction even if the model itself offers no explanation [40–42]. A popular family of explanation methods is saliency maps [43–46] that highlight the most relevant regions of an image for the AI's output.

The second sub-field has a different goal. While explanations of a model's predictions are valuable for practical applications, this thesis focuses on a different sub-field of XAI that aims to produce more fundamental insights: (mechanistic) interpretability. Here, the goal is not to understand an AI's decisions and behavior but instead its inner workings [47] — one is interested in mechanistically understanding an AI system. Different motivations exist, such as scientific curiosity and understanding deep learning better [48–50], or improving the safety of AI systems [31, 51, 52]. Among others, mechanistic interpretability includes understanding the role and behavior of individual parts of the system or understanding

the interactions between different parts [48, 53–56]. For neural networks, one might be interested in understanding which input feature a computational unit is responsive to or how units interact and form more complex structures [57, 58]. Here, a computational unit refers to an atomic building block of a neural network, such as an individual neuron for a multi-layer perceptron (MLP) [11], an entire (i.e., spatially averaged) channel or a single pixel of a channel in a convolutional neural network (CNN) [59, 60] or arbitrary activation vectors [61]. Different definitions of units can lead to different insights and conclusions; in this thesis, I follow previous work [48] and consider the spatial mean of channels of CNNs or individual neurons in MLPs.

Various tools have been developed to examine a neural network and uncover its inner mechanisms. These tools explain which input features drive a network's unit, and can be divided into two groups, which differ in how they convey information to users: visual [48, 52, 62–70] or textual explanations [71, 72]. While textual explanations are more accessible to humans, they also are prone to losing important information by being imprecise about the truly relevant input features — after all, an image is worth a thousand words. Importantly, most textual explanation methods are based on visual ones [71, 72] — thus, by advancing the state of visual explanation methods, one can hope to advance textual ones, too. Visual explanation methods aim to explain what input features are relevant for a unit by displaying potential input samples containing said feature. To meet this end, they all implement some type of optimization procedure: Either they perform a brute-force search over a sufficiently large corpus of valid input samples to find dataset examples representing a unit's features [48], or they synthesize images by using gradient-based optimization directly on the model's input [48, 68]. Throughout this thesis, the former type of methods will be referred to as dataset examples or exemplars, while the latter will be called feature visualizations, following Olah et al. [48].

Visual explanations such as feature visualizations have been used in various analyses, aiming to decipher the inner workings of neural networks. The majority [49, 57, 58, 73–78] focused so far on interpreting a single network called GoogLeNet [79], while few also investigated other networks [52, 80–82]. While analyzing a network using such visualizations is cumbersome, the community has made progress in understanding some parts of the GoogLeNet model: Starting with identifying the first units that correspond to meaningful concepts for humans, Olah et al. [48] and Yosinski et al. [83] found feature detectors similar to biological neurons employed in early stages of the human visual system. Next, those circuits that combine these low-level features into more complex ones were identified [49, 58]. Additionally, new low-level feature detectors previously not known to be present in biological vision systems were found [75], for which evidence has later been uncovered independently in biological systems, too [84]. Finally, certain circuits responsible for detecting high-level features such as the orientation of faces were identified [58, 82]. It is thus fair to say that (feature) visualizations have been useful for understanding some parts of some neural networks. However, there have also been critical voices [e.g., 26] questioning the applicability of such tools for understanding general units in networks due to the evolved engineering required for computing them [67].

The choice of what constitutes a unit in a network crucially influences how interpretable such a unit can be for humans. The most accessible starting point of an investigation into a model is to try to understand individual neurons in MLPs or channels in CNNs [48, 85]. This assumes that the units leverage human-understandable concepts and are axis-aligned with single features [48, 86]. However, empirically, this is only sometimes the case [49]. Units responding to single concepts, called monosemantic units, are often found to be easily interpretable, while those corresponding to a mixture of various unrelated

features, called polysemantic units, are harder to understand [48, 49, 87]. Therefore, it is desirable to decrease the degree of polysemanticity in a network [88, 89]. Recently, a potential explanation for such polysemantic units has been presented by the *superposition* hypothesis [86]. It argues that neurons must become polysemantic if a network's layer does not contain enough units to represent all necessary features independently as disentangled dimensions. As a solution, the network learns to represent multiple unrelated features that rarely co-occur and, thus, do not interfere in an entangled state [86]. To decrease the degree of polysemanticity, researchers developed post-hoc tools that are meant to transform polysemantic units into monosomenatic ones by finding a new, monosemantic basis to explain the activations of a layer [50, 90–92]. Among various approaches for finding such a change of basis, the sparse auto-encoders (SAE) have received most attention recently [31, 50, 90, 93].

For a long time, XAI research has been driven by intuition and qualitative evaluations [94]. A reason for the lack of quantitative evaluations is the difficulty of deciding which properties to measure [95] and the challenges of rigorously defining the goal of a method using falsifiable hypotheses [12]. This led to slow progress and even partially wrong claims and conclusions: For example, it was only after years of research on saliency maps that issues in common (qualitative) evaluation paradigms were found, invalidating various approaches [96]. To prevent the repetition of such a crisis, Leavitt and Morcos [97] called for the use of falsifiable statements in XAI research. Furthermore, various high-level desiderata of explanations have been proposed [95], most importantly the fidelity and comprehensibility of explanations [98]. Here, the former means how well an explanation corresponds to actual behavior meant to be explained, while the latter refers to how comprehensible the explanation is for humans.

1.1.2 Research Questions

In the previous section, I summarized the state of interpretability and introduced explanation approaches and their success stories. Now, I will motivate my research agenda through open problems. Specifically, I will formulate four research questions (RQ) that have guided my research on rendering the internal information processing of representation learners more understandable. For each question, I will point out relevant gaps in the community's previous knowledge and briefly summarize my contributions, before presenting them in more detail in Chapter 2.

The first research question (RQ) derives from a fundamental issue of deep neural networks, which might be linked to feature visualizations: adversarial examples. While neural networks can learn to implement complicated behavior, they are surprisingly brittle. The most striking demonstration of their brittleness was given by Biggio et al. [99] and Szegedy et al. [100] in 2013 in the form of *adversarial perturbations*. Adversarial perturbations are small, for humans imperceptible, modifications of an image that can change the output of a neural network in arbitrary ways. By applying such a perturbation to an input sample of the network, one obtains an adversarial example for which the network will produce arbitrary output. Let us consider the case of an image classifier $f : \mathcal{I} \to \{1, \dots, C\}$, recognizing C classes, where $\mathcal{I}$ denotes the space of valid input images. Here, an adversarial example $\hat{\mathbf{x}}$ is indistinguishable from a normal sample $\mathbf{x} \in \mathcal{I}$ for humans but will fool the classifier into misclassifying the image. Formally, adversarial examples (for classifiers) of a clean sample $\mathbf{x}$ can be defined by the following two constraints: (1) $d(\mathbf{x}, \hat{\mathbf{x}}) < \epsilon$, where d measures the perceptual similarity of two samples and ϵ is sufficiently small to guarantee an imperceptible perturbation, and (2) $f(\mathbf{x}) \neq f(\hat{\mathbf{x}})$. If the perturbation is constructed to misclassify an image, this is called an *untargeted* perturbation, and if it is created to make the classifier output a specific prediction, it is called *targeted* [101]. Interestingly, a common technique for

finding adversarial perturbations is to leverage gradient descent or ascent on the input [101]. Feature visualizations are generated by the same principle but with another goal in mind, prompting the question of whether they explain a unit's behavior on normal data or only visualize adversarial perturbations. Considering the problem of adversarial susceptibility, it is unclear how accurately these visualizations reflect a unit's behavior on naturally occurring data samples. Furthermore, to stabilize the optimization procedure and generate visually more coherent visualizations, various intricate engineering tricks such as non-trivial image parametrizations and augmentations have been introduced [48, 66, 70]. These technical choices might induce biases leading to visualizations that also do not reflect a unit's behavior but rather those biases [26].

While (feature) visualizations have been used to interpret neural networks, their usefulness or fidelity has not been quantified yet. I now want to recall the general demand for such quantitative evaluations of XAI methods stated in Section 1.1.1: How much do (feature) visualizations help humans understand the behavior and sensitivity of units? And are they helpful for general or only very specific units? Combining the skepticism regarding the generality of feature visualizations with the potential issue of feature visualizations only showing adversarial perturbation yields the first research question of this thesis:

***RQ 1.1**: How useful are (feature) visualizations for understanding general units?*

Investigated in:

(2.1.1) Judy Borowski*, **Roland S. Zimmermann***, Judith Schepers, Robert Geirhos, Thomas SA. Wallis, Matthias Bethge, and Wieland Brendel. "Exemplary Natural Images Explain CNN Activations Better than State-of-the-Art Feature Visualization." *ICLR (2020)*

(2.1.2) **Roland S. Zimmermann***, Judy Borowski*, Robert Geirhos, Matthias Bethge, Thomas SA. Wallis, and Wieland Brendel. "How Well do Feature Visualizations Support Causal Understanding of CNN Activations?." *NeurIPS (2021)*

To answer this question, together with my collaborators, I introduce two methodologies of quantifying the usefulness of (feature) visualizations for humans. These methodologies follow the call of Doshi-Velez and Kim [12] and leverage psychophysical experiments to quantify the usefulness by testing how well humans can predict the behavior of network units based on their explanations. Experimentally, we measure this for units of GoogLeNet, a CNN to which feature visualizations have often been applied [e.g., 48, 49, 102]. While we find that feature visualizations do provide some helpful information about the behavior of CNN units, they do not explain all of their behavior. Moreover, when we put their performance into context by comparing them against the simpler method of using dataset exemplars, the picture changes: This simple method performs on par or even outperforms the technically more evolved feature visualizations.

Considering feature visualizations' good but imperfect performance in the previously described human evaluations, a natural follow-up question is whether one can build a better visualization method. However, an even more important question is understanding where the performance gap comes from: Does it stem from how the explanation images are generated, meaning that we can hope to find a better explanation method eventually? Or is this an inherent limitation of interpretability methods based on a few explanatory visualizations? In RQ 1.2, I analyze the limitations of visualization methods and challenge the paradigm of using images yielding strong activation to explain a unit's behavior for arbitrary input:

***RQ 1.2**: How reliable are explanations of a unit's sensitivity in the worst case?*

Investigated in:

(2.2) Robert Geirhos*, **Roland S. Zimmermann***, Blair Bilodeau*, Wieland Brendel, and Been Kim. "Don't Trust Your Eyes: On The Unreliability of Feature Visualizations." *ICML (2024)*

Specifically, my co-authors and I show that such explanation methods are only reliable when posing strong assumptions on the network. We highlight the intuitive problem that knowing the position of the minimum and maximum of a function does not allow one to uniquely infer the function unless it is linear. Besides a theoretical investigation of the reliability of visualization techniques, we empirically demonstrate that (feature) visualizations can, in fact, be disconnected from a unit's behavior and be misleading. Despite the low theoretical reliability, we did find a non-zero performance in the human evaluations conducted for the earlier RQs. This suggests that, in practice, neural networks have favorable properties that enable humans to understand them via (feature) visualizations. We elucidate potential reasons for this difference and outline future research directions.

To better understand the inner workings of a neural network, two things can be improved: the tools that generate explanations and the network itself. While developing better interpretability and visualization tools has been an active area of research for the past years [48, 65, 66, 68], the success has been fairly limited as the resulting visualization techniques are still relatively similar. However, at the same time, enormous progress has been achieved in building better-performing or more efficient models [103–105]. While the improved interpretability techniques have not drastically advanced our understanding, it is conceivable that we *accidentally* built more interpretable neural networks by striving for better-performing ones. Previous work found higher alignment between the overall behavior of humans and machines with increasing model/dataset size and classification performance [103]. As models with more human-like behavior might leverage more human-like decision strategies that are, hence, more easily understandable for humans, those models are potentially more interpretable. Therefore, in RQ 1.3, I shift my focus from the visualization methods employed to the networks analyzed and ask:

***RQ 1.3**: Do Model and Training Design Choices Influence the per-unit Interpretability?*

Investigated in:

(2.3) **Roland S. Zimmermann***, Thomas Klein*, and Wieland Brendel. "Scale Alone Does not Improve Mechanistic Interpretability in Vision Models." *NeurIPS (2023)*

Here, in collaboration with other researchers, I test the influence of various design choices, such as the size of the training dataset, training objective or model architecture, and size on the human-perceived per-unit interpretability. For quantifying the interpretability, we leverage the human evaluations introduced in Section 2.1 to answer RQ 1.1. Due to the high time and financial cost of performing our human evaluation, we are limited in how many models we can evaluate. However, by efficiently choosing nine models, we can collect enough data to investigate the influence of the above design choices. A comparison of these nine models shows no signs of an accidental improvement in interpretability, as speculated above. Specifically, we find that for neither feature visualizations nor dataset exemplars, a

clear trend is visible: In contrast to the initial hypothesis, we find no evidence that larger models or models with higher classification performance are more interpretable.

While the methods introduced for RQ 1.1 enable one to measure how interpretable a neural network's inner workings are, these methods are costly and do not scale. This severely limits their application, as only testing a few carefully chosen hypotheses is feasible. Consequently, this limits the pace of development of the community's search for more interpretable neural networks. Conversely, it is said that whatever one can measure at scale, one can also optimize for. Therefore, automating and, thus, scaling up interpretability evaluations, is important as it will open up two powerful research directions. In the field of natural language processing (NLP), efforts to automate interpretability evaluations for LLMs existed before [e.g., 106]; however, doubts about their reliability were shared [107]. While investigating RQ 1.3, we conducted large-scale psychophysical experiments with several thousand participants and several hundred thousand responses that resulted in a large data set of human interpretability annotations. In the final interpretability RQ, I investigate whether machines can be used to approximate human interpretability annotations for vision models:

***RQ 1.4**: Can we automate the (human-centric) quantification of the per-unit interpretability?*

Investigated in:

(2.4) **Roland S. Zimmermann**, David Klindt, and Wieland Brendel. "Measuring Interpretability at Scale Without Humans." *NeurIPS (2024)*

Here, together with my collaborators, I leverage the latest advances in modeling perceptual image similarities [108–110] to build a simple but surprisingly powerful model for solving the psychophysical task designed to quantify the human-perceived interpretability introduced for RQ 1.1. After a simple post-processing of the output of this model, it can eventually be used to estimate how well humans can interpret a unit in a network. We find that our machine metric called *Machine Interpretability Score*, is well aligned with human interpretability scores by performing correlational and interventional evaluations. This new metric enables us now to scale up interpretability evaluations from a few units in a few networks to all units of many networks. This new data allowed us to revisit the analysis conducted for RQ 1.3. We now find an anti-correlation between a model's downstream classification performance and its interpretability.

1.2 Understanding Representation Learning Algorithms Through Theoretical Guarantees

While the first line of work presented in this thesis aims to increase the trust put in neural networks by developing a better understanding of the inner workings of an individual network, the second line has the same motivation but zooms out: It focuses on the overall behavior of a family of networks and the quality of their learned representations. Specifically, it analyzes representation learning algorithms instead of individual instantiations of the learned representations in the form of neural networks. Although such an investigation could be performed through empirical experiments, this thesis takes a theoretical approach: It will introduce theoretical explanations for why empirically successful representation learning algorithms work and what implicit assumptions about the training data they make. Furthermore, it will demonstrate how to derive new theoretically grounded learning algorithms. Compared

to the aforementioned line of work (see Section 1.1), the goal shifts from understanding the inner workings of a specific model to understanding the general behavior of a family of models.

The gap in theoretical knowledge and practical capabilities has become increasingly wide in recent years. Although the fields of artificial intelligence and machine learning were initiated by researchers with diverse backgrounds, such as logic [111], neurophysiology [111, 112], mathematics [113], or computer science [7], most early works share a common feature: They contain proofs and theoretical guarantees [e.g., 111]. While this was true initially, over time, machine learning developed two branches: one taking a theoretical approach focused on proving theoretical explanations and guarantees and one taking an empirical approach focused on engineering better algorithms. Although the knowledge in both branches has advanced, it has progressed at different speeds. Eventually, researchers were substantially faster at proposing novel learning algorithms or model architectures and demonstrating their practical efficacy than proving why and how they work. Nevertheless, understanding how learning algorithms work on a theoretical level can be crucial for understanding their limitations and, consequently, making a well-informed choice when training new models [114]. Similarly, theoretical insights might produce performance guarantees, which can be a step towards trustworthy machine learning models [115]. Therefore, while posing a potentially difficult task, closing the gap between empirical and theoretical knowledge of modern architectures and algorithms is a promising research direction.

1.2.1 Background

The dependence on costly human annotations for training models has limited the widespread applicability of deep learning and neural networks. Therefore, overcoming this limitation by reducing the need for labeled data has been of interest almost since the inception of DNNs [116–118]. With great motivation, a multitude of approaches for increasing data efficiency has been proposed. Most important have been the concepts of representation and transfer learning: They are based on the realization that (intermediate) features of DNNs trained for some task on a (somewhat general) dataset remain meaningful features for a new task (on potentially new data) [10, 15, 119, 120]. While representation learning focuses on learning powerful and general features, transfer learning focuses on the practical challenge of using such features for other tasks or datasets. To further reduce the need for scarce labeled data, the research community has looked into finding more data-efficient ways of learning such powerful features [6, 10, 15]. Quickly, researchers tried to completely remove the need for any human labels in *unsupervised learning* algorithms [10, 121]. A prominent subfamily of this learning approach is *self-supervised learning*: Here, one constructs meaningful training objectives that do not require any human labels but instead use synthetic ones or leverage natural symmetries and invariances [122]. A large number of self-supervised algorithms are based on the idea of *contrastive learning* (CL) [123]. Their underlying idea is that a good representation learner should yield similar features for visually or semantically similar data samples but discriminate dissimilar ones [e.g., 124]. Similar samples are also called positive pairs, while dissimilar ones are called negative data pairs. The positive pairs are mainly generated by some form of data augmentation that maintains the sample's relevant content such that both samples of a pair share meaningful content: Following the thesis' overall focus on the vision domain, popular choices are temporal (for videos) or spatial (for videos or images) crops [125–127], spatial transformations and visual/stylistic changes [125, 128, 129]. While multiple loss functions have been proposed to attract the representations of positive pairs while repelling those of negative pairs, one loss function has

been particularly seminal - the InfoNCE objective:

$$\mathcal{L}_{\text{InfoNCE}} = -\mathbb{E}_{\mathbf{x}} \left[\log \frac{e^{f(\mathbf{x})^\top f(\widetilde{\mathbf{x}})/\tau}}{e^{f(\mathbf{x})^\top f(\widetilde{\mathbf{x}})/\tau} + \sum_j^M e^{f(\mathbf{x})^\top f(\mathbf{x}_j^-)/\tau}} \right]. \tag{1.1}$$

Here, $\mathbf{x}$, $(\mathbf{x}, \widetilde{\mathbf{x}})$ and $\{ (\mathbf{x}, \mathbf{x}_i^-) \}_{i=1,\dots,M}$ denote a sample (also called anchor point), its positive pair and its M negative pairs, respectively. Originally proposed by Oord et al. [124], it has inspired various other learning algorithms [e.g., 125, 130–133]. While some theories attempt to explain why this objective affords meaningful features, they are incomplete, and gaps remain, as will be discussed in the next section.

A more specialized type of unsupervised representation learning is unsupervised *object-centric* learning [134, 135]. It is motivated by the rationale that natural scenes consist of multiple objects, and human perception has evolved to reflect this structure by grouping information from the same object together. Representing scenes in a structured way has been argued to be a critical component for the powerful planning and generalization capabilities of humans [134]. On the contrary, the unsupervised representation learning algorithms introduced in the previous paragraph do not use this structure but instead learn a single representation for an entire scene. This suggests a potential limitation regarding their generalization capabilities: It is conceivable that small changes in the scene, such as the introduction of novel and previously unseen objects, diminish the quality of the representation of the rest of the scene. To ensure that such small changes do not result in samples that are effectively out of the training distribution, it can be helpful to divide a scene into separate objects and represent each individually. This approach is called object-centric learning (OCL) [134, 136, 137]. By doing this, one can hope to obtain representations that maintain their quality for most of the scene, even in the presence of previously unseen objects [134, 138–140]. Thus, OCL is conceivably a critical step for building machine learning models that are robust to out-of-distribution samples and generalize well to new scenes [134, 135]. Moreover, one can argue that object-centric representations enable more data-efficient fine-tuning of models to downstream applications, further reducing the need for costly human labels [139, 140]. However, this argument is still actively debated, waiting for conclusive results [141]. A wide variety of OCL methods has been proposed for vision data over the past years, both for images [e.g., 139, 142–148] as well as for videos [149–153]. A common feature across many of them is that objects in a scene are represented by individual *slots*. While present before, this idea gained popularity through the seminal paper by Locatello et al. [139]. Contrary to the large body of empirical work, there is little theoretical work on object-centric learning. This is mainly caused by the lack of a commonly agreed-upon theoretical framework for investigating OCL. More specifically, while sounding intuitive initially, the notion of what constitutes an object is blurry and highly debated, making a mathematical formalization difficult. Although different definitions of what constitutes an object have been proposed before, which are based on ideas in psychology and cognitive science [154–159], such as the Gestalt principles [158] or Spelke objects [159], they have not been formalized yet to be used in theoretical machine learning.

Besides object-centric learning, a more general approach to learning well-scaling and generalizing representations is given by *disentangled* representation learning [15]. This paradigm is motivated by the observation that our physical world can be described by latent factors of variation that correspond to individual physical properties such as the position, rotation, velocity, color, or lighting of objects. Finding representations that disentangle a mixture of these factors into separate dimensions — ideally, each corresponding to one of the physical properties describing the training data — is argued to simplify reasoning tasks, increase robustness to distribution shifts, and, thereby, enable more powerful downstream

applications [15]. However, empirical results for this hypothesis are mixed and not yet conclusive [160, 161]. Moreover, disentangled representations are expected to be more interpretable to humans [15, 34]. Various empirically-motivated algorithms have been proposed to find such representations [e.g., 162–166]. Further work has demonstrated the applicability of such approaches to domains other than computer vision, e.g., neuroscience [162]. A concept related to disentanglement is that of *identifiability*. In short, a model is called identifiable if one can infer its parameters arbitrarily well based on observations from that model in the limit of infinite observations [167]. As proving that a model is identifiable by some inference model implies that the inference model learns disentangled representation, identifiability has proved an essential part of theoretical analysis of disentanglement work [168–170]. Another closely related field is that of *Independent Component Analysis* (ICA), which aims to recover the underlying sources of high-dimensional data using the assumption that the data's underlying sources are statistically independent [171, 172]. Previous work has shown that some existing deep learning techniques can be shown to be identifiable and solve an ICA problem [168, 169, 173]. Moreover, new learning models and algorithms have been introduced to be provably identifiable [e.g., 174–177].

1.2.2 Research Questions

Contrastive learning has been applied with great success in various domains and applications. Despite its empirical success, the research community still has not fully understood how and why this family of learning algorithms works so well [121]. However, understanding it can be crucial for two reasons: Firstly, it allows one to determine their scope and limitations, ensuring that one uses a reasonable algorithm for novel tasks. Secondly, one can mitigate the limitations and, thus, develop either generally better or more specialized learning algorithms suited for specific tasks. While various theoretical explanations have been proposed, they are incomplete: For example, explanations based on the InfoMax principle [178] that reason about the mutual information of different views [124, 125, 179–181] have been partially contradicted by empirical observations [182]. Another theory introduces latent classes to explain the behavior of CL [183]. However, some of this theory's predictions, namely that an excessive number of negative samples harms performance, faced mixed empirical observations [125, 128, 184–187]. Per the earlier description, the family of InfoNCE objectives has been a popular choice for performing CL [124, 125, 131, 187]. Therefore, my thesis focuses on this family of objectives and investigates this representatively for other contrastive approaches. Motivated by the aforementioned lack of a theory explaining the empirical success of said objective, I study the following RQ:

> ***RQ 2.1****: How can the empirical success of Contrastive Learning with the commonly used InfoNCE objective be explained?*
>
> *Investigated in:*
>
> (3.1) **Roland S. Zimmermann***, Yash Sharma*, Steffen Schneider*, Matthias Bethge, and Wieland Brendel. "Contrastive learning inverts the data generating process." *ICML (2021)*

Here, my collaborators and I connect the fields of identifiability, in the form of nonlinear ICA, and contrastive learning. By recognizing that CL with objectives from the InfoNCE family implicitly solves an ICA problem, we find a theory explaining the empirical success of CL. We introduce a latent variable model mapping low-dimensional latent factors to high-

dimensional observations. We then show that minimizing the InfoNCE objective on positive and negative pairs of observations leads to an inversion of the generative process: If the pairs of observations follow a certain family of distributions, an encoder minimizing InfoNCE will recover the ground-truth factors of variation up to simple linear transformations. The ground-truth factors of variation can be seen as a minimal lossless representation of the data. Therefore, if an encoder recovers them, it is plausible that its features are powerful for various downstream tasks. Thus, our theory explains CL's empirical success and the usefulness of its learned representations. Moreover, our theory suggests that CL with losses from the InfoNCE family includes certain implicit assumptions on the training data distribution. We demonstrate how new variants of InfoNCE, which are provably identifiable, can be derived for novel assumptions. This might lead to more specialized objectives and learning algorithms better suited for special data sources.

Unsupervised object-centric representation learning is another family of representation learning algorithms mainly driven empirically and lacking a sound theoretical framework. While there have been great algorithmic advances scaling OCL from simple toy data [145] to more realistic settings and datasets [153, 188–190], the theoretical understanding of these algorithms lacks behind: There exist little to no performance guarantees for such algorithms. Predominantly, this issue is caused by the difficulty of formalizing the task of learning object-centric representations without supervision: As per Section 1.2.1, no common agreement exists even on formal definitions of what constitutes an object. This renders the theoretical analysis of OCL algorithms cumbersome and prevents the derivation of guarantees on the quality of learned representations. However, I argue that the derivation of a mathematical theory for OCL can be valuable for steering the future development of algorithms. Following the previous RQ's motivation, it is conceivable that such a theory uncovers the limitations of existing approaches and makes practitioners more confident in choosing the best learning algorithm for a specific data source and task. This knowledge can increase the reliability and, thus, the trust put into object-centric learners. Furthermore, such theoretical work can produce hypotheses for novel and better-performing algorithms or algorithms better suited for specific tasks. Motivated by the potential impact a theoretical analysis of OCL has, this thesis investigates the following and last RQ:

***RQ 2.2**: Can we prove that networks learn meaningful object-centric representations?*

Investigated in:

(3.1) Jack Brady*, **Roland S. Zimmermann***, Yash Sharma, Bernhard Schölkopf, Julius von Kügelken, and Wieland Brendel. "Provably Learning Object-Centric Representations." *ICML (2023)*

In collaboration with my colleagues, I formalize the task of learning object-centric representations without supervision by introducing a latent variable model generating the observations an inference model will be trained on. We implicitly define what constitutes an object by assuming two properties on the structure of this latent variable model that we call *irreducibility* and *compositionality*. Put simply, these two properties imply that no pixel in the observations belongs to more than one object, and pixels belonging to the same object need to share unique information that is not shared across different objects. By imposing these assumptions on a learned encoder and its inverse, we derive an identifiability proof for the learned representations. Importantly, we propose a learning algorithm, leveraging a novel regularization loss applied to an auto-encoding task, that provably recovers the ground-truth separation of objects and their latent information. Empirically, we verify our theory by

demonstrating the correctness of our proposed learning algorithm and analyzing existing learning algorithms; we find behavior corroborating our theory. This study provides the first theoretical performance guarantee (through an identifiability proof) for unsupervised object-centric learning, which can inspire further research into theoretically-grounded learning algorithms.

1.3 Publications

This thesis presents results that were previously published as individual research papers. The full papers are shown in the Appendix. The papers' motivation, key findings, and discussion are summarized in Chapter 2 and Chapter 3. As all of these studies were conducted in collaboration with multiple researchers, an overview of the individual author contributions is given there. An asterisk (*) in the author list indicates authors with equal (technical) contributions, while a dagger (†) denotes joined supervisors. In addition, more papers were published pursuing the doctorate degree that are not explicitly included in this thesis.

Conference Publications Included in This Thesis

Poster ICLR	Judy Borowski*, **Roland S. Zimmermann***, Judith Schepers, Robert Geirhos, Thomas SA. Wallis, Matthias Bethge, and Wieland Brendel. "Exemplary Natural Images Explain CNN Activations Better than State-of-the-Art Feature Visualization." *ICLR (2020)*
Spotlight NeurIPS	**Roland S. Zimmermann***, Judy Borowski*, Robert Geirhos, Matthias Bethge, Thomas SA. Wallis, and Wieland Brendel. "How Well do Feature Visualizations Support Causal Understanding of CNN Activations?" *NeurIPS (2021)*
Spotlight NeurIPS	**Roland S. Zimmermann***, Thomas Klein*, and Wieland Brendel. "Scale Alone Does not Improve Mechanistic Interpretability in Vision Models." *NeurIPS (2023)*
Poster ICML	Robert Geirhos*, **Roland S. Zimmermann***, Blair Bilodeau*, Wieland Brendel, and Been Kim. "Don't Trust Your Eyes: On The Unreliability of Feature Visualizations." *ICML (2024)*
Poster NeurIPS	**Roland S. Zimmermann**, David Klindt, and Wieland Brendel. "Measuring Interpretability at Scale Without Humans." *NeurIPS (2024)*
Poster ICML	**Roland S. Zimmermann***, Yash Sharma*, Steffen Schneider*, Matthias Bethge, and Wieland Brendel. "Contrastive Learning Inverts the Data Generating Process." *ICML (2021)*
Oral ICML	Jack Brady*, **Roland S. Zimmermann***, Yash Sharma, Bernhard Schölkopf, Julius von Kügelken, and Wieland Brendel. "Provably Learning Object-Centric Representations." *ICML (2023)*

Publications Not Included in This Thesis

Oral ECCV	Evgenia Rusak*, Lukas Schott*, **Roland S. Zimmermann***, Julian Bitterwolf, Oliver Bringmann, Matthias Bethge, and Wieland Brendel. "A simple way to make neural networks robust against diverse image corruptions." *ECCV (2020)*
Journal	Jonas, Rauber, **Roland S. Zimmermann**, Matthias Bethge, and Wieland Brendel. "Foolbox native: Fast adversarial attacks to benchmark the robustness of machine learning models in pytorch, tensorflow, and jax." *Journal of Open Source Software (2020)*

Poster Workshop	**Roland S. Zimmermann**, Lukas Schott, Yang Song, Benjamin Adric Dunn, and David Klindt. "Score-based generative classifiers." *NeurIPS Workshop on Deep Generative Models and Downstream Applications (2021)*
Poster NeurIPS	**Roland S. Zimmermann**, Wieland Brendel, Florian Tramer, and Nicholas Carlini. "Increasing confidence in adversarial robustness evaluations." *NeurIPS (2022)*
arXiv	**Roland S. Zimmermann**, Sjoerd van Steenkiste, Mehdi SM Sajjadi, Thomas Kipf, and Klaus Greff. "Sensitivity of Slot-Based Object-Centric Models to their Number of Slots." *arXiv (2023)*
Poster Workshop	Evgenia Rusak, Patrik Reizinger, Attila Juhos, Oliver Bringmann, **Roland S. Zimmermann**†, and Wieland Brendel†. "Contrastive Learning: Reducing the Gap Between Theory and Practice." *GRaM Workshop ICML (2024)*
Poster NeurIPS	Prasanna Mayilvahanan*, **Roland S. Zimmermann***, Thaddäus Wiedemer, Evgenia Rusak, Attila Juhos, Matthias Bethge, and Wieland Brendel. "In Search of Forgotten Domain Generalization." *NeurIPS (2024)*

2 Understanding the Internal Information Processing of Neural Networks

This chapter presents work on understanding neural networks via the bottom-up approach introduced in Chapter 1. This approach aims to better understand networks by comprehending their inner workings and building blocks. Guided by the four research questions (RQ 1.1 to 1.4) introduced above, five research papers and their main results will be presented and discussed at a high level. The complete results in the form of published research papers can be found in the Appendix.

2.1 How Useful are (Feature) Visualizations for Understanding Units?

Understanding the internal information processing of neural networks requires access to the right tools. Although one might be interested in eventually understanding how the entire network or large submodules within the network operate, one needs to break this task down, e.g., by understanding single units. Therefore, interpretability tools must first convey which information, i.e., visual features in the input, a single unit is responsive to. Here, two aspects are especially important: The tools should work for arbitrary and not only for a few special network units, and they need to display information so that humans learn something from them. I now present two objective benchmarks to quantify how well humans can understand units based on the existing explanation tools.

2.1.1 Do Visualizations Enable Humans to Predict Neural Activations?

This section summarizes:

Judy Borowski*, **Roland S. Zimmermann***, Judith Schepers, Robert Geirhos, Thomas SA Wallis, Matthias Bethge, and Wieland Brendel. "Exemplary Natural Images Explain CNN Activations Better than State-of-the-Art Feature Visualization." *ICLR (2020)*

The full publication can be found in the appendix on page 86. * Equal contribution.

Author Contributions

The initiative to investigate the human predictability of CNN activations came from Wieland Brendel (WB). Judy Borowski (JB), WB, Matthias Bethge (MB), and Thomas S. A. Wallis (TSAW) jointly combined it with the idea of investigating the human interpretability of feature visualizations. JB led the project. JB, Roland S. Zimmermann (RSZ), and Judith Schepers (JS) jointly designed and implemented the experiments (with advice and feedback from TSAW, RG, MB, and WB). The data analysis was performed by JB and RSZ (with advice and feedback from Robert Geirhos (RG), TSAW, MB, and WB). JB designed, and JB and JS implemented the pilot study. JB conducted the experiments (with help from JS). RSZ performed the statistical significance tests (with advice from TSAW and feedback from JB and RG). MB helped shape the bigger picture and initiated intuitiveness trials. WB provided day-to-day supervision. JB, RSZ, and RG wrote the initial version of the manuscript. All authors contributed to the final version of the manuscript.

Motivation

Visual explanations such as feature visualizations were successfully used to understand some parts of neural networks [58, 82]. However, so far, it is unclear *how* helpful they are as no quantitative benchmark of their helpfulness exists yet. Considering the community's interest in developing better methods [63, 66, 67, 83], it is surprising that there is no objective way of comparing methods — after all, how should one determine which method works better without a quantitative comparison? Much of the development has been driven by intuition and qualitative evaluations [26]. Further, exisiting evaluations focused almost exclusively on the visual quality of visualizations. However, as introduced in Section 1.1, various researchers in the XAI community called for quantitative evaluations for explanation methods. We follow this call and aim to evaluate (feature) visualizations quantitatively. Specifically, we quantify how helpful they are for humans to understand a unit's behavior, measured by their ability to (coarsely) predict a unit's activation for some input. Our evaluation paradigm tests two important properties of visual explanations: First, it assesses how much information the explanation conveys about a unit. Second, it tests how easily this information is accessible and comprehensible for humans. While previous qualitative evaluations implicitly considered the latter property by assessing the visual quality of visualizations, they did not rigorously test the former.

Results

This study presents a general psychophysical setup for evaluating the helpfulness of visual explanations for humans. We conduct psychophysical experiments implementing a two-alternative-forced-choice (2-AFC) design. Specifically, humans are given one strongly and one weakly activating dataset sample for a unit (so-called query images) and have to identify the more activating one. Solving this task requires participants to know which features a unit is sensitive to. To supply this information the participants also see visual explanations of the unit (so-called reference images). This task is motivated by the reasoning that if the visual explanations convey information about the behavior of a unit and its feature sensitivity, humans should be able to (coarsely) predict the unit's activation, meaning they should at least be able to distinguish two samples eliciting activation values at very different levels.

By applying this setup to feature visualizations, we perform the first quantitative analysis of them. We analyze the interpretability of a subset of units[1] from GoogLeNet [79], a common choice in the mechanistic interpretability community. We find that synthetic feature visualizations provide humans with helpful information about the behavior/feature sensitivity of CNN units: Human participants can solve our 2AFC tasks with an accuracy of $82 \pm 4\,\%$ which is significantly above chance level (i.e., $50\,\%$). However, we also see that the technically simpler visualization technique of natural dataset exemplars leads to even higher performance at $92 \pm 2\,\%$. Moreover, participants are more confident in their choices and solve tasks faster when using dataset exemplars as explanations.

To investigate the generality of feature visualizations' helpfulness, we compared human performance for two groups of network units: (1) randomly chosen ones and (2) hand-picked units that were used by earlier studies to illustrate the utility of feature visualizations [48]. While we find slightly higher performance for hand-picked units when using feature visualizations, this difference is not statistically significant. An extension of our experiments that considers more units and, thus, produces more data is therefore needed to answer this question with certainty.

Understanding the behavior and feature sensitivity of CNN units through visual explanations might benefit from prior experience and knowledge, such as which features convolutional filters or biological neural networks tend to use for image processing. This means that such explanations might be less amenable for lay people than for experts. Surprisingly, however, a comparison of human performance between experts (i.e., machine learning and computer vision scientists) and lay people (i.e., people with no technical background) shows no significant difference in their performance — neither for feature visualizations nor dataset exemplars. We conclude that future experiments can be conducted with lay people who are easier to recruit.

Previous publications differ in how they present feature visualizations: Some show a single visualization [102, 191, 192] and some show multiple [83, 87]; some show only visualizations of the features and few also show those of the antifeatures [48]. Naturally, this prompts the question of how these differences in presentation influence human understanding of units. Our investigation reveals that showing more visualizations benefits both visualization techniques investigated. This benefit is especially pronounced for dataset exemplars. For feature visualizations, we see the strongest gain in performance when showing not only visualizations of a unit's feature but also of its antifeature.

[1] We investigated 89 units, consisting of randomly chosen units and some that were listed as particularly interpretable units in previous work [48].

Discussion

Conducting well-controlled psychophysical experiments to quantify interpretability is challenging. The chosen experimental setup proposed in this study probes the forward simulation abilities of humans [12], i.e., how well they can predict a unit's activations based on explanations. However, this is only one possible way of quantifying the interpretability of a unit, and other paradigms might produce further insights. Specifically, our task design does not test for a fine-grained understanding of a unit's sensitivity: A unit may be only sensitive to one feature (feature A) but this feature naturally rarely occurs in isolation but mostly with another one (feature B). In our paradigm, an explanation method that correctly isolates feature A will not perform better than one that shows both feature A and B or only feature B to a user. As feature visualizations are synthetic images not limited to any dataset, they might produce more specific visualizations (at the price of less naturalistic and, thus, harder to parse images). I will revisit this question in Section 2.1.2.

Moreover, recruiting, instructing, and supervising participants during the experiment is time-consuming. These costs constrain the experimental design and how many participants can be included. In this study, we faced these limitations on three fronts:
First, the number of units investigated is fairly low. While our main finding, that dataset exemplars outperform feature visualizations, is statistically significant, it is conceivable that a special subset of units exists for which the opposite is true: For example, units that respond to small visual features, making them hard to see in dataset examples, could be more interpretable when using synthetic explanations. However, testing all units in a network with the proposed psychophysical paradigm is infeasible. I revisit this issue in a later study (Section 2.4) for answering RQ 1.4.

Second, this study considers only one algorithm to generate feature visualizations. Generating feature visualizations requires making various choices about hyperparameters or more general design choices, such as how to parameterize the image or which augmentations to use to stabilize the optimization. Due to the experimental constraints, we could only test one algorithm and set of choices. Although we chose the most commonly used method [48], other variants of feature visualizations may perform differently. We hope that our experiments will be used as a benchmark for developers of future (feature) visualization techniques.

Third, our psychophysical experiments come with various design choices. Most importantly, how the query images are sampled influences the task's difficulty: We make the task as easy as possible by choosing query images that yield the most extreme activations, testing human understanding only for few images. On the other hand, while choosing them from less extreme activation ranges will make the task harder, it can give us more general insights into how well humans understand a unit's activation for most images. This limitation will be revisited in Section 2.3 when investigating RQ 1.3.

2.1.2 Do Visualizations Support Causal Understanding of Neural Activations?

This section summarizes:

Roland S. Zimmermann*, Judy Borowski*, Robert Geirhos, Matthias Bethge, Thomas Wallis, and Wieland Brendel. "How Well do Feature Visualizations Support Causal Understanding of CNN Activations?" *NeurIPS (2021)*

The full publication can be found in the appendix on page 128. * Equal contribution.

Author Contributions

The idea to test how well feature visualizations support causal understanding of CNN activations was born out of several reviewer and audience comments on our previous paper presented in Section 2.1.1. The first idea of how to test this in a psychophysical experiment came from Thomas S. A. Wallis (TSAW). Judy Borowski (JB) led the project. JB, Roland S. Zimmermann (RSZ), Wieland Brendel (WB), and TSAW jointly improved the experimental set-up with input from Matthias Bethge (MB) and Robert Geirhos (RG). RSZ led, and JB helped with the implementation and execution of the experiment; JB led, and RSZ contributed to the generation of stimuli. RSZ and JB both coded the baselines, and TSAW guided the replication experiment with statistical power simulations. The data analysis was performed by RSZ and JB with advice and feedback from RG, TSAW, WB, and MB. TSAW and WB provided day-to-day supervision. While JB and RSZ created the first draft of the manuscript, RG and TSAW heavily edited the manuscript, and all authors contributed to the final version.

Motivation

Feature visualizations can be more precise explanations than dataset exemplars. While dataset exemplars can theoretically explain units responding to arbitrary complex features assuming an infinitely large dataset, this does not apply in practice. Here, due to the finite size of the dataset, e.g., the training dataset of the network, visual explanations sampled from the dataset can be imprecise: They can contain visual features that often co-occur with the relevant feature a unit is sensitive to, suggesting that the unit responds to either one of these features or the joint presence of them. Contrarily, feature visualizations do not suffer from these spurious correlations and are potentially more precise: They are synthetic images generated to elicit strong activations — independent of any dataset. For this reason, previous work [48] praises feature visualizations as a more powerful tool than only using dataset exemplars.

While my study presented in the previous section finds an advantage of dataset exemplars when probing how well humans can predict the activations of images, it does not specifically test for the precision of visualizations. However, feature visualizations are argued to allow a better understanding of which visual feature *caused* high activation compared to dataset examples that also show features that are spuriously correlated with the relevant one [48, 75]. We now revisit this issue and propose a new psychophysical evaluation paradigm that tests how precise the information of visual explanations is. This paradigm tests whether humans can identify the parts of a strongly activating image that are least and most important for the image's high activations. The underlying motivation here is that if an explanation

method does provide fine-grained information about the single feature a unit is responsive to (compared to multiple ones out of which one one is relevant), then humans should be able to identify the most relevant patch of an image.

Results

To add a new facet to the quantitative evaluation of visual explanations presented in Section 2.1.1, we introduce another psychophysical evaluation probing how well humans can identify the most important patch of an image. As argued above, we use this to test whether visualizations enable a *causal* understanding of network activations. Our experiment again follows a 2-AFC paradigm. Specifically, users are shown a highly activating dataset sample in three versions: once unperturbed, once with a square occlusion with a fixed size placed to reduce the activation as much as possible, and once with a square occlusion placed to minimally perturb the activation. Supported by a set of visual explanations (reference images), participants now have to decide which of the two occluded images yields higher activation. Our motivation for this task is as follows: If the explanations support causal understanding of activations, participants should be able to predict the effect of an (image) intervention and, thus, should be able to solve this task.

Continuing our evaluation visualizations in the context of understanding units of GoogLeNet shows that feature visualizations provide humans with some helpful information about the most important image patch: Humans solve our 2AFC tasks with an accuracy of $67 \pm 4\,\%$ which is above the theoretical chance level. Further analysis shows, however, that humans achieve similar performance ($60 \pm 3\,\%$) when prompted without any explanations. As this accuracy is above the theoretical chance level, we conclude systematic biases exist in what constitutes the most important image patch for a unit.

Moreover, comparing the performance of feature visualizations, which were praised for their strong causal explanations, to other explanation methods further darkens the picture: Augmenting versions of dataset exemplars (i.e., by coarsely blurring less important image parts) or mixtures of dataset exemplars and feature visualizations perform all similarly and achieve the same performance as feature visualizations. This suggests that feature visualizations do not enable a causal understanding of network activations particularly well.

The observation that humans performed above the theoretical chance level in tasks without explanations prompts further investigation. For this, we leverage multiple explanation- and, thus, unit-agnostic decision strategies to simulate human behavior. While various strategies achieve similar accuracy (between $61\,\%$ and $63\,\%$), one method additionally shows high agreement with human responses on a by-task level: This strategy simulated human behavior by always picking the image in which the occlusion was placed at a less salient image region, as judged by a human saliency prediction model [193].

While the accuracy of human participants in our study only mildly depends on the visualization technique used, we find more important factors: First, we see that the performance varies across different units. Second, this also holds for different tasks (i.e., different strongly activating images with occlusions) of the same unit. However, the low number of participants per task might partly cause this variance. Third, the relative difference between the activations elicited by the two occluded images is a good predictor of human performance for many units. In light of the low average performance of humans in our task, this raises the question of whether the activation differences have been too low, i.e., the tasks have been too difficult to see substantial performance differences of the explanation techniques.

Discussion

This study is the first to investigate the claim that feature visualizations enable a *causal* understanding of network activations. While we shed some light on this question, we have not explored every aspect yet. Specifically, we see two major directions to be explored:

First, it is important to note that we present only one experimental paradigm to quantify causal understanding, but certainly not the only one. Our setup is limited in that the interventions are all square occlusions of the same fixed size. Although we deemed this constraint necessary to control the experiments well, it certainly influenced the precision of the interventions. This imprecision might result in harder tasks, as the activation difference between the two occluded images is not that pronounced. One might make the tasks more informative by constraining the interventions not to a fixed size and shape but by how strongly they impact a unit's relative activation. Moreover, completely different setups requiring participants to highlight the area of an image they deem most important for a unit's strong activation could also be used to probe causal understanding.

Second, our experimental setup could be applied on a larger scale: Although none of the investigated explanation methods performed particularly well in our benchmark, it is conceivable that later released variants of feature visualizations [e.g., 70] perform differently. In another direction, it is also possible there are differences between explanation methods when analyzing units of networks other than GoogLeNet. We hope that our work will serve as a powerful benchmark for research on building better explanation methods as well as on building more interpretable neural networks.

2.2 How Reliable Can Any Feature Visualization Method Be in General?

This section summarizes:

Robert Geirhos*, **Roland S. Zimmermann***, Blair Bilodeau*, Wieland Brendel, and Been Kim. "Don't Trust Your Eyes: On The Unreliability of Feature Visualizations" *ICML (2024)*

The full publication can be found in the appendix on page 194. * Equal contribution.

Author Contributions

The project was led and coordinated by Robert Geirhos (RG). Wieland Brendel (WB) developed the core idea that the argmax does not constrain a function sufficiently, which can be exploited in order to manipulate feature visualizations (key insight behind Section 2 and Section 4). RG had the idea for Section 2.1; RG and Roland S. Zimmermann (RSZ) conducted the experiments. RSZ and WB had the idea for Section 2.2; RSZ conducted the experiments. RG had the idea and ran the analysis for Section 3 based on discussions with Been Kim (BK). RG conceived of Lemma 1, and BB proved it with input from RG and RSZ. Blair Bilodeau (BB) conceived of Lemma 2, and BB proved it with input from RG and RSZ. BB and RG conceived of the main results in Section 4. BB formalized and proved the results in Section 4 and the corresponding appendix. WB had the idea; RSZ ran the analysis for Section 5 based on discussions with WB and RG. BK, and at a later stage WB, provided overall guidance throughout the course of the project, helping with presentation and experiment details. The first draft was written by RG apart from Section 2.2 (RSZ), Section 4 (BB; intro jointly with RG) and Section 5 (RSZ and RG). BB curated the final presentation of theoretical results and plain-language descriptions with input from RG, RSZ, and BK. All authors contributed to the final writing.

Motivation

While different tools to explain the inner workings of neural networks and the behavior of single units exist, most of them follow the same paradigm: They attempt to explain a unit via the input features the unit is sensitive to by showing (or describing) potential strongly activating inputs. While they differ in how they find these strongly activating samples and in how they present them to users, they can all be reduced to finding the most strongly activating samples. Mathematically, this corresponds to finding the location of the maximum and minimum of the unit's response function. In general, it is common knowledge that without further assumptions, information only about the argmin and argmax do not fully describe a function. Therefore, one should ask whether the current paradigm of explaining units can reliably be used to understand units. Moreover, as introduced before in Section 1.1, we already know of the phenomenon of adversarial susceptibility, meaning that small perturbations of the input can produce almost arbitrary activations. This makes it even more pressing to investigate whether optimization and especially gradient-based visualization techniques show meaningful features or whether they only display adversarial artifacts. This study asks the simple but important questions: How reliable can such an explanation method be in general? How much additional information about the unit or network does one need to rely on its explanations?

Results

In this study, we demonstrate that a common approach to (visually) explain network units, e.g., feature visualizations, needs to be used with a grain of salt. By taking three different perspectives, we see that the paradigm of looking at strongly activating samples has generally no theoretical justification and can be manipulated to intentionally mislead practitioners:

By taking an adversarial perspective, we demonstrate that feature visualizations can be fooled and are not reliable: They can be manipulated to show arbitrary patterns disconnected from the features a unit is responding to in production systems (e.g., the test datasets). We introduce two methods of fooling visualizations that serve as proofs of concept: In our first approach, we introduce a so-called *fooling circuit*, which is inserted at the end of the network and manipulates the visualizations of the class neurons in an image classifier. This circuit works for any visualization technique synthesizing images that can be distinguished from standard (natural) test samples. Specifically, it leverages their perceptual difference to change the behavior of a unit when feature visualizations are computed. Our second method is based on *orthogonal filters*, and does not explicitly distinguish between natural and synthetic images. Instead, it leverages the fact that (synthetic) visual explanations often elicit much higher activation than (natural) test samples (see Section 2.1.1). We modify a convolutional network by introducing a new layer whose output is added to the output of a layer that needs to be manipulated. The added layer controls the feature visualizations. Leveraging the difference in activations between visualizations and natural samples, the layer uses a strong negative bias to remain silent unless strong activations occur, which can only be caused by the synthetic patterns of feature visualizations. The layer's filter is equal to that of the original layer except for a newly added (strong) direction orthogonal to the original filter, which determines how the manipulated visualization looks. We demonstrate both proofs of concept by showing they can (arbitrarily) manipulate an existing network's feature visualizations without substantially changing its downstream performance.

Our theoretical work shows that strong assumptions on the network are necessary to guarantee that (feature) visualizations are reliable. We investigate whether visualizations can reliably be used to predict a unit's activation exactly, approximately or at least distinguish between a strongly and weakly activating input. Note that the third setting corresponds to the experimental setup of Section 2.1.1. By analyzing the worst-case outcome for various function classes, we see that visualizations are only provably reliable if one poses fairly strict assumptions on the network: Only for units known to be simple affine functions do these visualizations solve the first two tasks; the third task can also be reliably solved for networks with sufficiently small Lipschitz constant. This theoretical finding confirms the intuitive skepticism of this extreme visualization paradigm described above and aligns with our experimental proofs of concept.

In an empirical direction, we propose a sanity check for (feature) visualization techniques. Here, we argue that any visual explanation needs to be processed similarly to real test samples (i.e., samples the network will be applied to later in production systems), to inform users about a unit's behavior on such real data. If visualizations are processed along very different paths and subnetworks, it is conceivable that they illustrate behavior different from the unit's relevant behavior. Our sanity check starts by computing (feature) visualizations for units in the last layer, where every unit corresponds to one semantic class. Then, we compare the activations these visualizations cause with those caused by real samples of the same or different classes throughout the entire network. A visualization method passes our check if its visualizations are processed more similarly to natural (test) samples of the same class than those of other classes.

Discussion

This study demonstrated the *unreliability* of visual explanations such as feature visualization both theoretically and experimentally. Our results suggest that such explanations should not be taken at face value but must be critically interpreted — especially if these explanations are used to audit a model by regulatory agencies.

While our experiments demonstrated that synthetic feature visualizations can be fooled and are unreliable, practitioners rarely only use a single visualization method to understand neural networks [102]. However, our theoretical results are more general and show the unreliability of any visual explanation method depicting only the (anti-) feature a unit responds to. Furthermore, concurrent work experimentally demonstrated the unreliability of another commonly used visualization technique [194]. Future work needs to explore whether multiple visualization methods can be fooled jointly or whether such a combined approach has higher reliability.

In the long term, our work suggests major changes in interpretability are needed to reliably explain neural networks. We highlight two specific directions:
First, one can look for new explanation methods following a new and different paradigm: Instead of illustrating a unit's behavior through few, very isolated examples, they need to paint a more holistic picture of the unit. Only if the visualizations are representative of a sufficiently large number of (realistically occurring) input samples can we hope to understand a unit's relevant behavior. Potentially, this can be achieved by showing how a unit's activation changes when performing a realistic interpolation between the extreme explanations. Besides visual explanations, textual explanations exist, too, and have gained popularity. Expanding our reliability analysis to these explanations will be important for future work. However, as many textual explanation methods internally use visual explanations, it is plausible that they can be fooled, too.

Second, one can move away from black-box model explanations toward inherently interpretable models. This consequence is in line with previous work on other flavors of explainability [195–198]. Following the theoretical approach introduced in this work, one could look for architectural primitives enabling reliable visual explanations, e.g., more linear units.

2.3 Do Model and Training Design Choices Influence the Per-Unit Interpretability?

This section summarizes:

Roland S. Zimmermann*, Thomas Klein*, and Wieland Brendel. "Scale Alone Does not Improve Mechanistic Interpretability in Vision Models." *NeurIPS (2023)*

The full publication can be found in the appendix on page 160. * Equal contribution.

Author Contributions

Roland S. Zimmermann (RSZ) and Wieland Brendel (WB) conceived the idea for the project as a continuation of their earlier work, and Thomas Klein (TK) joined at an early stage. Roland S. Zimmermann (RSZ) led the project. WB initiated and supervised the project. RSZ and TK jointly implemented and conducted the experiment, building heavily on the existing setup by RSZ, with advice and feedback from WB. TK contributed code to extend the preparation of natural and synthetic stimuli to support multiple models with help from RSZ. The a priori power analysis was done by TK. RSZ conducted the final analysis and was responsible for the figures with contributions from TK. The manuscript was written jointly by RSZ and TK with advice from WB.

Motivation

An increased understanding of the inner workings of a network can either come from better interpretability tools or from more interpretable neural networks. While there has been continuous interest in building better (visual) interpretability methods [48, 66, 67, 83], the overall progress in terms of fully understanding neural networks has been slow. At the same time, however, we have seen tremendous improvements in building vision neural networks, as measured by various downstream applications [103, 104]. The most important change was to scale up networks and training datasets in terms of parameters and samples, respectively [37, 104]. Moreover, previous studies have found that these networks also make more human-like decisions [103], which could come with more human-like and especially more human-understandable decision strategies, resulting in higher interpretability. Therefore, we now turn our interest away from interpretability methods and focus on networks: We investigate whether this scaling trend led to more interpretable models. By comparing models differing in various design choices, we analyze the potential influence of these design choices on the per-unit interpretability afforded by current explanation methods.

Results

This study scales up the psychophysical experiments conducted to investigate RQ 1.1 in Section 2.1.1. We use the same paradigm but now compare nine different models in terms of their per-unit interpretability afforded by two standard explanation methods — dataset exemplars and feature visualizations. This allows us to analyze the potential effects of design choices, such as the training objective, the network's architecture and parameter count, and the training dataset, on a model's interpretability. For the sake of feasibility, we cannot

investigate all units of these models but use random subsets of units, for a total of 768 investigated units.

Scaling models up has not coincided with improving their per-unit interpretability. While we test for potential scaling effects along four design axes, we find almost no evidence for these axes to impact interpretability. Neither the size of the model ($6.8\,\mathrm{M}$ vs. $89\,\mathrm{M}$ parameters), nor that of the dataset ($1.2\,\mathrm{M}$ [199] vs. $400\,\mathrm{M}$ [200]), nor model architecture (convolutional neural network vs. vision transformer [201]) or training scheme (supervised vs. self-supervised [37]) improve a model's average per-unit interpretability. One notable exception here is the design choice of using a training procedure known for increasing the adversarial robustness of a model (i.e., adversarial training [202]): Here, we find a significant improvement in the interpretability afforded by feature visualizations ($61 \pm 3\,\%$ vs. $71 \pm 3\,\%$). Intuitively, this result can be explained by adversarial training reducing the likelihood that feature visualizations depict adversarial directions instead of sensible features. Moreover, previous work on the adversarial robustness of neural networks reasoned about the smoothness of activation surfaces [203] and connected robustness to continuous networks with constrained Lipschitz constants [204] or their linearity [205]. These results allow us to offer another explanation for the observation: According to the theory presented in Section 2.2, linear networks or networks with low Lipschitz constants are exactly those for which the used (feature) visualization techniques can be reliable. More generally speaking, we find no evidence that better downstream performance is correlated with higher interpretability. The underlying hypothesis of this study was that models with more human-like decisions might leverage decision strategies that are, consequently, easier to understand. However, disregarding whether we use the models' downstream image classification performance or their behavioral similarity to humans [103] as a measure for human likeness, we find no such trend.

This study reaffirms and generalizes my previous finding (see Section 2.1) that natural dataset exemplars are more helpful for humans and, thus, afford higher interpretability than synthetic feature visualizations. We find that the former outperforms the latter consistently for all models considered.

So far we looked at the per-unit interpretability averaged over an entire model - but how does the interpretability vary within a single model? We investigated whether the position of a unit's layer influences the unit's interpretability, i.e., are earlier layers more or less interpretable than later ones? Since we sampled units from multiple layers at various depths for our experiments, we can explore potential correlations between depth and interpretability. While we find none for most models, we detect a significant and moderate to strong correlation for models trained with the self-supervised CLIP objective [37]. As the average interpretability of those models is not significantly higher than that of their counterparts trained with supervision, we conclude that these models trade the interpretability of early layers for that of later ones.

Instead of detecting strong differences between the models in their per-unit interpretability, we find similar and high interpretability scores. While this result can be interpreted negatively, as done above, there is also a positive perspective: Maybe all of our models are already reasonably interpretable. However, one needs to remember that the psychophysical experiments used for gauging the per-unit interpretability were chosen to be as simple as possible to give us also some signal for lowly interpretable models (see Section 2.1.1): In these experiments, we only test how well humans can predict a unit's behavior for few, extremely activating, samples. But does this also mean they can predict their behavior for less extreme samples, which is, arguably, the bigger bulk of samples? We study this

for two models by using increasingly less strongly activating samples as query images in our experiments and, thus, increasing the underlying tasks's difficulty; instead of using the images with the most extreme activations, we sample from the 99 th, 95 th, or 85 th percentile. The results show that the level of understanding achieved by the explanation methods is fairly limited and brittle: Slight changes in the task's difficulty already lead to a large drop in performance (e.g., $83 \pm 2\,\%$ to $73 \pm 3\,\%$ to $63 \pm 3\,\%$ to $59 \pm 3\,\%$ for a ResNet-50).

The data collected for our analyses might serve as a starting point for future interpretability research. To achieve sufficient statistical power in our study, we have to run extensive psychophysical experiments with more than $130'000$ individual trials. As a result, we can compile the first dataset that assigns human-perceived interpretability scores to 768 units from nine different models for two explanation methods. We expect our dataset, called *ImageNet Mechanistic Interpretability* (IMI), to be a valuable resource for finding machine-based metrics or heuristics that predict how interpretable humans find a unit. For one, such a metric could be a helpful analysis and debugging tool. For another, it could also be used to train more interpretable neural networks by explicitly optimizing for this metric. Finding such a metric is part of RQ 1.4, and results will be presented in Section 2.4.

Discussion

Human psychophysical trials are costly and limit the scale of our experiments, as stated above in Sections 2.1.1 and 2.1.2. Therefore, we were limited in how many models could be included in our comparison. Moreover, adding more models to the comparison increases the number of trials needed per model to reach the same statistical power when comparing their performances. We limited our analysis to four design axes to reconcile this constraint with the desire to investigate the influence of various design choices. Through an efficient choice of nine models, we could coarsely test the influence of these four design axes. However, it needs to be noted that one cannot conclude from our experiments that no more interpretable model exists: While we chose the most promising model candidates, it is still conceivable that some models with higher average per-unit interpretability exist that were not included in our experiments. Similarly, our experiments were also limited in how many units were included per model. Due to the cost constraints of the experiments, we determined that 86 units per model are sufficient to reach reasonable statistical power. However, if the true distribution of units' interpretability differs severely and systematically from the one assumed in the power analysis, it could mean that we have missed differences in models. We will revisit this issue and the limited number of models later in Section 2.4 after lifting the experimental constraints through an automated interpretability measure.

In our study, we do not quantify a unit's interpretability as a general property but as a property depending on the explanation/visualization tools used. This means that the choice of tools is crucial. While various explanation and visualization techniques exist, we could not consider all of them because of the high costs involved in conducting the psychophysical evaluation. Instead, we restricted our study to the two best-performing and most commonly used tools, previously analyzed in Section 2.1. Therefore, it is conceivable that the models tested already display a larger difference in interpretability — we simply do not have the right explanation tools yet.

This study's motivation is based on the reasoning that humans might find it easier to interpret neural networks that behave more human-like. However, as this study produced a negative outcome, i.e., more human-like models are not necessarily more interpretable, it is advisable to revisit this argument. While it might be true that more human-like models use visual

features akin to those humans internally and unconsciously use, there is no guarantee that humans can understand them: After all, neuro- and vision scientists still have not deciphered the human brain, its circuits, and all relevant features [206].

2.4 Can We Automate the (Human-Centric) Quantification of the Per-Unit Interpretability?

This section summarizes:

Roland S. Zimmermann, David Klindt, and Wieland Brendel. "Measuring Per-Unit Interpretability at Scale Without Humans." *NeurIPS (2024)*

The full publication can be found in the appendix on page 232.

Author Contributions

Roland S. Zimmermann (RSZ) led the project, which David Klindt (DK) and RSZ initiated. DK proposed using perceptual similarity functions to build an interpretability metric. RSZ and Wieland Brendel (WB) conceived the final formulation of the metric. RSZ conducted all the experiments with suggestions from WB and feedback from DK. RSZ executed the data analysis, except for the estimation of the noise ceiling conducted by DK. RSZ created all the figures in the paper and wrote the manuscript with suggestions from DK and WB.

Motivation

A fast and scalable interpretability measure can be a crucial tool for better understanding neural networks. Such a measure would allow fast hypothesis testing in the real world of interpretability, which can be valuable for finding new approaches to interpreting existing networks. Furthermore, it could be used to build new networks that are explicitly trained to be more interpretable. For a long time, no objective measure of per-unit interpretability existed. Section 2.1.1 introduced the first measure, but the necessary human labor renders this measure costly due to its reliance on psychophysical experiments. The high cost of hypothesis testing with this measure limits the opportunities to explore and conduct open-ended research. Therefore, we are now looking for a way to automate this human evaluation to substantially reduce its cost. It is important, however, to note that interpretability is an inherently human-centered task, as, eventually, humans should understand the inner workings of networks. Hence, in this study, we look for a fully automated interpretability measure that is fast and scalable and also shown to be well-aligned with human judgments. This is in contrast with earlier attempts for LLMs that skipped human validation [106], for follow-up work to identify issues with the proposed automated metrics [107].

Results

While the previous study in Section 2.1.1 introduces a psychophysical task enabling the quantification of the interpretability perceived by humans, its application is limited due to its reliance on costly human labor. However, we can hope to automate the quantification with the great advances in computer vision. In this study, we propose the *Machine Interpretability Score* (MIS) as a fully automated interpretability measure. We leverage the latest generation of perceptual image similarity functions [108–110] to solve the underlying task of the psychophysical task proposed in Section 2.1.1 and used in Section 2.3. Specifically, we use these functions to build a classifier that follows the same solution strategy as humans: By

comparing the visualizations of both the most important feature and anti-feature to every positive and negative reference image with the two query images, the classifier picks the query image more similar to the positive references. Through experimental exploration, we realized that the predicted *confidence* is a good predictor of the human-assigned interpretability scores when averaged over multiple tasks of the same unit (i.e., using different query images and possibly different explanations).

We begin our study by validating the proposed MIS measure, ensuring alignment with human interpretability annotations. Here, we perform correlational and interventional experiments. First, by leveraging the human interpretability annotations in the previously presented IMI dataset (Section 2.3), we demonstrate that the MIS is highly correlated with human annotations, both when analyzing the average interpretability of a model (Spearman's correlation $\rho_s = 0.94$, Pearson's correlation $\rho_p = 0.98$) or of individual units ($\rho_s = \rho_p = 0.80$). Further statistical evaluations explain the sub-optimal correlation on a per-unit level: Due to the limited number of human responses for individual units in the IMI dataset, these annotations have uncertainties, resulting in a noise ceiling ($\rho_s = \rho_p = 0.82$) matching the performance of MIS. Second, we show evidence that the MIS can also be used for *novel* predictions in an interventional experiment. The IMI dataset contains annotations for randomly sampled units; we now use our MIS to determine two sets containing the most and least interpretable units, respectively. Applying the psychophysical paradigm from Sections 2.1 and 2.3 to these two sets allows us to evaluate their predictive power: We find that, as predicted by the MIS, the hardest units do yield lower scores than the random units in IMI which yield lower scores than the easiest units. Taken together, this is strong evidence for the alignment between the proposed MIS and human interpretability annotations.

The proposed MIS allows fast and cheap interpretability evaluations. We demonstrate this through a series of experiments. First, we scale previous evaluations up by multiple orders of magnitude: The most extensive evaluation yet (see Section 2.3) considered 768 units sampled across 9 models. We now investigate *every* unit of 835 standard computer vision backbones [207], suitable for ImageNet classification [199]. This results in a truly large-scale evaluation of more than $80\,\text{M}$ units — five orders of magnitude bigger than the previous one. We analyze the results along multiple dimensions:

First, we use it to revisit RQ 1.3 and compare the average per-unit interpretability of 835 models. Interestingly, we again find that GoogLeNet, previously praised for its seemingly high interpretability, is the best-performing model. However, note that the differences in interpretability scores are limited ($80.60\,\%$ vs $90.76\,\%$ between best and worst model).

Second, our additional data allows us to finally answer whether better classifiers are more interpretable models: We find a significant anticorrelation between a model's ImageNet classification performance and its average MIS.

Third, we compare how the type of layer, its relative width, or depth influence its units' interpretability. Regarding the first question, we find that units in linear and convolutional layers are mostly more interpretable than those in normalization layers. Secondly, we see that layer depth has a non-trivial influence on a unit's interpretability: It first has a negative influence up to the first fifth, then a positive influence up to a peak around the fourth fifth before having a strongly negative influence again. Thirdly, we see evidence that wider layers are slightly more interpretable than narrower ones. However, the layer width and depth results need to be taken with a grain of salt due to the relatively small overall differences, which, although statistically significant, might not be relevant.

Besides this large-scale comparison, we also demonstrate that the MIS can be used to understand learning dynamics better. While the previous results show that better classifiers are, on average, less interpretable, there are many potentially confounding factors, such as different architectures or training datasets. By tracking the MIS for each unit of a ResNet-50 [5] image classifier during its training, we can eliminate these confounding factors and obtain new insights into how classification training influences interpretability. We see that the MIS of a randomly initialized network is surprisingly high (≈ 0.75) and peaks after training for a single epoch (0.93). From there, it steadily decreases throughout the rest of the training. Manual inspection reveals a strong color sensitivity of units in the randomly initialized network, which explains their surprisingly high interpretability scores. Further, we find that most of the changes in interpretability are caused by a distinct set of batch normalization layers. We have not yet uncovered why these layers are especially important or why the interpretability decreases throughout the training. As this experiment was mainly meant to demonstrate potential applications of the MIS, we leave further investigations of this for future work.

Discussion

Although this study rigorously evaluated and validated the proposed MIS as a measure of interpretability, certain caveats and room for future research exist:
First, we demonstrated a high alignment between the proposed MIS and interpretability annotations, which are either included in the IMI dataset or collected through the methodology of Section 2.3. These annotations are based on a large number of participants and are meant to represent the ability of an *average* human to understand network units. While my previous study in Section 2.1.2 has not found a significant effect of prior knowledge on this ability, it only considered fundamental knowledge, such as familiarity with convolutional neural networks, and not explicit experience in mechanistic interpretability. Therefore, it is conceivable that *experts* in mechanistic interpretability would assign different interpretability scores. It remains to be checked whether this expert knowledge would only introduce an offset in the perceived interpretability such that MIS could still be proportional or whether it changes the order of how interpretable units are drastically.

Second, while we find a close to perfect correlation between the MIS and human scores when averaging scores per model, we see a performance gap on a per-unit level. However, it is important to note that this gap can be attributed to the noise in the existing human annotations. Only collecting more human annotations will improve the noise ceiling performance and, thus, allow us to evaluate the MIS' full performance and see if it is really not perfectly correlated.

Future research needs to explore the applications of MIS further for interpretability research. Our experiments showed that the MIS allows us to cheaply quantify the per-unit interpretability afforded by an explanation method, lowering the barrier for future interpretability researchers to conduct quantitative evaluations. We hope that the MIS will make research on understanding neural networks faster, most notably through two research directions:

First, as a tool to find better explanations or more interpretable representations of existing networks. As approaches introduced to better understand existing networks by resolving their polysemanticity (e.g., SAEs introduced in Section 1.1) come with various knobs and hyperparameters to tune, access to a reliable interpretability metric will be essential for tuning them.

Second, as a tool for explicitly making neural networks more interpretable. Explicitly

optimizing for high MIS should yield more interpretable networks. While the current formulation of the MIS is not yet differentiable, using gradient-free optimizers or finding differentiable approximations of the metric will be worthwhile approaches. However, how far this direction can be pursued before being limited by Goodhart's law remains to be checked. Specifically, one needs to be careful that blindly trusting the score without any manual verification of the models can result in a detrimental development of model interpretability.

3 Understanding Neural Networks Through Theoretical Guarantees

This chapter presents work on understanding neural networks via the top-up approach introduced in Chapter 1. Contrary to the bottom-up approach of Chapter 2, the approach here aims to better understand networks by providing theoretical insights and guarantees for learning algorithms and families of networks through theoretical analysis. Guided by the two research questions (RQ 2.1 and 2.2) introduced above, the theoretical contributions of two research papers will be presented and discussed on a high level. As for the previous chapter, the Appendix contains published research papers with the full results.

3.1 How Can the Empirical Success of Contrastive Learning be Explained?

This section summarizes:

Roland S. Zimmermann*, Yash Sharma*, Steffen Schneider*, Matthias Bethge, and Wieland Brendel. "Contrastive learning inverts the data generating process." *ICML (2021)*

The full publication can be found in the appendix on page 270. * Equal contribution.

Author Contributions

The project was initiated by Wieland Brendel (WB). Roland S. S Zimmermann (RSZ), Steffen Schneider (StS) and WB jointly derived the theory. RSZ and Yash Sharma (YS) implemented and executed the experiments. The 3DIdent dataset was created by RSZ with feedback from StS, YS, WB, and Matthas Bethge. RSZ, YS, StS, and WB contributed to the final version of the manuscript.

Motivation

Understanding why a learning algorithm works is important for detecting its limitations and overcoming them. In the quest to find data-efficient learning algorithms, self-supervised methods such as contrastive learning (CL) have proven powerful [121]. The community has demonstrated CL leads to high-quality representations that generalize well to various downstream tasks [125, 131]. However, it is still not clear yet why this is the case. We here set out to investigate one representative family of contrastive losses, the commonly used InfoNCE objective (see eq. (1.1)). While theoretical work analyzing this family exists, it is not conclusive yet [183, 208]. Thus, we aim to provide a new framework enabling the theoretical analysis of this family. Besides providing an explanation for the high quality of the learned representations, such a theory can also be used to identify shortcomings of current learning algorithms and, hence, develop the next generation of contrastive losses.

Results

This study presents a novel theoretical approach to understanding CL with objectives from the InfoNCE family. Our theory starts by introducing a generative process of the observational data. Here, we assume the existence of an injective and differentiable generator mapping ground-truth latent variables that describe the state of the physical world to observations (e.g., images or videos). The theory then links optimizing the InfoNCE objective to solving a non-linear ICA problem (see Section 1.2 and [171, 209]) of the ground-truth latent variables. By placing assumptions on the distribution of training samples — on both negative and positive pairs (see Section 1.2) — we find that encoders minimizing the InfoNCE objective successfully solve the ICA problem and recover the ground-truth latents up to simple linear transformations. Put differently, this result means that encoders trained with InfoNCE identify the inverse of the observations' generator. As the ground-truth latent factors of the data can be seen as the most minimal lossless representation of the data, recovering them implies the encoder has learned very informative features of the data. Hence, this result offers an explanation for the empirical success of contrastive learning for learning (visual) representations without any supervision.

We derive our identifiability proof in three steps.
First, we fully define the data-generating process. Here, we introduce the ground-truth latent space, which we assume to be spherical, and an injective and differentiable generator mapping latents to observations. In addition, we put assumptions on the distribution of the latents of anchor points and their positive and negative pairs: We assume a uniform distribution for anchor points and the negative pair's second sample but a von Mises–Fisher (vMF) distribution for the positive pair's second sample.

Second, inspired by an earlier analysis [208] of InfoNCE in the limit of infinite batch sizes and, thus, infinite number of negative pairs, we reformulate InfoNCE: We show that this objective equals the cross-entropy between the ground-truth conditional distribution of the positive pair and a distribution in the space of the encoder's output. The latter distribution is implicitly defined by the loss function and its underlying similarity measure: The commonly used cosine similarity [124, 125] corresponds to also modeling a vMF distribution, while other similarity measures can result in different distributions from the family of generalized exponential distributions. Thus, when using the standard cosine similarity with InfoNCE, and under the previously stated assumptions, the modeled distribution can match the ground-truth distribution if the encoder's architecture is sufficiently expressive. Using a well-known property of the cross-entropy cross-entropy regarding its minimizers, we show that the two distributions match when minimizing the objective.

Third, we show that the equality of the conditional distribution of the reconstructed and ground-truth latents of positive pairs implies that the encoder reconstructs the ground-truth latents up to simple orthogonal linear transformations such as rotations or permutations of dimensions.

Our proof highlights implicit assumptions of the standard InfoNCE objective. Most importantly, the objective assumes that the ground-truth latents are defined on a sphere and positive pairs follow a vMF distribution. Intriguingly, our theoretical approach shows a path towards changing these implicit assumptions: Specifically, it allows us to generalize our identifiability proof and cover general convex latent spaces and positive pair conditional distributions described by the exponential of the pair's similarity measured by an arbitrary (semi-)metric. In summary, our theory provides identifiability results for the commonly used form of InfoNCE and a family of generalized variants of it, which fit to different

data-generating processes.

We also verify our theoretical results experimentally. In numerical experiments on synthetic and, thus, fully-controlled data, we observe results in line with our theoretical claims: When the correct form of InfoNCE is used, i.e., a similarity function fitting the ground-truth positive conditional distribution, encoders can perfectly recover the ground-truth latents up to linear transformations. In addition, we see that mismatches between the implicitly defined distribution of the loss and the ground-truth one do not always result in bad performance, but the severity of their mismatch controls the deterioration in reconstruction quality.

Besides numerical experiments, we also validate our theory with controlled image experiments. To verify that trained encoders invert the generative process, we need access to the ground-truth latent variables of the images. As this is impossible with existing datasets containing images with at least moderate visual complexity, we introduce a new dataset called *3DIdent*. The dataset contains renderings of a single 3D object with varying rotation, color, and position in differently colored and illuminated scenes. In line with our numerical experiments, we find that the better the implicit assumptions of the loss fit the ground truth, the better the latents get reconstructed - in the case of a perfect fit, the latents get (almost) perfectly reconstructed. Interestingly, we see that sampling positive pairs with image augmentations, as is common practice due to the inaccessibility of the ground-truth latent distribution, instead of sampling in latent space results in a substantial deterioration of the reconstructed latent's quality. This warrants future research on finding variants of InfoNCE that better fit real-world positive pair distributions.

Discussion

Although our theoretical work offers an explanation for the high quality of representations learned by encoders, which minimize the InfoNCE objective, some questions remain. As common with theoretical approaches, our results come with certain assumptions. Most importantly, as described above, we pose assumptions on the data-generating process. While they are mathematically easy to understand, interpreting them in the context of real-world applications is more difficult. Nevertheless, it will be important to understand further how well these assumptions apply in practical scenarios and extend the theory accordingly. For example, extending our theory to non-spherical and non-convex, potentially even unbounded, latent spaces will be worthwhile as they are conceivably more relevant for real data. Additionally, our work analyzed the relationship between positive and negative pairs of data in terms of their latent distributions. However, these pairs are, in practice, predominantly produced by data augmentation of the observations, showing a limitation of our approach. Overcoming this by establishing a link between the used augmentations and the latent distribution will be important to further the community's understanding of contrastive learning. On a related note, our theoretical results do not yet explain a common observation and practice in CL, namely, that the representations of the encoder's penultimate layer are more informative for downstream tasks than its last one [125]. While this gap in our theoretical understanding might relate to the previous point (i.e., using data augmentations), no theoretical results fully explain it yet.

Besides its main theoretical result, this work has also demonstrated how to develop new contrastive objectives that better reflect the structure and nature of training data sets. Our theory has revealed that the most commonly used form of InfoNCE encodes inductive biases, such as the shape of the assumed latent space or the form of the positive pair conditional distribution. Moreover, we demonstrated how to modify this objective to address the specific

nature of the (training) data. I expect this insight to be leveraged in two ways in future work: On the one hand, it can be used to develop more specialized learning objectives for special domains. On the other hand, it might lead to training objectives with fewer inductive biases and work better for general applications. Similarly, incorporating more of the data's/world's structure, such as compositionality or objects and part-whole hierarchies, into the learning problem through further inductive biases might lead to more efficient learning algorithms.

3.2 Is There a Theory for Object-Centric Learning Providing Performance Guarantees for Models?

This section summarizes:

Jack Brady*, **Roland S. Zimmermann***, Yash Sharma, Bernhard Schölkopf, Julius von Kügelken, and Wieland Brendel. "Provably Learning Object-Centric Representations." *ICML (2023)*

The full publication can be found in the appendix on page 292. * Equal contribution.

Author Contributions

Jack Brady (JB) developed the theory with technical help from Roland S. Zimmermann (RSZ), insight from Yash Sharma (YS), and advising from Wieland Brendel (WB) and Julius von Kügelken (JvK). JB implemented and executed the experiments with help from RSZ and YS, while RSZ implemented the compositional contrast and SIS metrics on image data. JB and JvK led the writing of the manuscript with help from WB, BS, and RSZ. WB and RSZ created all figures in the manuscript.

Motivation

While the previous section provides a theoretical analysis and performance guarantees for one form of unsupervised learning, i.e., contrastive learning, this one focuses on another special form: unsupervised object-centric representation learning. Object-centric learning (OCL) aims to decompose scenes into individual objects and represent them separately [134, 139]. It is motivated by the belief that factorizing the representation increases data efficiency when training and fine-tuning models and their robustness to distribution shifts [134]. Inspired by the potential benefits, there is substantial empirical work on finding OCL algorithms [189]. So far, this research has been guided mostly by heuristics and intuition, not theoretical insights. However, introducing a rigorous theoretical framework will be helpful for multiple reasons: First, it can guide future research, show limitations of existing approaches, and indicate directions for novel learning paradigms. Second, it can enable the derivation of performance and quality guarantees, increasing the trust in such methods. We, therefore, aim to introduce a theoretical framework for investigating OCL, analyze existing empirically motivated approaches, and suggest directions for future learning algorithms.

Results

Despite the large empirical interest in object-centric learning, the community still lacks theoretical guarantees about when this will be possible. This study performs a theoretical analysis of unsupervised OCL. We aim to overcome the lack of theoretical results, understand when learning object-centric representations without supervision is possible, and provide theoretical guarantees for learning such representations. To approach this challenge, we first introduce structure into the problem by specifying what constitutes an object and, hence, a multi-object scene. While various definitions have been proposed before, no commonly agreed definition exists yet. We propose to define objects through a structured latent

variable model that maps slots (i.e., subsets of latents) to multi-object scenes. We further introduce two properties, called *compositionality* and *irreducibility*, which will be defined in the following paragraphs. By assuming the generative models to be compositional and irreducible, we show how object-centric representations can be provably learned without supervision on scenes produced by the generative model. Specifically, we analyze the model's identifiability and find conditions under which inference models are guaranteed to invert the generative model and recover the ground-truth latents for each object slot. Based on this result, our work is the first to point out ways to construct future object-centric models that provably learn meaningful representations.

The core of our theoretical framework is how we define what constitutes an object. We do not make any assumption on the statistical distribution of objects or their underlying latent variables but instead put functional assumptions on the generative model. Specifically, we assume the generative model to be a diffeomorphism mapping slots of latent variables to multi-object scenes. Furthermore, we assume the generative model to be compositional and irreducible — two key properties we introduce and define through the Jacobian of the generative model:

Compositionality means the generator's Jacobian can be rearranged into disjoint (slot-wise) blocks. Put simply, each pixel in the observation is directly influenced by at most a single latent slot and not by two or more slots simultaneously. This formalizes the desideratum that a single object is described by a single latent slot. However, it does not yet define what constitutes such an object and, thus, does not ensure the division into objects is aligned with human perception. To ensure that scenes are divided into meaningful parts and slots do not describe multiple (human-defined) objects at once, we introduce the concept of irreducibility.

Irreducibility means slots should only consider two segments in the observation to be parts of the same object if they depend on each other. Instead of using the notion of statistical dependence, i.e., a global notion, we use a more local notion of dependence that describes instance-wise relations of parts in scenes akin to algorithmic dependence [210]. Here, two segments are considered dependent if they share sufficient information, so encoding them jointly is more efficient than encoding them separately. For example, dividing a solid object into multiple parts makes them dependent, as the information necessary to encode the different parts' location or orientation is fully redundant. However, merging multiple unrelated and, e.g., individually moving objects together would not be more efficient; thus, they are independent. This dependence can also be formalized through the generator's Jacobian: Two segments are independent if the rank of the Jacobian restricted to the union of both segments' pixels (also called a mechanism) is the same as the added rank of the Jacobian restricted to each segment individually. Conversely, for dependent segments/mechanisms, the sum of ranks is higher than the rank of the segment's union. We can now define irreducibility: A generative model is irreducible if no set of pixels exists that can be divided into two disjoint sets whose pixel values are described by two dependent mechanisms. Picking up the above example, a generative model is irreducible if it does not use a single slot to generate multiple objects that could be generated individually in an equally efficient way.

The previous mathematical definition of multi-object scenes by specifying their generative model allows us to investigate how object-centric representations can be learned. We derive conditions under which one can learn object-centric representations corresponding to the ground-truth latent slots without supervision. Specifically, we look for conditions under which the individual ground-truth slots can be fully recovered (up to some slot-wise non-linear transformation) — a property termed slot-identification. Our main theoretical result is as follows: If observations are produced by the aforementioned generative model, then

any encoder with the following properties slot-identifies the ground-truth latent: First, the encoder's output needs to be divided in individual slots, whose number has to equal the number of ground-truth objects and the encoder's output dimensionality must match the (unknown) ground truth. Second, the encoder must be a diffeomorphism. Third, the encoder's inverse (i.e., a learned decoder/generative model matching the encoder) has to be compositional. This is a powerful result as it states the first theoretical conditions for learning object-centric representations. While previous object-centric learning algorithms were mainly based on heuristics or intuition, this result enables the community to build novel algorithms grounded in theory. Finally, we introduce a learning objective, called *compositional contrast*, whose invertible minimizers are proven to be compositional. It enforces compositionality by discouraging the model's Jacobians of different slots from overlapping by ensuring, at most, a single slot contributes to each pixel. Equipped with this new objective, we propose a surprisingly simple, yet not particularly efficient, OCL paradigm: We augment the standard auto-encoder setup by applying our new compositional contrast to the model's decoder and minimizing it jointly with the reconstruction error. Our theory shows that if the auto-encoder minimizes both its reconstruction error and the proposed compositional contrast, it will slot-identify the ground-truth objects.

We also experimentally validate our theoretical results. We demonstrate that the described auto-encoding setup produces encoders that slot-identify the ground-truth latents on both numerical data and simple and well-controlled image datasets. Moreover, we also analyze two popular empirical learning approaches for unsupervised OCL — MONet [211] and Slot Attention [139] — that both learn a scene encoder and decoder. In line with our theory, measuring how well the learned encoders slot-identify the ground truth and how much their decoders minimize the compositional contrast shows a relation between these properties.

Discussion

This paper is the first to propose a theoretical framework for object-centric learning and investigate when it is possible to provably learn object-centric representations. Hence, it does not solve every challenge in the field but comes with some limitations: As for the previous section's theoretical result, our theoretical framework here comes with some assumptions, namely, about the data and the underlying notion of what constitutes an object. While our assumptions can be intuitively motivated and are partially in line with existing non-mathematical definitions, they might still be too restrictive for practical applications. Most importantly, the assumption of compositionality does not accommodate reflections, transparencies, or (partial) occlusions of objects. Moreover, our theory assumes the underlying generative model to be invertible, which is violated by the commonly used slot-permutation-invariant architectures used for OCL in practice. Similarly, the proposed theory requires the encoder's assumed number of slots (i.e., how many objects the encoder expects in a scene) to match the ground truth exactly. While this rarely applies in practical settings, we empirically observe that our results show some robustness against violations of this assumption, e.g., through training scenes with a varying number of objects. Thus, although our current theory does not allow this flexibility, extensions to both assumptions are conceivable. Empirically, while we introduced an algorithm for provably learning object-centric representations, it is computationally inefficient as it performs second-order optimization with high memory costs and, thus, does not scale well.

For the future, I expect this paper to guide and open up avenues for future research on object-centric learning — both on the theoretical and empirical sides: Our theoretical framework can be refined by addressing the aforementioned limitations and relaxing some of its restrictive

constraints. Moreover, while this paper demonstrated a setting in which provably learning object-centric representations is possible (i.e., sufficient conditions), analyzing whether this is the only setting (i.e., necessary conditions) will be interesting. In another direction, extending the notion of objectness to include both hierarchical definitions and task/scenario dependence will be relevant for practical applications. For example, in robotic tasks, the notion of what constitutes a single object and what is part of another object might not be static but depends on the robot's current goal —- a desideratum that cannot be reflected in the framework's current form. On the empirical side, finding a more efficient formulation/implementation of our proposed learning algorithm will be important, as its current computational costs render its usage infeasible for practical applications. Potential directions include finding more efficient approximations of the proposed compositional contrast objective or resource-efficient fine-tuning protocols. This would allow applying the proposed learning algorithm to practically relevant data.

4 Conclusion

This chapter summarizes the results presented in this thesis and reflects on my progress toward understanding how neural networks work and perceive the world. As the individual projects and research papers upon which the thesis is based have been discussed in detail before in Chapter 2 and Chapter 3, this chapter takes a broader view and presents a meta-discussion. Specifically, it revisits the previous state of research outlined in Chapter 1 and describes how my work has advanced the field. In line with the earlier structure, the bottom-up and top-down approaches for understanding neural networks will be treated separately.

Bottom-Up: Understanding the Internal Information Processing

The thesis started by identifying a conceptual gap in previous research on understanding how neural networks internally process information (see Section 1.1): While the field of (per-unit) mechanistic interpretability has received considerable attention, it has primarily relied on qualitative evaluations and driven by subjective intuition [67]. At the same time, a stronger focus on research based on falsifiable hypotheses in interpretability has been demanded in XAI before [97], which necessitates quantitative evaluations. Thus, the lack of quantitative analyses has posed a critical limitation of earlier research. The present thesis overcame this limitation by introducing quantitative evaluations to the field and demonstrating their benefits over qualitative ones. This work employs such quantitative evaluation on two fronts to advance the community's knowledge:

First, the thesis assessed the usefulness of existing explanation methods in helping humans understand individual units in neural networks (see Section 2.1 and Section 2.1.2). To meet this end, I started with qualitative claims about explanation methods, translated them into quantifiable hypotheses, and designed two concrete psychophysical experiments to test them. These experiments enable quantitative evaluations and will guide future research on explanation methods. An evaluation of the two most common explanation methods [48] at that time confirmed some qualitative claims but rejected others. More recent and, thus, potentially better performing methods have not been evaluated yet [70] or only partially [212] by their original authors. This means there is a significant gap in the current understanding of their effectiveness, and further studies are necessary to assess these newer methods. Furthermore, while this thesis introduced two approaches for estimating how useful an explanation is for humans to understand a part of a neural network, it makes no claim to completeness. As usefulness is a task-dependent property, alternative tasks can be designed that require, and thus probe, human understanding of units in different ways. Moreover, different types of explanations — visual [48, 63, 70, 212] vs. textual [71, 213, 214] — might require different evaluation paradigms. While concurrent work has proposed alternative evaluations [70, 214, 215], it is important to note that in their attempt to automate evaluations, they ignore the *human* aspect of understanding network units and can thus, at best, only paint an incomplete picture. It remains unclear how aligned these alternative metrics are with how well humans understand network units. In addition to identifying weaknesses in existing evaluations, this thesis analyzed the reliability of the currently predominant paradigm for explanations (see Section 2.2). While

the two quantitative evaluation paradigms above show that existing explanation methods provide some level of understanding to humans, the reliability analysis uncovered theoretical shortcomings. These suggest that explanations might be misleading and should always be taken with a grain of salt. While our study's experiments focused on a single explanation method, they inspired subsequent studies that empirically demonstrated the same issue for different methods, too [194, 216].

Second, after scrutinizing explanation methods, the thesis moved beyond them and compared different neural networks in their interpretability (see Section 2.3). Although the psychophysical experiments above were originally introduced to evaluate explanation methods, they could also be used to quantify the interpretability of a neural network afforded by an explanation method. To perform a model comparison, I acknowledged that existing explanation methods, although imperfect, provide *some* level of insight into neural networks and examined how much this *differs* for various networks. As psychophysical experiments come with a high cost, it has been crucial to develop an automated interpretability evaluation that produces the same results as the manual psychophysical evaluation (see Section 2.4). Compared to earlier approaches mentioned above, this novel evaluation is shown to quantify how well *humans* understand parts of neural networks. Equipped with this human-aligned automated interpretability measure, I was able to perform a large-scale comparison of models. While this comparison did not uncover a clear path to more interpretable architectures, it highlighted future research directions by finding (architectural) properties linked with higher interpretability. Previously, the lack of an automated evaluation that reliably measures how well humans understand units has been described by various works as an obstacle. Most importantly, this required researchers to use time-consuming human evaluations, resulting in partially inconclusive insights [e.g., 50, 93]. The automated evaluation presented in my thesis can overcome this obstacle — it can be leveraged in future work to perform more efficient interpretability research or directly optimize networks for higher interpretability.

Top-Down: Understanding Representations through Theoretical Guarantees

This thesis also presented approaches to understanding neural networks better through theoretical analyses. As empirical progress has been much faster than theoretical progress in recent years, a gap exists between empirical observations and theoretical understanding in various areas of machine learning. I focused on two sub-fields that have received much empirical attention. Specifically, for two types of learning algorithms/paradigms, I presented theories that elucidate when and why they work and when they fail: contrastive learning (CL, see Section 3.1) and object-centric learning (OCL, see Section 3.2).

Previous work has formulated different hypotheses explaining why contrastive learning works [e.g., 124, 183, 217]. However, no one fully explained why the learned representations are useful for downstream tasks. Our work is the first to achieve this by linking CL with the concept of identifiability. We proved that CL yields models that recover the ground-truth latent factors that fully determine the observations. While our theoretical result explains the potency of the learned representations across various downstream tasks, some gaps remain. For example, we have not yet understood the impact different data augmentation schemes have on the learned representations. Moreover, while our work presents modifications to a common contrastive loss that works better on synthetic data with certain properties, it remains to be seen whether real-world scenarios with said properties exist.

The second learning paradigm investigated in this work is object-centric learning. Here, the state of theoretical understanding has been even poorer than for contrastive learning: Due to the challenges in finding a mathematical definition of what constitutes an object, no complete theoretical framework for investigating OCL has been presented so far. While multiple attempts at finding such a definition exist in previous work, they had not been leveraged before to explain when learning object-centric representations is possible, i.e., to derive performance guarantees. In contrast, my work presents a complete theoretical framework and a new learning algorithm that comes with such theoretical guarantees. However, its implications for practical applications of OCL are not directly apparent, as the new algorithm scales poorly beyond simple synthetic data. Thus, this approach lags behind the current generation of empirically motivated learning algorithms/model architectures that were scaled to (more) realistic data over multiple years ago [139, 189]. It remains to be seen whether similar progress can also be achieved for our theoretically-grounded approach.

While this thesis' theoretical results advance the community's understanding of two popular learning and modeling paradigms, they come with limitations. First, it is important to note that any theory makes certain assumptions — how well these are met in practice can be hard or virtually impossible to verify, as they often idealize the complicated nature of real-world data. Second, while theoretical results can potentially explain the behavior of models or give some performance guarantees, they often lag behind the existing empirical observations. For example, while my work on OCL has advanced the community's theoretical understanding of such models, it does neither come with any immediate empirical implications nor practically feasible suggestions for achieving OCL. Thus, empirical progress on new methods could very well be so fast and broad that time-consuming and narrowly focused theoretical studies cannot keep up and provide sufficiently meaningful new insights into understanding these methods. Therefore, in the next chapter, when looking at potential directions for future research on understanding neural networks better, I will solely focus on empirical approaches. However, it is important to note that other, conceptually different approaches exist: For one, instead of searching for a theory explaining an entire method, one can aim to find theoretical laws explaining empirical observations of a single phenomenon to make practically relevant predictions [e.g., 218–221]. While such approaches are, strictly speaking, not theoretical, they are also not purely empirical either. The potential high impact of such approaches can be seen in other sciences such as physics [222]. For another, one can argue that the implications of theoretical work might not be first-order effects but only become noticeable far in the future. Just one of the most important practical applications of number theory, cryptography, became important only hundreds of years after its inception [223], it is conceivable that theoretical work on machine learning will pay off later, too.

5 Outlook

After discussing how this thesis advanced the community's understanding of how neural networks perceive and represent the world, it is time to consider what comes next. Based on my research results and concurrent progress made, both in terms of understanding neural networks as well as building more powerful ones, I identify four research directions as particularly promising.

5.1 Improving the Interpretability of Individual Units

Chapter 2 introduced various approaches to quantifying the interpretability of individual units in a network. These evaluations demonstrated that we have some understanding of what some units do, but our overall understanding is still fairly *limited*. Moreover, even the existing knowledge is brittle: Often, we can only understand a unit's behavior for samples yielding the most extreme activations, failing to understand their behavior for most other samples. As the latter set of samples has substantially higher cardinality, understanding how the network operates on these is essential for claiming an understanding of its function.

We can close the gap in our understanding through two means: First, one can blame the currently available explanation methods and demand more informative and robust explanations. Language-based explanations are sometimes presented as an approach for making explanations more accessible [71, 72, 214]. But at the same time, there are doubts about the preciseness of language explanations [107], rendering this an unlikely solution. Second, one can blame the networks or how we try to interpret them: At the moment, while varying in their activation strength, most units are activated by almost every sample and, thus, contribute something to the network's output. This can only be the case if units detect very general and almost omnipresent features (unlikely, since such features would not be very informative and hence useless for the classification decision) or if units detect various features simultaneously or no closely defined feature at all (more likely). Sparsifying the activation patterns of units such that each unit responds only to a few samples will greatly simplify the task of understanding the unit. However, it is important to find the right level of sparsity to target such that units react to single concepts while not responding just to single images (i.e., in the limit of full sparsity). Note that sparsity here does not correspond to the notion of weight or activation sparsity over units which is relevant for, e.g., power-efficiency [224], but instead to the sparsity of activations over input samples.

The desire for higher sparsity can be fulfilled two-fold: The post-hoc application of sparse auto-encoders (SAE) to find a sparse basis of an existing network has recently received much attention [31, 93, 225]. While current results are still limited, they promise better interpretability of existing models. However, it is important to note that due to the imperfection of trained SAEs, explanations can also get imprecise or potentially even misleading and, once again, need to be taken with a grain of salt. Alternatively, one could directly train networks to show a sparse activation pattern across samples. Compared to the post-hoc approach of SAEs, this approach comes with a higher training cost but does not introduce any further approximation or uncertainty to the network's explanation. Finally, beyond explicitly optimizing for sparsity, one could also achieve higher interpretability by directly optimizing

networks for it. Using the automated interpretability measure introduced in this thesis, or a differentiable version of it, one could explicitly train networks to be more interpretable.

5.2 Transferring the Insights on Interpretability from Vision to Language Models

With the increasing traction of large language models (LLM) in production systems and research, the focus of many interpretability researchers has shifted from the domain of vision to that of language. While this shift solved some issues in interpretability, such as the modality mismatch between the network's input (vision data) and the arguable most accessible explanations for humans (text), major issues remained unchanged. Most importantly, finding ways to rigorously quantify the interpretability of units is yet again an open challenge. In computer vision, where the community faced the same challenge, I presented multiple approaches to solve it, with a special focus on ensuring that the evaluations faithfully reflect how well *humans* can understand units. In language, however, such work is still in its early phase. While approaches such as that of Bills et al. [106] promise to quantify the per-unit interpretability of LLMs, follow-up work raised multiple issues showing its imprecision and unreliability [107, 225]. It is, thus, important to transfer the lessons learned from investigating the interpretability of vision to language models and ensure reasonable metrics are used for future optimization. This includes, first, designing and conducting psychophysical experiments to ensure alignment between proposed evaluations and the true level of interpretability perceived by humans and, second, finding new metrics with higher alignment. As demonstrated above in Section 2.4, psychophysical results can be leveraged to arrive at an interpretability measure that scales well and does not require explicit human labor. Based on the insights in this thesis, and in line with current observations in LLM research [50, 93], I expect repeating this for LLMs will be an important step to both accelerate and robustify interpretability research.

Besides the challenges observed in vision models, interpreting (multimodal) language models also comes with new ones. At first sight, language models might appear easier to interpret as their data modality is inherently more understandable for humans. After all, these models consume and/or produce text — a high-level data modality that is relatively easy to understand for humans. However, this comes with the risk of misleading explanations: Textual explanations (for LLMs) might appear sensible and understandable but can be incorrect or misleading [107]. Thus, verifying the faithfulness of explanations is even more important than for visual explanations of vision models, as for visual explanations, unfaithful ones are often incomprehensible and rarely misleading. This means that future psychophysical experiments must explicitly test their (un-)faithfulness.

5.3 Scaling Interpretability from Single Units to Entire Models

Throughout Chapter 2, I investigated the interpretability of neural networks on a *per-unit* basis. So far, and in line with previous work [e.g., 48, 66, 102], these units were restricted to individual nodes in the network's computational graph, such as single neurons in an MLP. Understanding such atomic units, however, should only be seen as a first step: As the underlying goal of interpretability is to make the behavior and decisions of the full network understandable, we eventually have to bridge the gap between single units and the full network. A promising step in this direction is (automatic) circuit analysis (CA),

which tries to identify circuits consisting of multiple interconnected units that implement a non-trivial detection mechanism [34]. While this is still a relatively young field, the literature on CA algorithms is already rich [226–235] and first experimental results appear promising [30, 227]. CA corresponds to identifying the network's most relevant subgraph for computing/detecting some high-level feature.

While algorithms for solving CA vary, most start by identifying the (causally) most relevant units and extracting the subgraph involving them [e.g., 226–230]. These approaches would greatly benefit from the target network being sparse as introduced in Section 5.1: The more specific units are, the easier it is to identify their causal relevance to a task. However, I hypothesize that activation sparsity alone is insufficient for successful CA and, ultimately, understanding networks fully. Instead, the weights connecting subsequential layers need to be sparse, too. For example, we gain relatively little if we find units responding to isolated concepts by finding a sparse basis for individual layers, but they interact in a highly unordered way in the next layer. Preferably, the weights controlling how different features of one layer contribute to those of the next layer need to be considered when sparsifying a network. Such an approach could be seen as finding a shared sparse basis for the entire network instead of multiple independent ones per layer. If done right, this will lead to a sparse computational graph, which has multiple benefits over a non-shared sparse basis. Most importantly, model debugging and identifying critical failure cases or biases will be greatly simplified.

Scaling model interpretability up from individual units to entire circuits does not only involve scaling up CA algorithms. It also entails scaling up *evaluation* paradigms: How do we quantify the progress of novel CA algorithms? When do we consider a CA to be successful? And what are important properties of sparse computational graphs worth measuring? These questions show that evaluations could focus on multiple aspects. In other fields of XAI, evaluations often focused on two properties [12]: faithfulness, which in the context of CA describes whether an extracted circuit is responsible for performing the claimed computation, and accessibility, which here means how easily humans can understand the extracted circuits. Previous work has proposed initial steps towards measuring a circuit's faithfulness but not its accessibility to humans [230]. However, for circuit analysis to boost the research community's understanding of neural networks, both properties should be optimized for. Thus, also probing the accessibility of circuit analyses to humans can be a fruitful direction for future research. However, while metrics that measure such low-level properties of explanations — be circuit analyses or per-unit explanations — grant some insights into an explanation's helpfulness, they can only approximate it at best and might not be the most meaningful. I will detail a potential paradigm shift in the next section.

5.4 Benchmarking Future Progress on Interpretability

Extracting meaningful signals out of future interpretability methods might require a paradigm shift. So far, this thesis has focused in varying detail on evaluating the helpfulness of explanations based on relatively simple tasks. However, I argue that these tasks cannot be seen as the final goal of interpretability but rather as intermediate and necessary but insufficient goals. Instead, we need to find goals that are aligned with how interpretability methods are *used in practice*. Such a goal-oriented evaluation might produce a weak signal at the moment since interpretability methods are not fully mature yet. However, if these methods ever become practically relevant, goal-oriented evaluations will become crucial for giving us further signals for refining them. Thus, I argue that we should, in addition to existing evaluations, also strive for those capturing how interpretability methods could be used in practice for

real networks. Recently, interpretability researchers started demonstrating the usefulness of proposed methods through selected real-world applications of these methods [31, 50]. While such examples are encouraging and motivate research on interpretability, they are not suited for quantitative evaluations and comparisons — which are defining for steering future development of even better interpretability tools.

Inspiration for such goal-oriented evaluations can be obtained from recent studies on various applications that require a non-trivial understanding of the inner workings of neural networks:
First, editing specific knowledge in trained networks has been described as an impactful application for large language models [226, 236–238] as well as large vision language models [239, 240]. Knowledge editing could serve as a goal-oriented revelation for future circuit analysis algorithms. However, it might not be meaningful for, e.g., testing per-unit explanations of vision models as those explain the low-level feature extraction of models instead of high-level reasoning. Thus, goal-oriented evaluations of such explanation methods require a modification of the current predominant paradigm of model editing, i.e., knowledge editing. An alternative could be a sub-form of knowledge editing called concept erasure, which focuses on fully removing certain concepts/features from a network's representations [241]. Using model editing as a goal-oriented and quantitative evaluation might require the construction of models that are known to use certain unintended/harmful concepts and features that should be removed through model editing. Such models could be obtained, for example, by training on datasets with intentionally introduced spurious correlations [e.g., 242, 243].

Second, controlling the behavior of a trained network and adapting it to different scenarios without modifying its weights requires a detailed understanding of the inner workings of the network [244]. This steering of a model's behavior has become especially relevant with the development of foundation models as a resource-efficient way of improving, e.g., LLMs' behavior [31, 244–246]. Using this task as a goal-oriented and quantitative interpretability evaluation will require careful construction of the models such that a known optimal (ground-truth) solution exists. This goes beyond existing work that uses qualitative examples [31] or quantitative comparisons with no known ground-truth [246].

Third, understanding how a network internally works can potentially be leveraged to increase a network's safety. With an increased understanding of LLMs, the first examples of safety-critical applications arrived, such as controlling a network's ability to deceive, manipulate, or seek power [31]. Furthermore, interpretability tools have been proposed to detect backdoors, i.e., patterns triggering unexpected and potentially adversarial behavior, in vision models [52, 234]. Ideally, interpretability should enable machine learning engineers to detect safety-critical behavior of their newly trained models before deployment. Thus, such tasks might provide insights into how well we understand a network, but also requires constructing a quantitative benchmark around this task. Due to the complexity of the underlying topic, this might prove a difficult task. The next section adds to this thought of safety-inspired evaluations by presenting a different perspective.

5.5 Understanding Models: Moving from Interpretability to Behavioral Analysis

Do we need to change how we analyze and examine models as they become more capable and intelligent? All of the interpretability approaches investigated in this thesis and potential

future directions described in this chapter share a common assumption akin to reductionism: Namely, that one can break down the complex behavior of a neural network, or generally speaking, a machine learning system, and understand its overall behavior by understanding more and more of its building blocks. When neural networks were small, achieving this was a manageable effort. But what happens if we have ever more of these building blocks? While some work on scaling up existing paradigms exists [31, 225], and potential future directions have been outlined in this chapter, too, it is not clear yet that these approaches remain feasible. Especially when considering the exponential growth neural networks have experienced in recent years regarding their size [247]. As the community strives to build machines showing or surpassing human-like intelligence, we must find tools to analyze such models properly.

As models continue to grow in scale, we may need to develop new analytical tools that can keep pace with their complexity. However, we do not necessarily need to reinvent the wheel. Instead, we could draw inspiration from an unexpected source: human psychology. If models become more human-like and we have human-like ways to interact with them, e.g., through natural language for LLMs and VLMs or physical interactions for embodied AI, why do we not analyze them like we do humans? Approaches for understanding neural networks, e.g., in the mechanistic interpretability community, are often motivated by drawing an analogy to neuroscience that tries to understand human brains (see Chapter 1); now, the time might have come to find inspiration in another field trying to understand the human mind. Psychology and psychotherapy have produced a large body of theories [248, 249] and diagnostic tools [250–255] for understanding and classifying the behavior of humans. This also extends to detecting unusual and potentially safety-critical character traits and behavior [254, 255]. While such an approach would not enable a mechanistic understanding of models, it might grant insights into their overall behavior. Thus, I argue that human-like machine learning systems can be (partially) evaluated through such psychological approaches [256]. While the conceptual approaches of biology and neuroscience have inspired mechanistic interpretability [51], psychology and psychotherapy could inspire a new subfield of behavioral research. Such a psychology-inspired evaluation of models needs to be executed carefully. Existing diagnostic tests cannot be used as they are but should be seen as a source of inspiration for creating new tests to ensure that models have not been trained on them. While neural networks have been evaluated extensively and for a long time regarding their hard skills, such as their task-solving ability or their speed [e.g., 257–262], only recently a few studies started to focus on their soft skills, such as empathy or situational awareness [263–266]. I hypothesize that by evaluating the emotional intelligence of machine learning systems better through existing psychological tests, we can reveal behavioral patterns that can become safety-critical [267] or suggest the system's unreliability in novel scenarios.

Bibliography

[1] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5, 1943. Cited on page 11.

[2] OpenAI. GPT-4 Technical Report. *arXiv preprint*, 2303.08774, 2023. URL https://arxiv.org/abs/2303.08774. Cited on page 11.

[3] Gemini Team. Gemini: A Family of Highly Capable Multimodal Models. *arXiv preprint*, 2312.11805, 2023. URL https://arxiv.org/abs/2312.11805. Cited on page 11.

[4] Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro, and Yi Zhang. Sparks of Artificial General Intelligence: Early experiments with GPT-4. *arXiv preprint*, 2303.12712, 2023. URL https://arxiv.org/abs/2303.12712. Cited on page 11.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*. IEEE, 2016. Cited on pages 11 and 43.

[6] Nils J. Nilsson. *The Quest for Artificial Intelligence*. Cambridge University Press, 2009. ISBN 978-05-11819-34-6. Cited on pages 11 and 20.

[7] A. L. Samuel. Some Studies in Machine Learning Using the Game of Checkers. *IBM Journal of Research and Development*, 3(3), 1959. Cited on pages 11 and 20.

[8] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson, 2016. ISBN 978-1-292-40113-3. Cited on page 11.

[9] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006. ISBN 978-0-387-31073-2. Cited on page 11.

[10] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. ISBN 9780262035613. Cited on pages 11 and 20.

[11] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 1958. Cited on pages 11 and 15.

[12] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint*, 1702.08608, 2017. URL https://arxiv.org/abs/1702.08608. Cited on pages 11, 14, 16, 17, 30, and 59.

[13] Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8), 2019. Cited on pages 11 and 14.

[14] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense Passage Retrieval for Open-Domain Question Answering. *arXiv preprint*, 2004.04906, 2020. URL https://arxiv.org/abs/2004.04906. Cited on page 11.

[15] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8), 2013. Cited on pages 11, 20, 21, and 22.

[16] Council of European Union. Regulation (eu) 2016/679 of the european parliament and of the council on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation). *Official Journal of the European Union*, L119/59, 2016. Cited on page 12.

[17] Council of European Union. Proposal for a regulation of the european parliament and of the council laying down harmonised rules on artificial intelligence (artificial intelligence act) and amending certain union legislative acts, 2021. URL https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206. Cited on page 12.

[18] Bryce Goodman and Seth Flaxman. European Union regulations on algorithmic decision-making and a "right to explanation". *arXiv preprint*, 1606.08813, 2016. URL https://arxiv.org/abs/1606.08813. Cited on page 12.

[19] Stanley Finger and Stanley Finger. *Origins of Neuroscience: A History of Explorations into Brain Function*. Oxford University Press, Oxford, New York, 2001. ISBN 978-0-19-514694-3. Cited on page 13.

[20] Glen E. Getz. *Applied Biological Psychology*. Springer Publishing Company, 2023. ISBN 978-0-8261-0922-4. Cited on page 13.

[21] Enrico Crivellato and Domenico Ribatti. Soul, mind, brain: Greek philosophy and the birth of neuroscience. *Brain Research Bulletin*, 71(4), 2007. Cited on page 13.

[22] Martin Schrimpf, Jonas Kubilius, Ha Hong, Najib J. Majaj, Rishi Rajalingham, Elias B. Issa, Kohitij Kar, Pouya Bashivan, Jonathan Prescott-Roy, Franziska Geiger, Kailyn Schmidt, Daniel L. K. Yamins, and James J. DiCarlo. Brain-score: Which artificial neural network for object recognition is most brain-like? *bioRxiv preprint*, 2018. URL https://www.biorxiv.org/content/10.1101/407007v2. Cited on page 14.

[23] Martin Schrimpf, Jonas Kubilius, Michael J. Lee, N. Apurva Ratan Murty, Robert Ajemian, and James J. DiCarlo. Integrative benchmarking to advance neurally mechanistic models of human intelligence. *Neuron*, 2020. URL https://www.cell.com/neuron/fulltext/S0896-6273(20)30605-X. Cited on page 14.

[24] Anna A. Ivanova, John Hewitt, and Noga Zaslavsky. Probing artificial neural networks: Insights from neuroscience. *arXiv preprint*, 2104.08197, 2021. URL https://arxiv.org/abs/2104.08197. Cited on page 14.

[25] Daniel L. K. Yamins, Ha Hong, Charles F. Cadieu, Ethan A. Solomon, Darren Seibert, and James J. DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111 (23), 2014. Cited on page 14.

[26] Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1, 2015. Cited on pages 14, 15, 17, and 28.

[27] Chengxu Zhuang, Siming Yan, Aran Nayebi, Martin Schrimpf, Michael C. Frank, James J. DiCarlo, and Daniel L. K. Yamins. Unsupervised neural network models of the ventral visual stream. *Proceedings of the National Academy of Sciences*, 118(3), 2021. Cited on page 14.

[28] Johannes Mehrer, Courtney J. Spoerer, Emer C. Jones, Nikolaus Kriegeskorte, and Tim C. Kietzmann. An ecologically motivated image dataset for deep learning yields better models of human vision. *Proceedings of the National Academy of Sciences*, 118 (8), 2021. Cited on page 14.

[29] Pawel A. Pierzchlewicz, Konstantin Willeke, Arne Nix, Pavithra Elumalai, Kelli Restivo, Tori Shinn, Cate Nealley, Gabrielle Rodriguez, Saumil S. Patel, Katrin Franke, Andreas S. Tolias, and Fabian H. Sinz. Energy guided diffusion for generating neurally exciting images. In *NeurIPS*, 2023. Cited on page 14.

[30] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html. Cited on pages 14 and 59.

[31] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L. Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *Transformer Circuits Thread*, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html. Cited on pages 14, 16, 57, 60, and 61.

[32] Aniek F. Markus, Jan A. Kors, and Peter R. Rijnbeek. The role of explainability in creating trustworthy artificial intelligence for health care: A comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of Biomedical Informatics*, 113, 2021. Cited on page 14.

[33] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael A. Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, 2018. Cited on page 14.

[34] Tilman Räuker, Anson Ho, Stephen Casper, and Dylan Hadfield-Menell. Toward Transparent AI: A Survey on Interpreting the Inner Structures of Deep Neural Networks. *arXiv preprint*, 2207.13243, 2022. URL https://arxiv.org/abs/2207.13243. Cited on pages 14, 22, and 59.

[35] Mara Graziani, Lidia Dutkiewicz, Davide Calvaresi, José Pereira Amorim, Katerina Yordanova, Mor Vered, Rahul Nair, Pedro Henriques Abreu, Tobias Blanke, Valeria Pulignano, John O. Prior, Lode Lauwaert, Wessel Reijers, Adrien Depeursinge, Vincent Andrearczyk, and Henning Müller. A global taxonomy of interpretable AI: Unifying the terminology for the technical and social sciences. *Artificial Intelligence Review*, 56 (4), 2023. Cited on page 14.

[36] Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. Learning to generate reviews and discovering sentiment. *arXiv preprint*, 1704.01444, 2017. URL https://arxiv.org/abs/1704.01444. Cited on page 14.

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*. PMLR, 2021. Cited on pages 14, 37, and 38.

[38] Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. A Survey of the State of Explainable AI for Natural Language Processing. *arXiv preprint*, 2010.00711, 2020. URL https://arxiv.org/abs/2010.00711. Cited on page 14.

[39] Andreas Madsen, Sarath Chandar, and Siva Reddy. Are self-explanations from Large Language Models faithful? *arXiv preprint*, 2401.07927, 2024. URL https://arxiv.org/abs/2401.07927. Cited on page 14.

[40] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016. Cited on page 14.

[41] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*. IEEE, 2017. Cited on page 14.

[42] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *ICML*. PMLR, 2018. Cited on page 14.

[43] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014. Cited on page 14.

[44] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7), 2015. Cited on page 14.

[45] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint*, 1706.03825, 2017. URL https://arxiv.org/abs/1706.03825. Cited on page 14.

[46] Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint*, 1702.04595, 2017. URL https://arxiv.org/abs/1702.04595. Cited on page 14.

[47] Chris Olah. Mechanistic interpretability, variables, and the importance of interpretable bases. https://www.transformer-circuits.pub/2022/mech-interp-essay, 2022. Accessed on 01.06.2024. Cited on page 14.

[48] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11), 2017. Cited on pages 14, 15, 16, 17, 18, 29, 30, 31, 37, 53, and 58.

[49] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. URL https://distill.pub/2020/circuits/zoom-in. Cited on pages 14, 15, 16, and 17.

[50] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/index.html. Cited on pages 14, 16, 54, 58, and 60.

[51] Leonard Bereska and Efstratios Gavves. Mechanistic Interpretability for AI Safety – A Review. *arXiv preprint*, 2404.14082, 2024. URL https://arxiv.org/abs/2404.14082. Cited on pages 14 and 61.

[52] Stephen Casper, Tong Bu, Yuxiao Li, Jiawei Li, Kevin Zhang, Kaivalya Hariharan, and Dylan Hadfield-Menell. Red teaming deep neural networks with feature synthesis tools. In *NeurIPS*, 2023. Cited on pages 14, 15, and 60.

[53] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*. IEEE, 2017. Cited on page 15.

[54] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Revisiting the importance of individual units in cnns via ablation. *arXiv preprint*, 1806.02891, 2018. Cited on page 15.

[55] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48), 2020. Cited on page 15.

[56] Ari S. Morcos, David G. T. Barrett, Neil C. Rabinowitz, and Matthew M. Botvinick. On the importance of single directions for generalization. In *ICLR*, 2018. Cited on page 15.

[57] Nick Cammarata, Gabriel Goh, Shan Carter, Ludwig Schubert, Michael Petrov, and Chris Olah. Curve detectors. *Distill*, 2020. URL https://distill.pub/2020/circuits/curve-detectors. Cited on page 15.

[58] Nick Cammarata, Shan Carter, Gabriel Goh, Chris Olah, Michael Petrov, Ludwig Schubert, Chelsea Voss, Ben Egan, and Swee Kiat Lim. Thread: Circuits. *Distill*, 5(3), 2020. Cited on pages 15 and 28.

[59] Kunihiko Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36(4), 1980. Cited on page 15.

[60] Yann LeCun, Bernhard E. Boser, John S. Denker, Donnie Henderson, Richard E. Howard, Wayne E. Hubbard, and Lawrence D. Jackel. Handwritten digit recognition with a back-propagation network. In *NeurIPS*. Morgan Kaufmann, 1989. Cited on page 15.

[61] Mara Graziani, Laura O'Mahony, An-phi Nguyen, Henning Müller, and Vincent Andrearczyk. Uncovering Unique Concept Vectors through Latent Space Decomposition. *Transactions on Machine Learning Research*, 2023. Cited on page 15.

[62] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *CVPR*. IEEE, 2015. Cited on page 15.

[63] Anh Mai Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *CVPR*. IEEE, 2015. Cited on pages 15, 28, and 53.

[64] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks. URL https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html, 2015. Accessed on 01.06.2024. Cited on page 15.

[65] Anh Mai Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *NeurIPS*, 2016. Cited on pages 15 and 18.

[66] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *CVPR*. IEEE, 2017. Cited on pages 15, 17, 18, 28, 37, and 58.

[67] Anh Nguyen, Jason Yosinski, and Jeff Clune. Understanding neural networks via feature visualization: A survey. In *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer, 2019. Cited on pages 15, 28, 37, and 53.

[68] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. In *Deep Learning Workshop, ICML*, 2015. Cited on pages 15 and 18.

[69] Dumitru Erhan, Y. Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *Technical Report, Univeristé de Montréal*, 2009. Cited on page 15.

[70] Thomas Fel, Thibaut Boissin, Victor Boutin, Agustin Picard, Paul Novello, Julien Colin, Drew Linsley, Tom Rousseau, Rémi Cadène, Laurent Gardes, and Thomas Serre. Unlocking Feature Visualization for Deeper Networks with MAgnitude Constrained Optimization. *arXiv preprint*, 2306.06805, 2023. URL https://arxiv.org/abs/2306.06805. Cited on pages 15, 17, 33, and 53.

[71] Evan Hernandez, Sarah Schwettmann, David Bau, Teona Bagashvili, Antonio Torralba, and Jacob Andreas. Natural language descriptions of deep visual features. In *ICLR*, 2022. Cited on pages 15, 53, and 57.

[72] Neha Kalibhat, Shweta Bhardwaj, C. Bayan Bruss, Hamed Firooz, Maziar Sanjabi, and Soheil Feizi. Identifying interpretable subspaces in image representations. In *ICML*. PMLR, 2023. Cited on pages 15 and 57.

[73] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. An overview of early vision in inceptionv1. *Distill*, 2020. URL https://distill.pub/2020/circuits/early-vision. Cited on page 15.

[74] Chris Olah, Nick Cammarata, Chelsea Voss, Ludwig Schubert, and Gabriel Goh. Naturally occurring equivariance in neural networks. *Distill*, 5(12), 2020. URL https://distill.pub/2020/circuits/equivariance/. Cited on page 15.

[75] Ludwig Schubert, Chelsea Voss, Nick Cammarata, Gabriel Goh, and Chris Olah. High-low frequency detectors. *Distill*, 6(1), 2021. URL https://distill.pub/2020/circuits/frequency-edges/. Cited on pages 15 and 31.

[76] Chelsea Voss, Nick Cammarata, Gabriel Goh, Michael Petrov, Ludwig Schubert, Ben Egan, Swee Kiat Lim, and Chris Olah. Visualizing weights. *Distill*, 6(2), 2021. URL https://distill.pub/2020/circuits/visualizing-weights/. Cited on page 15.

[77] Chelsea Voss, Gabriel Goh, Nick Cammarata, Michael Petrov, Ludwig Schubert, and Chris Olah. Branch specialization. *Distill*, 6(4), 2021. URL https://distill.pub/2020/circuits/branch-specialization/. Cited on page 15.

[78] Michael Petrov, Chelsea Voss, Ludwig Schubert, Nick Cammarata, Gabriel Goh, and Chris Olah. Weight banding. *Distill*, 6(4), 2021. URL https://distill.pub/2020/circuits/weight-banding/. Cited on page 15.

[79] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*. IEEE, 2015. Cited on pages 15 and 29.

[80] Santiago A. Cadena, Marissa A. Weis, Leon A. Gatys, Matthias Bethge, and Alexander S. Ecker. Diverse feature visualizations reveal invariances in early layers of deep neural networks. In *ECCV*. Springer, 2018. Cited on page 15.

[81] Nicolas Gonthier, Yann Gousseau, and Saïd Ladjal. An analysis of the transfer learning of convolutional neural networks for artistic images. *arXiv preprint*, 2011.02727, 2020. URL https://arxiv.org/abs/2011.02727. Cited on page 15.

[82] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 6(3), 2021. Cited on pages 15 and 28.

[83] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint*, 1506.06579, 2015. URL https://arxiv.org/abs/1506.06579. Cited on pages 15, 28, 29, and 37.

[84] Zhiwei Ding, Dat T. Tran, Kayla Ponder, Erick Cobos, Zhuokun Ding, Paul G. Fahey, Eric Wang, Taliah Muhammad, Jiakun Fu, Santiago A. Cadena, Stelios Papadopoulos, Saumil Patel, Katrin Franke, Jacob Reimer, Fabian H. Sinz, Alexander S. Ecker, Xaq Pitkow, and Andreas S. Tolias. Bipartite invariance in mouse primary visual cortex. *bioRxiv preprint*, 2023. URL https://www.biorxiv.org/content/10.1101/2023.03.15.532836v1. Cited on page 15.

[85] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*. IEEE, 2017. Cited on page 15.

[86] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022.

URL https://transformer-circuits.pub/2022/toy_model/index.html. Cited on pages 15 and 16.

[87] Anh Nguyen, Jason Yosinski, and Jeff Clune. Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *arXiv preprint*, 1602.03616, 2016. URL https://arxiv.org/abs/1602.03616. Cited on pages 16 and 29.

[88] Thomas Fel, Victor Boutin, Louis Béthune, Rémi Cadène, Mazda Moayeri, Léo Andéol, Mathieu Chalvidal, and Thomas Serre. A holistic approach to unifying automatic concept extraction and concept importance estimation. In *NeurIPS*, 2023. Cited on page 16.

[89] Adam S. Jermyn, Nicholas Schiefer, and Evan Hubinger. Engineering Monosemanticity in Toy Models. *arXiv preprint*, 2211.09169, 2022. URL https://arxiv.org/abs/2211.09169. Cited on page 16.

[90] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse Autoencoders Find Highly Interpretable Features in Language Models. *arXiv preprint*, 2309.08600, 2023. URL https://arxiv.org/abs/2309.08600. Cited on page 16.

[91] Laura O'Mahony, Vincent Andrearczyk, Henning Müller, and Mara Graziani. Disentangling Neuron Representations With Concept Vectors. In *CVPR*. IEEE, 2023. Cited on page 16.

[92] David Klindt, Sophia Sanborn, Francisco Acosta, Frédéric Poitevin, and Nina Miolane. Identifying interpretable visual features in artificial and biological neural systems. *arXiv preprint*, 2310.11431, 2023. URL https://arxiv.org/abs/2310.11431. Cited on page 16.

[93] Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. Improving Dictionary Learning with Gated Sparse Autoencoders. *arXiv preprint*, 2404.16014, 2024. URL https://arxiv.org/abs/2404.16014. Cited on pages 16, 54, 57, and 58.

[94] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267, 2019. Cited on page 16.

[95] Diogo Vieira Carvalho, Eduardo Marques Pereira, and Jaime S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 2019. Cited on page 16.

[96] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *NeurIPS*, 2018. Cited on page 16.

[97] Matthew L. Leavitt and Ari Morcos. Towards falsifiable interpretability research. *arXiv preprint*, 2010.12016, 2020. URL https://arxiv.org/abs/2010.12016. Cited on pages 16 and 53.

[98] Marko Robnik-Šikonja and Marko Bohanec. Perturbation-based explanations of prediction models. *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*, 2018. Cited on page 16.

[99] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion Attacks against Machine Learning at Test Time. In *Machine Learning and Knowledge Discovery in Databases*, Berlin, Heidelberg, 2013. Springer. Cited on page 16.

[100] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint*, 1312.6199, 2013. URL https://arxiv.org/abs/1312.6199. Cited on page 16.

[101] Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On Evaluating Adversarial Robustness. *arXiv preprint*, 1902.06705, 2019. URL https://arxiv.org/abs/1902.06705. Cited on pages 16 and 17.

[102] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 3 (3), 2018. Cited on pages 17, 29, 36, and 58.

[103] Robert Geirhos, Kantharaju Narayanappa, Benjamin Mitzkus, Tizian Thieringer, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Partial success in closing the gap between human and machine vision. In *NeurIPS*, 2021. Cited on pages 18, 37, and 38.

[104] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme Ruiz, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd van Steenkiste, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Collier, Alexey A. Gritsenko, Vighnesh Birodkar, Cristina Nader Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetic, Dustin Tran, Thomas Kipf, Mario Lucic, Xiaohua Zhai, Daniel Keysers, Jeremiah J. Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters. In *ICML*. PMLR, 2023. Cited on pages 18 and 37.

[105] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *ICML*. PMLR, 2019. Cited on page 18.

[106] Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. URL https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html, 2023. Cited on pages 19, 41, and 58.

[107] Jing Huang, Atticus Geiger, Karel D'Oosterlinck, Zhengxuan Wu, and Christopher Potts. Rigorously assessing natural language explanations of neurons. *arXiv preprint*, 2309.10312, 2023. URL https://arxiv.org/abs/2309.10312. Cited on pages 19, 41, 57, and 58.

[108] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*. IEEE, 2018. Cited on pages 19 and 41.

[109] Jonathan Dinu, Jeffrey Bigham, and J. Zico Kolter. Challenging common interpretability assumptions in feature attribution explanations. *arXiv preprint*, 2012.02748, 2020. URL https://arxiv.org/abs/2012.02748. Cited on pages 19 and 41.

[110] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. DreamSim: Learning New Dimensions of Human Visual Similarity using Synthetic Data. *arXiv preprint*, 2306.09344, 2023. URL https://arxiv.org/abs/2306.09344. Cited on pages 19 and 41.

[111] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4), 1943. Cited on page 20.

[112] Donald Olding Hebb. *The organization of behavior: A neuropsychological theory*. Psychology press, 2005. Cited on page 20.

[113] A. M. Turing. Computing Machinery and Intelligence. *Mind*, LIX(236), 1950. Cited on page 20.

[114] Greg Yang, Edward J. Hu, Igor Babuschkin, Szymon Sidor, David Farhi, Jakub Pachocki, Xiaodong Liu, Weizhu Chen, and Jianfeng Gao. Tensor Programs V: Tuning Large Neural Networks via Zero-Shot Hyperparameter Transfer. In *NeurIPS 2021*, 2022. Cited on page 20.

[115] Jeremy Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. In *ICML*. PMLR, 2019. Cited on page 20.

[116] Grégoire Mesnil, Yann Dauphin, Xavier Glorot, Salah Rifai, Yoshua Bengio, Ian Goodfellow, Erick Lavoie, Xavier Muller, Guillaume Desjardins, David Warde-Farley, Pascal Vincent, Aaron Courville, and James Bergstra. Unsupervised and Transfer Learning Challenge: A Deep Learning Approach. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 2012. Cited on page 20.

[117] O. Chapelle, B. Scholkopf, and A. Zien, Eds. Semi-Supervised Learning. *IEEE Transactions on Neural Networks*, 20(3), 2009. Cited on page 20.

[118] Yoshua Bengio. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, 2(1), 2009. Cited on page 20.

[119] Rajat Raina, Alexis Battle, Honglak Lee, Benjamin Packer, and Andrew Y. Ng. Self-taught learning: Transfer learning from unlabeled data. In *ICML*, New York, NY, USA, 2007. PMLR. ISBN 978-1-59593-793-3. Cited on page 20.

[120] Sebastian Thrun. Is Learning The n-th Thing Any Easier Than Learning The First? In *Advances in Neural Information Processing Systems*, volume 8. MIT Press, 1995. Cited on page 20.

[121] Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Gregoire Mialon, Yuandong Tian, Avi Schwarzschild, Andrew Gordon Wilson, Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum. A Cookbook of Self-Supervised Learning. *arXiv preprint*, 2304.12210, 2023. URL https://arxiv.org/abs/2304.12210. Cited on pages 20, 22, and 45.

[122] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. Cited on page 20.

[123] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*. IEEE, 2005. Cited on page 20.

[124] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint*, 1807.03748, 2018. URL https://arxiv.org/abs/1807.03748. Cited on pages 20, 21, 22, 46, and 54.

[125] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *ICML*. PMLR, 2020. Cited on pages 20, 21, 22, 45, 46, and 47.

[126] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge J. Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *CVPR*. IEEE, 2021. Cited on page 20.

[127] Ishan Dave, Rohit Gupta, Mamshad Nayeem Rizve, and Mubarak Shah. TCLR: Temporal Contrastive Learning for Video Representation. *arXiv preprint*, 2101.07974, 2021. URL https://arxiv.org/abs/2101.07974. Cited on page 20.

[128] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*. IEEE, 2020. Cited on pages 20 and 22.

[129] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved Baselines with Momentum Contrastive Learning. *arXiv preprint*, 2003.04297, 2020. URL https://arxiv.org/abs/2003.04297. Cited on page 20.

[130] Ting Chen, Calvin Luo, and Lala Li. Intriguing properties of contrastive losses. In *NeurIPS*, 2021. Cited on page 21.

[131] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. In *NeurIPS*, 2020. Cited on pages 21, 22, and 45.

[132] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, 2020. Cited on page 21.

[133] Chun-Hsiao Yeh, Cheng-Yao Hong, Yen-Chi Hsu, Tyng-Luh Liu, Yubei Chen, and Yann LeCun. Decoupled contrastive learning. In *ECCV*. Springer, 2022. Cited on page 21.

[134] Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. On the Binding Problem in Artificial Neural Networks. *arXiv preprint*, 2012.05208, 2020. URL https://arxiv.org/abs/2012.05208. Cited on pages 21 and 49.

[135] Jinyang Yuan, Tonglin Chen, Bin Li, and Xiangyang Xue. Compositional Scene Representation Learning via Reconstruction: A Survey. *arXiv preprint*, 2202.07135, 2022. URL https://arxiv.org/abs/2202.07135. Cited on page 21.

[136] Klaus Greff, Antti Rasmus, Mathias Berglund, Tele Hotloo Hao, Harri Valpola, and Jürgen Schmidhuber. Tagger: Deep unsupervised perceptual grouping. In *NeurIPS*, 2016. Cited on page 21.

[137] S. M. Ali Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, Koray Kavukcuoglu, and Geoffrey E. Hinton. Attend, Infer, Repeat: Fast Scene Understanding with Generative Models. *arXiv preprint*, 1603.08575, 2016. URL https://arxiv.org/abs/1603.08575. Cited on page 21.

[138] Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Caglar Gulcehre, Francis Song, Andrew Ballard, Justin Gilmer, George Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. Relational inductive biases, deep learning, and graph networks. *arXiv preprint*, 1806.01261, 2018. URL https://arxiv.org/abs/1806.01261. Cited on page 21.

[139] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. In *NeurIPS*, 2020. Cited on pages 21, 49, 51, and 55.

[140] Thomas N. Kipf, Elise van der Pol, and Max Welling. Contrastive learning of structured world models. In *ICLR*, 2020. Cited on page 21.

[141] Andrea Dittadi, Samuele S. Papa, Michele De Vita, Bernhard Schölkopf, Ole Winther, and Francesco Locatello. Generalization and robustness implications in object-centric learning. In *ICML*. PMLR, 2022. Cited on page 21.

[142] Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. Effective Approaches to Attention-based Neural Machine Translation. *arXiv preprint*, 1508.04025, 2015. URL https://arxiv.org/abs/1508.04025. Cited on page 21.

[143] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015. Cited on page 21.

[144] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *arXiv preprint*, 1706.03762, 2017. URL https://arxiv.org/abs/1706.03762. Cited on page 21.

[145] Klaus Greff, Rupesh Kumar Srivastava, and Jürgen Schmidhuber. Binding via Reconstruction Clustering. *arXiv preprint*, 1511.06418, 2015. URL https://arxiv.org/abs/1511.06418. Cited on pages 21 and 23.

[146] Sjoerd van Steenkiste, Michael Chang, Klaus Greff, and Jürgen Schmidhuber. Relational Neural Expectation Maximization: Unsupervised Discovery of Objects and their Interactions. *arXiv preprint*, 1802.10353, 2018. URL https://arxiv.org/abs/1802.10353. Cited on page 21.

[147] Christopher P. Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. MONet: Unsupervised Scene Decomposition and Representation. *arXiv preprint*, 1901.11390, 2019. URL https://arxiv.org/abs/1901.11390. Cited on page 21.

[148] Klaus Greff, Raphaël Lopez Kaufman, Rishabh Kabra, Nick Watters, Chris Burgess, Daniel Zoran, Loic Matthey, Matthew M. Botvinick, and Alexander Lerchner. Multi-object representation learning with iterative variational inference. In *ICML*. PMLR, 2019. Cited on page 21.

[149] Martin Engelcke, Adam R. Kosiorek, Oiwi Parker Jones, and Ingmar Posner. GENESIS: Generative Scene Inference and Sampling with Object-Centric Latent Representations. *arXiv preprint*, 1907.13052, 2019. URL https://arxiv.org/abs/1907.13052. Cited on page 21.

[150] Rishabh Kabra, Daniel Zoran, Goker Erdogan, Loic Matthey, Antonia Creswell, Matthew Botvinick, Alexander Lerchner, and Christopher P. Burgess. SIMONe: View-Invariant, Temporally-Abstracted Object Representations via Unsupervised Video Decomposition. *arXiv preprint*, 2106.03849, 2021. URL https://arxiv.org/abs/2106.03849. Cited on page 21.

[151] Rishi Veerapaneni, John D. Co-Reyes, Michael Chang, Michael Janner, Chelsea Finn, Jiajun Wu, Joshua B. Tenenbaum, and Sergey Levine. Entity Abstraction in Visual Model-Based Reinforcement Learning. *arXiv preprint*, 1910.12827, 2019. URL https://arxiv.org/abs/1910.12827. Cited on page 21.

[152] Gautam Singh, Yi-Fu Wu, and Sungjin Ahn. Simple unsupervised object-centric learning for complex and naturalistic videos. In *NeurIPS*, 2022. Cited on page 21.

[153] Thomas Kipf, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Austin Stone, Sara Sabour, Georg Heigold, Rico Jonschkowski, Alexey Dosovitskiy, and Klaus Greff. Conditional object-centric learning from video. In *ICLR*, 2022. Cited on pages 21 and 23.

[154] E. J. Green. A Theory of Perceptual Objects. *Philosophy and Phenomenological Research*, 99(3), 2019. Cited on page 21.

[155] Amy C. Gross and Eric J. Fox. Relational frame theory: An overview of the controversy. *The Analysis of Verbal Behavior*, 25(1), 2009. Cited on page 21.

[156] Stellan Ohlsson. Restructuring revisited: I. Summary and critique of the Gestalt theory of problem solving. *Scandinavian Journal of Psychology*, 25(1), 1984. Cited on page 21.

[157] Johan Wagemans, James H. Elder, Michael Kubovy, Stephen E. Palmer, Mary A. Peterson, Manish Singh, and Rüdiger von der Heydt. A century of Gestalt psychology in visual perception: I. Perceptual grouping and figure–ground organization. *Psychological Bulletin*, 138(6), 2012. Cited on page 21.

[158] K. Koffka. *Principles of Gestalt Psychology*. Principles of Gestalt Psychology. Harcourt, Brace, Oxford, England, 1935. Cited on page 21.

[159] Elizabeth S. Spelke. Principles of object perception. *Cognitive Science*, 14(1), 1990. Cited on page 21.

[160] Lukas Schott, Julius von Kügelgen, Frederik Träuble, Peter Vincent Gehler, Chris Russell, Matthias Bethge, Bernhard Schölkopf, Francesco Locatello, and Wieland Brendel. Visual representation learning does not generalize strongly within the same domain. In *ICLR*, 2022. Cited on page 22.

[161] Sébastien Lachapelle, Tristan Deleu, Divyat Mahajan, Ioannis Mitliagkas, Yoshua Bengio, Simon Lacoste-Julien, and Quentin Bertrand. Synergies between Disentanglement and Sparsity: Generalization and Identifiability in Multi-Task Learning. *arXiv preprint*, 2211.14666, 2022. URL https://arxiv.org/abs/2211.14666. Cited on page 22.

[162] Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017. Cited on page 22.

[163] Tian Qi Chen, Xuechen Li, Roger B. Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. In *NeurIPS*, 2018. Cited on page 22.

[164] Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *ICML*. PMLR, 2018. Cited on page 22.

[165] Diane Bouchacourt, Ryota Tomioka, and Sebastian Nowozin. Multi-Level Variational Autoencoder: Learning Disentangled Representations From Grouped Observations. *AAAI*, 32(1), 2018. Cited on page 22.

[166] Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. Variational inference of disentangled latent concepts from unlabeled observations. In *ICLR*, 2018. Cited on page 22.

[167] Erich L. Lehmann and George Casella. *Theory of point estimation*. Springer Science & Business Media, 2006. ISBN 978-0-387-22728-3. Cited on page 22.

[168] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *ICML*. PMLR, 2020. Cited on page 22.

[169] David A. Klindt, Lukas Schott, Yash Sharma, Ivan Ustyuzhaninov, Wieland Brendel, Matthias Bethge, and Dylan M. Paiton. Towards nonlinear disentanglement in natural data with temporal sparse coding. In *ICLR*, 2021. Cited on page 22.

[170] Kartik Ahuja, Divyat Mahajan, Vasilis Syrgkanis, and Ioannis Mitliagkas. Towards efficient representation identification in supervised learning. In *Proceedings of the First Conference on Causal Learning and Reasoning*. PMLR, 2022. Cited on page 22.

[171] Aapo Hyvärinen and Petteri Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3), 1999. Cited on pages 22 and 46.

[172] Aapo Hyvärinen, Juha Karhunen, and Erkki Oja. *Independent Component Analysis*. Wiley Interscience, 2001. ISBN 978-04-71405-40-5. Cited on page 22.

[173] Ilyes Khemakhem, Diederik P. Kingma, Ricardo Pio Monti, and Aapo Hyvärinen. Variational autoencoders and nonlinear ICA: A unifying framework. In *AISTATS*. PMLR, 2020. Cited on page 22.

[174] Ilyes Khemakhem, Ricardo Pio Monti, Diederik P. Kingma, and Aapo Hyvärinen. Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ICA. In *NeurIPS*, 2020. Cited on page 22.

[175] Aapo Hyvärinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ICA. In *NeurIPS*, 2016. Cited on page 22.

[176] Aapo Hyvärinen and Hiroshi Morioka. Nonlinear ICA of temporally dependent stationary sources. In *AISTATS*. PMLR, 2017. Cited on page 22.

[177] Aapo Hyvärinen, Hiroaki Sasaki, and Richard E. Turner. Nonlinear ICA using auxiliary variables and generalized contrastive learning. In *AISTATS*. PMLR, 2019. Cited on page 22.

[178] Ralph Linsker. Self-organization in a perceptual network. *Computer*, 21(3), 1988. Cited on page 22.

[179] Philip Bachman, R. Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, 2019. Cited on page 22.

[180] R. Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Philip Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *ICLR*, 2019. Cited on page 22.

[181] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *arXiv preprint*, 2005.10243, 2020. URL https://arxiv.org/abs/2005.10243. Cited on page 22.

[182] Michael Tschannen, Josip Djolonga, Paul K. Rubenstein, Sylvain Gelly, and Mario Lucic. On mutual information maximization for representation learning. In *ICLR*, 2020. Cited on page 22.

[183] Nikunj Saunshi, Orestis Plevrakis, Sanjeev Arora, Mikhail Khodak, and Hrishikesh Khandeparkar. A theoretical analysis of contrastive unsupervised representation learning. In *ICML*. PMLR, 2019. Cited on pages 22, 45, and 54.

[184] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*. IEEE, 2018. Cited on page 22.

[185] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint*, 1906.05849, 2019. URL https://arxiv.org/abs/1906.05849. Cited on page 22.

[186] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. *arXiv preprint*, 2010.04592, 2020. URL https://arxiv.org/abs/2010.04592. Cited on page 22.

[187] Ching-Yao Chuang, Joshua Robinson, Yen-Chen Lin, Antonio Torralba, and Stefanie Jegelka. Debiased contrastive learning. In *NeurIPS*, 2020. Cited on page 22.

[188] Gamaleldin F. Elsayed, Aravindh Mahendran, Sjoerd van Steenkiste, Klaus Greff, Michael C. Mozer, and Thomas Kipf. SAVi++: Towards End-to-End Object-Centric Learning from Real-World Videos. *arXiv preprint*, 2206.07764, 2022. URL https://arxiv.org/abs/2206.07764. Cited on page 23.

[189] Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, and Francesco Locatello. Bridging the gap to real-world object-centric learning. In *ICLR*, 2023. Cited on pages 23, 49, and 55.

[190] Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Object-centric learning for real-world videos by predicting temporal feature similarities. In *NeurIPS*, 2023. Cited on page 23.

[191] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3), 2009. Cited on page 29.

[192] Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. Activation atlas. *Distill*, 2019. https://distill.pub/2019/activation-atlas. Cited on page 29.

[193] Matthias Kümmerer, Thomas S. A. Wallis, Leon A. Gatys, and Matthias Bethge. Understanding low- and high-level contributions to fixation prediction. In *ICCV*. IEEE, 2017. Cited on page 32.

[194] Géraldin Nanfack, Alexander Fulleringer, Jonathan Marty, Michael Eickenberg, and Eugene Belilovsky. Adversarial attacks on the interpretation of neuron activation maximization. In *AAAI*. AAAI Press, 2024. Cited on pages 36 and 54.

[195] Suraj Srinivas and François Fleuret. Full-gradient representation for neural network visualization. In *NeurIPS*, 2019. Cited on page 36.

[196] Blair Bilodeau, Natasha Jaques, Pang Wei Koh, and Been Kim. Impossibility theorems for feature attribution. *Proceedings of the National Academy of Sciences*, 121(2), 2024. Cited on page 36.

[197] Hidde Fokkema, Rianne de Heide, and Tim van Erven. Attribution-based explanations that provide recourse cannot be robust. *arXiv preprint*, 2205.15834, 2022. URL https://arxiv.org/abs/2205.15834. Cited on page 36.

[198] Tessa Han, Suraj Srinivas, and Himabindu Lakkaraju. Which explanation should I choose? A function approximation perspective to characterizing post hoc explanations. In *NeurIPS*, 2022. Cited on page 36.

[199] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3), 2015. Cited on pages 38 and 42.

[200] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: an open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. Cited on page 38.

[201] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. Cited on page 38.

[202] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. Cited on page 38.

[203] Yao-Yuan Yang, Cyrus Rashtchian, Hongyang Zhang, Ruslan Salakhutdinov, and Kamalika Chaudhuri. A closer look at accuracy vs. robustness. In *NeurIPS*, 2020. Cited on page 38.

[204] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. RobustBench: A standardized adversarial robustness benchmark. *arXiv preprint*, 2010.09670, 2020. URL https://arxiv.org/abs/2010.09670. Cited on page 38.

[205] Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. In *NeurIPS*, 2019. Cited on page 38.

[206] Christof Koch. What Is Consciousness? *Nature*, 557(7704), 2018. Cited on page 40.

[207] Ross Wightman. Pytorch image models. URL https://github.com/rwightman/pytorch-image-models, 2019. Cited on page 42.

[208] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*. PMLR, 2020. Cited on pages 45 and 46.

[209] Christian Jutten, Massoud Babaie-Zadeh, and Juha Karhunen. Nonlinear mixtures. *Handbook of Blind Source Separation, Independent Component Analysis and Applications*, 2010. Cited on page 46.

[210] Dominik Janzing and Bernhard Schölkopf. Causal inference using the algorithmic markov condition. *IEEE Transactions on Information Theory*, 56(10), 2010. Cited on page 50.

[211] Christpher P. Burgess, Loic Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matt Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *arXiv preprint*, 1901.11390, 2019. URL https://arxiv.org/abs/1901.11390. Cited on page 51.

[212] Chris Hamblin, Thomas Fel, Srijani Saha, Talia Konkle, and George Alvarez. Feature accentuation: Revealing 'what' features respond to in natural images. *arXiv preprint*, 2402.10039, 2024. URL https://arxiv.org/abs/2402.10039. Cited on page 53.

[213] Nicholas Bai, Rahul A. Iyer, Tuomas Oikarinen, and Tsui-Wei Weng. Describe-and-Dissect: Interpreting Neurons in Vision Networks with Language Models. *arXiv preprint*, 2403.13771, 2024. URL https://arxiv.org/abs/2403.13771. Cited on page 53.

[214] Tuomas Oikarinen and Tsui-Wei Weng. CLIP-Dissect: Automatic Description of Neuron Representations in Deep Vision Networks. *arXiv preprint*, 2204.10965, 2022. URL https://arxiv.org/abs/2204.10965. Cited on pages 53 and 57.

[215] Laura Kopf, Philine Lou Bommer, Anna Hedström, Sebastian Lapuschkin, Marina M.-C. Höhne, and Kirill Bykov. CoSy: Evaluating Textual Explanations of Neurons. *arXiv preprint*, 2405.20331, 2024. URL https://arxiv.org/abs/2405.20331. Cited on page 53.

[216] Dilyara Bareeva, Marina M.-C. Höhne, Alexander Warnecke, Lukas Pirch, Klaus-Robert Müller, Konrad Rieck, and Kirill Bykov. Manipulating Feature Visualizations with Gradient Slingshots. *arXiv preprint*, 2401.06122, 2024. URL https://arxiv.org/abs/2401.06122. Cited on page 54.

[217] Geoffrey Roeder, Luke Metz, and Durk Kingma. On linear identifiability of learned representations. In *ICML*. PMLR, 2021. Cited on page 54.

[218] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Mostofa Ali Patwary, Yang Yang, and Yanqi Zhou. Deep Learning Scaling is Predictable, Empirically. *arXiv preprint*, 1712.00409, 2017. URL https://arxiv.org/abs/1712.00409. Cited on page 55.

[219] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan

Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal Large Language Models. *arXiv preprint*, 2203.15556, 2022. URL https://arxiv.org/abs/2203.15556. Cited on page 55.

[220] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling Laws for Neural Language Models. *arXiv preprint*, 2001.08361, 2020. URL https://arxiv.org/abs/2001.08361. Cited on page 55.

[221] Jonathan S. Rosenfeld, Amir Rosenfeld, Yonatan Belinkov, and Nir Shavit. A constructive prediction of the generalization error across scales. In *ICLR*, 2020. Cited on page 55.

[222] Karl Popper and Karl Popper. *The Logic of Scientific Discovery*. Routledge, London, 2 edition, 2002. ISBN 978-0-203-99462-7. Cited on page 55.

[223] Neal Koblitz. *A course in number theory and cryptography*. Springer Science & Business Media, 1994. ISBN 978-1-4419-8592-7. Cited on page 55.

[224] Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *Journal of Machine Learning Research*, 22(241), 2021. Cited on page 57.

[225] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint*, 2406.04093, 2024. URL https://arxiv.org/abs/2406.04093. Cited on pages 57, 58, and 61.

[226] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *NeurIPS*, 2022. Cited on pages 59 and 60.

[227] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the Wild: A Circuit for Indirect Object Identification in GPT-2 small. *arXiv preprint*, 2211.00593, 2022. URL https://arxiv.org/abs/2211.00593. Cited on page 59.

[228] Aaquib Syed, Can Rager, and Arthur Conmy. Attribution Patching Outperforms Automated Circuit Discovery. *arXiv preprint*, 2310.10348, 2023. URL https://arxiv.org/abs/2310.10348. Cited on page 59.

[229] János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. AtP*: An efficient and scalable method for localizing LLM behaviour to components. *arXiv preprint*, 2403.00745, 2024. URL https://arxiv.org/abs/2403.00745. Cited on page 59.

[230] Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *NeurIPS*, 2023. Cited on page 59.

[231] Lawrence Chan, Adrià Garriga-Alonso, Nicholas Goldowsky-Dill, Ryan Greenblatt, Jenny Nitishinskaya, Ansh Radhakrishnan, Buck Shlegeris, and Nate Thomas. Causal Scrubbing: A method for rigorously testing interpretability hypotheses. *AI Alignment Forum*, 2022. URL https://www.lesswrong.com/posts/JvZhhzycHu2Yd57RN/causal-scrubbing-a-method-for-rigorously-testing. Cited on page 59.

[232] Nicola De Cao, Leon Schmid, Dieuwke Hupkes, and Ivan Titov. Sparse Interventions in Language Models with Differentiable Masking. In Jasmijn Bastings, Yonatan Belinkov, Yanai Elazar, Dieuwke Hupkes, Naomi Saphra, and Sarah Wiegreffe, editors, *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, Abu Dhabi, United Arab Emirates (Hybrid), 2022. Association for Computational Linguistics. Cited on page 59.

[233] Alex Foote, Neel Nanda, Esben Kran, Ionnis Konstas, and Fazl Barez. N2G: A Scalable Approach for Quantifying Interpretable Neuron Representations in Large Language Models. *arXiv preprint*, 2304.12918, 2023. URL https://arxiv.org/abs/2304.12918. Cited on page 59.

[234] Achyuta Rajaram, Neil Chowdhury, Antonio Torralba, Jacob Andreas, and Sarah Schwettmann. Automatic Discovery of Visual Circuits. *arXiv preprint*, 2404.14349, 2024. URL https://arxiv.org/abs/2404.14349. Cited on pages 59 and 60.

[235] Maximilian Dreyer, Erblina Purelku, Johanna Vielhaben, Wojciech Samek, and Sebastian Lapuschkin. PURE: Turning Polysemantic Neurons Into Pure Features by Identifying Relevant Circuits. *arXiv preprint*, 2404.06453, 2024. URL https://arxiv.org/abs/2404.06453. Cited on page 59.

[236] Anshita Gupta, Debanjan Mondal, Akshay Sheshadri, Wenlong Zhao, Xiang Li, Sarah Wiegreffe, and Niket Tandon. Editing Common Sense in Transformers. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Singapore, 2023. Association for Computational Linguistics. Cited on page 60.

[237] Jason Hoelscher-Obermaier, Julia Persson, Esben Kran, Ioannis Konstas, and Fazl Barez. Detecting Edit Failures In Large Language Models: An Improved Specificity Benchmark. *arXiv preprint*, 2305.17553, 2023. URL https://arxiv.org/abs/2305.17553. Cited on page 60.

[238] Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing Knowledge Editing in Language Models via Multi-Hop Questions. *arXiv preprint*, 2305.14795, 2023. URL https://arxiv.org/abs/2305.14795. Cited on page 60.

[239] Siyuan Cheng, Bozhong Tian, Qingbin Liu, Xi Chen, Yongheng Wang, Huajun Chen, and Ningyu Zhang. Can We Edit Multimodal Large Language Models? *arXiv preprint*, 2310.08475, 2023. URL https://arxiv.org/abs/2310.08475. Cited on page 60.

[240] Han Huang, Haitian Zhong, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. KEBench: A Benchmark on Knowledge Editing for Large Vision-Language Models. *arXiv preprint*, 2403.07350, 2024. URL https://arxiv.org/abs/2403.07350. Cited on page 60.

[241] Nora Belrose, David Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. LEACE: Perfect linear concept erasure in closed form. *arXiv preprint*, 2306.03819, 2023. URL https://arxiv.org/abs/2306.03819. Cited on page 60.

[242] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant Risk Minimization. *arXiv preprint*, 1907.02893, 2019. URL https://arxiv.org/abs/1907.02893. Cited on page 60.

[243] Weixin Liang and James Zou. MetaShift: A Dataset of Datasets for Evaluating Contextual Distribution Shifts and Training Conflicts. *arXiv preprint*, 2202.06523, 2022. URL https://arxiv.org/abs/2202.06523. Cited on page 60.

[244] Nishant Subramani, Nivedita Suresh, and Matthew Peters. Extracting Latent Steering Vectors from Pretrained Language Models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, Dublin, Ireland, 2022. Association for Computational Linguistics. Cited on page 60.

[245] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-Time Intervention: Eliciting Truthful Answers from a Language Model. In *NeurIPS*, 2023. Cited on page 60.

[246] Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J. Vazquez, Ulisse Mini, and Monte MacDiarmid. Activation Addition: Steering Language Models Without Optimization. *arXiv preprint*, 2308.10248, 2023. URL https://arxiv.org/abs/2308.10248. Cited on page 60.

[247] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint*, 2001.08361, 2020. Cited on page 61.

[248] Gary Groth-Marnat. *Handbook of Psychological Assessment*. John Wiley & Sons, 2009. Cited on page 61.

[249] DSMTF American Psychiatric Association, American Psychiatric Association, et al. *Diagnostic and Statistical Manual of Mental Disorders: DSM-5*, volume 5. American psychiatric association Washington, DC, 2013. Cited on page 61.

[250] Kate Miriam Loewenthal and Christopher Alan Lewis. *An Introduction to Psychological Tests and Scales*. Routledge, London, 3 edition, 2020. ISBN 978-1-315-56138-7. Cited on page 61.

[251] Gemma-Claire Ali, Grace Ryan, and Mary J. De Silva. Validated Screening Tools for Common Mental Disorders in Low and Middle Income Countries: A Systematic Review. *PLOS ONE*, 11(6), 2016. Cited on page 61.

[252] David M. Clarke, Graeme C. Smith, and Helen E. Herrman. A Comparative Study of Screening Instruments for Mental Disorders in General Hospital Patients. *The International Journal of Psychiatry in Medicine*, 23(4), 1993. Cited on page 61.

[253] Taanvi Ramesh, Artemis Igoumenou, Maria Vazquez Montes, and Seena Fazel. Use of risk assessment instruments to predict violence in forensic psychiatric hospitals: A systematic review and meta-analysis. *European Psychiatry*, 52, 2018. Cited on page 61.

[254] Jay P. Singh and Seena Fazel. Forensic Risk Assessment: A Metareview. *Criminal Justice and Behavior*, 37(9), 2010. Cited on page 61.

[255] Jay P. Singh, Martin Grann, and Seena Fazel. A comparative study of violence risk assessment tools: A systematic review and metaregression analysis of 68 studies involving 25,980 participants. *Clinical Psychology Review*, 31(3), 2011. Cited on page 61.

[256] Michael C. Frank. Baby steps in evaluating the capacities of large language models. *Nature Reviews Psychology*, 2(8), 2023. Cited on page 61.

[257] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating Large Language Models Trained on Code. *arXiv preprint*, 2107.03374, 2021. URL https://arxiv.org/abs/2107.03374. Cited on page 61.

[258] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training Verifiers to Solve Math Word Problems. *arXiv preprint*, 2110.14168, 2021. URL https://arxiv.org/abs/2110.14168. Cited on page 61.

[259] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *ICLR*, 2021. Cited on page 61.

[260] Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring How Models Mimic Human Falsehoods. *arXiv preprint*, 2109.07958, 2021. URL https://arxiv.org/abs/2109.07958. Cited on page 61.

[261] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a Machine Really Finish Your Sentence? *arXiv preprint*, 1905.07830, 2019. URL https://arxiv.org/abs/1905.07830. Cited on page 61.

[262] Colin White, Samuel Dooley, Manley Roberts, Arka Pal, Ben Feuer, Siddhartha Jain, Ravid Shwartz-Ziv, Neel Jain, Khalid Saifullah, Siddartha Naidu, Chinmay Hegde, Yann LeCun, Tom Goldstein, Willie Neiswanger, and Micah Goldblum. LiveBench: A challenging, contamination-free LLM benchmark, 2024. Cited on page 61.

[263] Jen-tse Huang, Wenxuan Wang, Eric John Li, Man Ho Lam, Shujie Ren, Youliang Yuan, Wenxiang Jiao, Zhaopeng Tu, and Michael Lyu. On the Humanity of Conversational AI: Evaluating the Psychological Portrayal of LLMs. In *ICLR*, 2023. Cited on page 61.

[264] Thilo Hagendorff. Machine Psychology: Investigating Emergent Capabilities and Behavior in Large Language Models Using Psychological Methods. *arXiv preprint*, 2303.13988, 2023. URL https://arxiv.org/abs/2303.13988. Cited on page 61.

[265] Kristina Schaaff, Caroline Reinig, and Tim Schlippe. Exploring ChatGPT's Empathic Abilities. In *2023 11th International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2023. Cited on page 61.

[266] Xiting Wang, Liming Jiang, Jose Hernandez-Orallo, David Stillwell, Luning Sun, Fang Luo, and Xing Xie. Evaluating General-Purpose AI with Psychometrics. *arXiv preprint*, 2310.16379, 2023. URL https://arxiv.org/abs/2310.16379. Cited on page 61.

[267] Xingxuan Li, Yutong Li, Lin Qiu, Shafiq Joty, and Lidong Bing. Evaluating Psychological Safety of Large Language Models. *arXiv preprint*, 2212.10529, 2022. URL https://arxiv.org/abs/2212.10529. Cited on page 61.

Appendix

This chapter contains the complete and unmodified publications presented and discussed before in Chapter 2 and Chapter 3 of the present thesis.

A.1 Exemplary Natural Images Explain CNN Activations Better than State-of-the-Art Feature Visualization

The following 41 pages were published as:

Judy Borowski*, **Roland S. Zimmermann***, Judith Schepers, Robert Geirhos, Thomas SA Wallis, Matthias Bethge, and Wieland Brendel. "Exemplary Natural Images Explain CNN Activations Better than State-of-the-Art Feature Visualization." *ICLR (2020)*

A summary is given in Section 2.1.1 on page 28. * Equal contribution.

Abstract

Feature visualizations such as synthetic maximally activating images are a widely used explanation method to better understand the information processing of convolutional neural networks (CNNs). At the same time, there are concerns that these visualizations might not accurately represent CNNs' inner workings. Here, we measure how much extremely activating images help humans to predict CNN activations. Using a well-controlled psychophysical paradigm, we compare the informativeness of synthetic images by Olah et al. (2017) with a simple baseline visualization, namely exemplary natural images that also strongly activate a specific feature map. Given either synthetic or natural reference images, human participants choose which of two query images leads to strong positive activation. The experiments are designed to maximize participants' performance, and are the first to probe intermediate instead of final layer representations. We find that synthetic images indeed provide helpful information about feature map activations ($82 \pm 4\,\%$ accuracy; chance would be $50\,\%$). However, natural images - originally intended as a baseline - outperform synthetic images by a wide margin ($92 \pm 2\,\%$). Additionally, participants are faster and more confident for natural images, whereas subjective impressions about the interpretability of the feature visualizations are mixed. The higher informativeness of natural images holds across most layers, for both expert and lay participants as well as for hand- and randomly-picked feature visualizations. Even if only a single reference image is given, synthetic images provide less information than natural images ($65 \pm 5\,\%$ vs. $73 \pm 4\,\%$). In summary, synthetic images from a popular feature visualization method are significantly less informative for assessing CNN activations than natural images. We argue that visualization methods should improve over this baseline.

Published as a conference paper at ICLR 2021

EXEMPLARY NATURAL IMAGES EXPLAIN CNN ACTIVATIONS BETTER THAN STATE-OF-THE-ART FEATURE VISUALIZATION

Judy Borowski[*], **Roland S. Zimmermann**[*], **Judith Schepers, Robert Geirhos, Thomas S. A. Wallis**[†‡], **Matthias Bethge**[‡], **Wieland Brendel**[‡]
University of Tübingen, Germany

ABSTRACT

Feature visualizations such as synthetic maximally activating images are a widely used explanation method to better understand the information processing of convolutional neural networks (CNNs). At the same time, there are concerns that these visualizations might not accurately represent CNNs' inner workings. Here, we measure how much extremely activating images help humans to predict CNN activations. Using a well-controlled psychophysical paradigm, we compare the informativeness of synthetic images by Olah et al. (2017) with a simple baseline visualization, namely exemplary natural images that also strongly activate a specific feature map. Given either synthetic or natural reference images, human participants choose which of two query images leads to strong positive activation. The experiments are designed to maximize participants' performance, and are the first to probe *intermediate* instead of final layer representations. We find that synthetic images indeed provide helpful information about feature map activations ($82 \pm 4\%$ accuracy; chance would be 50%). However, natural images — originally intended to be a baseline — outperform these synthetic images by a wide margin ($92 \pm 2\%$). Additionally, participants are faster and more confident for natural images, whereas subjective impressions about the interpretability of the feature visualizations by Olah et al. (2017) are mixed. The higher informativeness of natural images holds across most layers, for both expert and lay participants as well as for hand- and randomly-picked feature visualizations. Even if only a single reference image is given, synthetic images provide less information than natural images ($65 \pm 5\%$ vs. $73 \pm 4\%$). In summary, synthetic images from a popular feature visualization method are significantly less informative for assessing CNN activations than natural images. We argue that visualization methods should improve over this simple baseline.

1 INTRODUCTION

As Deep Learning methods are being deployed across society, academia and industry, the need to understand their decisions becomes ever more pressing. Under certain conditions, a "right to explanation" is even required by law in the European Union (GDPR, 2016; Goodman & Flaxman, 2017). Fortunately, the field of *interpretability* or *explainable artificial intelligence* (XAI) is also growing: Not only are discussions on goals and definitions of interpretability advancing (Doshi-Velez & Kim, 2017; Lipton, 2018; Gilpin et al., 2018; Murdoch et al., 2019; Miller, 2019; Samek et al., 2020) but the number of explanation methods is rising, their maturity is evolving (Zeiler & Fergus, 2014; Ribeiro et al., 2016; Selvaraju et al., 2017; Kim et al., 2018) and they are tested and

[*]Joint first and corresponding authors: firstname.lastname@uni-tuebingen.de
[†]Current affiliation: Institute of Psychology and Center for Cognitive Science, Technische Universität Darmstadt
[‡]Joint senior authors

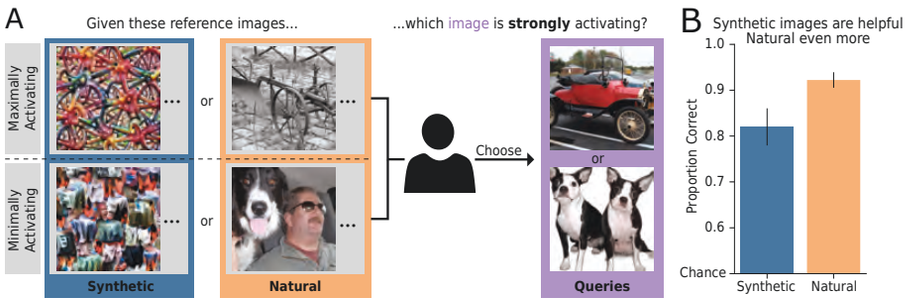


Figure 1: How useful are synthetic compared to natural images for interpreting neural network activations? **A: Human experiment.** Given extremely activating reference images (either synthetic or natural), a human participant chooses which out of two query images is also a strongly activating image. Synthetic images were generated via feature visualization (Olah et al., 2017). **B: Core result.** Participants are well above chance for synthetic images — but even better when seeing *natural* reference images.

used in real-world scenarios like medicine (Cai et al., 2019; Kröll et al., 2020) and meteorology (Ebert-Uphoff & Hilburn, 2020).

We here focus on the popular post-hoc explanation method (or interpretability method) of *feature visualizations via activation maximization*[1]. First introduced by Erhan et al. (2009) and subsequently improved by many others (Mahendran & Vedaldi, 2015; Nguyen et al., 2015; Mordvintsev et al., 2015; Nguyen et al., 2016a; 2017), these synthetic, maximally activating images seek to visualize features that a specific network unit, feature map or a combination thereof is selective for. However, feature visualizations are surrounded by a great controversy: How accurately do they represent a CNN's inner workings—or in short, how useful are they? This is the guiding question of our study.

On the one hand, many researchers are convinced that feature visualizations are interpretable (Graetz, 2019) and that "features can be rigorously studied and understood" (Olah et al., 2020b). Also other applications from Computer Vision and Natural Language Processing support the view that features are meaningful (Mikolov et al., 2013; Karpathy et al., 2015; Radford et al., 2017; Zhou et al., 2014; Bau et al., 2017; 2020) and might be formed in a hierarchical fashion (LeCun et al., 2015; Güçlü & van Gerven, 2015; Goodfellow et al., 2016). Over the past few years, extensive investigations to better understand CNNs are based on feature visualizations (Olah et al., 2020b;a; Cammarata et al., 2020; Cadena et al., 2018), and the technique is being combined with other explanation methods (Olah et al., 2018; Carter et al., 2019; Addepalli et al., 2020; Hohman et al., 2019).

On the other hand, feature visualizations can be equal parts art and engineering as they are science: vanilla methods look noisy, thus human-defined regularization mechanisms are introduced. But do the resulting beautiful visualizations accurately show what a CNN is selective for? How representative are the seemingly well-interpretable, "hand-picked" (Olah et al., 2017) synthetic images in publications for the entirety of all units in a network, a concern raised by e.g. Kriegeskorte (2015)? What if the features that a CNN is truly sensitive to are imperceptible instead, as might be suggested by the existence of adversarial examples (Szegedy et al., 2013; Ilyas et al., 2019)? Morcos et al. (2018) even suggest that units of easily understandable features play a less important role in a network. Another criticism of synthetic maximally activating images is that they only visualize extreme features, while potentially leaving other features undetected that only elicit e.g. 70% of the maximal activation. Also, polysemantic units (Olah et al., 2020b), i.e. units that are highly activated by different semantic concepts, as well as the importance of combinations of units (Olah et al., 2017; 2018; Fong & Vedaldi, 2018) already hint at the complexity of how concepts are encoded in CNNs.

One way to advance this debate is to measure the utility of feature visualizations in terms of their helpfulness for *humans*. In this study, we therefore design well-controlled psychophysical experiments that aim to quantify the informativeness of the popular visualization method by Olah et al. (2017). Specifically, participants choose which of two natural images would elicit a higher activa-

[1]Also known as *input maximization* or *maximally exciting images (MEIs)*.

tion in a CNN given a set of reference images that visualize the network selectivities. We use natural query images because real-world applications of XAI require understanding model decisions to natural inputs. To the best of our knowledge, our study is the first to probe how well humans can predict *intermediate* CNN activations. Our data shows that:

- Synthetic images provide humans with helpful information about feature map activations.
- Exemplary natural images are even more helpful.
- The superiority of natural images mostly holds across the network and various conditions.
- Subjective impressions of the interpretability of the synthetic visualizations vary greatly between participants.

2 Related Work

Significant progress has been made in recent years towards understanding CNNs for image data. Here, we mention a few selected methods as examples of the plethora of approaches for understanding CNN decision-making: *Saliency maps* show the importance of each pixel to the classification decision (Springenberg et al., 2014; Bach et al., 2015; Smilkov et al., 2017; Zintgraf et al., 2017), *concept activation vectors* show a model's sensitivity to human-defined concepts (Kim et al., 2018), and other methods - amongst feature visualizations - focus on explaining individual units (Bau et al., 2020). Some tools integrate interactive, software-like aspects (Hohman et al., 2019; Wang et al., 2020; Carter et al., 2019; Collaris & van Wijk, 2020; OpenAI, 2020), combine more than one explanation method (Shi et al., 2020; Addepalli et al., 2020) or make progress towards automated explanation methods (Lapuschkin et al., 2019; Ghorbani et al., 2019). As overviews, we recommend Gilpin et al. (2018); Zhang & Zhu (2018); Montavon et al. (2018) and Carvalho et al. (2019).

Despite their great insights, challenges for explanation methods remain. Oftentimes, these techniques are criticized as being over-engineered; regarding feature visualizations, this concerns the loss function and techniques to make the synthetic images look interpretable (Nguyen et al., 2017). Another critique is that interpretability research is not sufficiently tested against falsifiable hypotheses and rather relies too much on intuition (Leavitt & Morcos, 2020).

In order to further advance XAI, scientists advocate different directions. Besides the focus on developing additional methods, some researchers (e.g. Olah et al. (2020b)) promote the "natural science" approach, i.e. studying a neural network extensively and making empirical claims until falsification. Yet another direction is to quantitatively evaluate explanation methods. So far, only decision-level explanation methods have been studied in this regard. Quantitative evaluations can either be realized with humans directly or with mathematically-grounded models as an approximation for human perception. Many of the latter approaches show great insights (e.g. Hooker et al. (2019); Nguyen & Martínez (2020); Fel & Vigouroux (2020); Lin et al. (2020); Tritscher et al. (2020); Tjoa & Guan (2020)). However, a recent study demonstrates that metrics of the explanation quality computed without human judgment are inconclusive and do not correspond to the *human* rankings (Biessmann & Refiano, 2019). Additionally, Miller (2019) emphasizes that XAI should build on existing research in philosophy, cognitive science and social psychology.

The body of literature on human evaluations of explanation methods is growing: Various combinations of data types (tabular, text, static images), task set-ups and participant pools (experts vs. laypeople, on-site vs. crowd-sourcing) are being explored. However, these studies all aim to investigate final model decisions and do not probe intermediate activations like our experiments do. For a detailed table of related studies, see Appendix Sec. A.3. A commonly employed task paradigm is the "forward simulation / prediction" task, first introduced by Doshi-Velez & Kim (2017): Participants guess the model's computation based on an input and an explanation. As there is no absolute metric for the goodness of explanation methods (yet), comparisons are always performed within studies, typically against baselines. The same holds for additional data collected for confidence or trust ratings. According to the current literature, studies reporting positive effects of explanations (e.g. Kumarakulasinghe et al. (2020)) slightly outweigh those reporting inconclusive (e.g. Alufaisan et al. (2020); Chu et al. (2020)) or even negative effects (e.g. Shen & Huang (2020)).

Figure 2: Example trial in psychophysical experiments. A participant is shown minimally and maximally activating reference images for a certain feature map on the sides and is asked to select the image from the center that also strongly activates that feature map. The answer is given by clicking on the number according to the participant's confidence level (1: not confident, 2: somewhat confident, 3: very confident). After each trial, the participant receives feedback which image was indeed the maximally activating one. For screenshots of each step in the task, see Appendix Fig. 7.

To our knowledge, no study has yet evaluated the popular explanation method of feature visualizations and how it improves human understanding of intermediate network activations. This study therefore closes an important gap: By presenting data for a forward prediction task of a CNN, we provide a quantitative estimate of the informativeness of maximally activating images generated with the method of Olah et al. (2017). Furthermore, our experiments are unique as they probe for the first time how well humans can predict *intermediate* model activations.

3 METHODS

We perform two human psychophysical studies[2] with different foci (Experiment I ($N = 10$) and Experiment II ($N = 23$)). In both studies, the task is to choose the one image out of two natural query images (two-alternative forced choice paradigm) that the participant considers to also elicit a strong activation given some reference images (see Fig. 2). Apart from the image choice, we record the participant's confidence level and reaction time. Specifically, responses are given by clicking on the confidence levels belonging to either query image. In order to gain insights into how intuitive participants find feature visualizations, their subjective judgments are collected in a separate task and a dynamic conversation after the experiment (for details, see Appendix Sec. A.1.1 and Appendix Sec. A.2.6).

All design choices are made with two main goals: (1) allowing participants to achieve the *best performance possible* to approximate an upper bound on the helpfulness of the explanation method, and (2) gaining a *general* impression of the helpfulness of the examined method. As an example, we choose the natural query images from among those of lowest and highest activations ($\rightarrow$ best possible performance) and test many different feature maps across the network ($\rightarrow$ generality). For more details on the human experiment besides the ones below, see Appendix Sec. A.1.

In Experiment I, we focus on comparing the performance of synthetic images to two baseline conditions: natural reference images and no reference images. In Experiment II, we compare lay vs. expert participants as well as different presentation schemes of reference images. Expert participants qualify by being familiar or having practical experience with feature visualization techniques or at least CNNs. Regarding presentation schemes, we vary whether only maximally or both maximally and minimally activating images are shown; as well as how many example images of each of these are presented (1 or 9).

Following the existing work on feature visualization (Olah et al., 2017; 2018; 2020b;a), we use an Inception V1 network[3] (Szegedy et al., 2015) trained on ImageNet (Deng et al., 2009; Russakovsky

[2]Code and data is available at `https://bethgelab.github.io/testing_visualizations/`

[3]also known as GoogLeNet

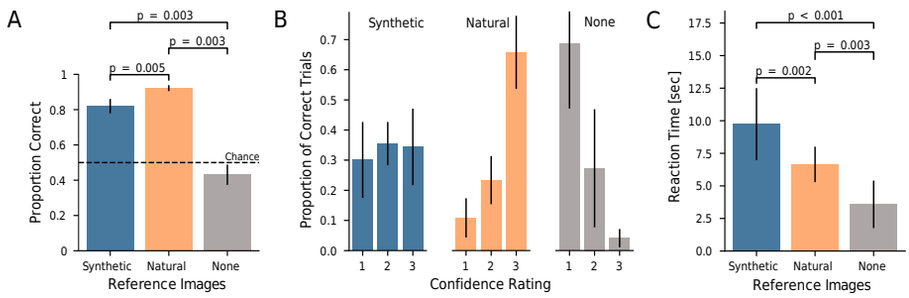


Figure 3: Participants are better, more confident and faster at judging which of two query images causes higher feature map activation with natural than with synthetic reference images. **A: Performance.** Given synthetic reference images, participants are well above chance (proportion correct: $82 \pm 4\%$), but even better for natural reference images ($92 \pm 2\%$). Without reference images (baseline comparison "None"), participants are close to chance. **B: Confidence.** Participants are much more confident (higher rating = more confident) for natural than for synthetic images on correctly answered trials (χ^2, $p < .001$). **C: Reaction time.** For correctly answered trials, participants are on average faster when presented with natural than with synthetic reference images. We show additional plots on confidence and reaction time for incorrectly answered trials and all trials in the Appendix (Fig. 16); for Experiment II, see Fig. 17.). The p-values in A and C correspond to Wilcoxon signed-rank tests.

et al., 2015). The synthetic images throughout this study are the optimization results of the feature visualization method by Olah et al. (2017) with the spatial average of a whole feature map ("channel objective"). The natural stimuli are selected from the validation set of the ImageNet ILSVRC 2012 dataset (Russakovsky et al., 2015) according to their activations for the feature map of interest. Specifically, the images of the most extreme activations are sampled, while ensuring that each lay or expert participant sees different query and reference images. A more detailed description of the specific sampling process for natural stimuli and the generation process of synthetic stimuli is given in Sec. A.1.2.

4 Results

In this section, all figures show data from Experiment I except for Fig. 5A+C, which show data from Experiment II. All figures for Experiment II, which replicate the findings of Experiment I, as well as additional figures for Experiment I (such as a by-feature-map analysis), can be found in the Appendix Sec. A.2. Note that (unless explicitly noted otherwise), error bars denote two standard errors of the mean of the participant average metric.

4.1 Participants are better, more confident and faster with natural images

Synthetic images can be helpful: Given synthetic reference images generated via feature visualization (Olah et al., 2017), participants are able to predict whether a certain network feature map prefers one over the other query image with an accuracy of $82 \pm 4\%$, which is well above chance level (50%) (see Fig. 3A). However, performance is even higher in what we intended to be the baseline condition: natural reference images ($92 \pm 2\%$). Additionally, for correct answers, participants much more frequently report being highly certain on natural relative to synthetic trials (see Fig. 3B), and their average reaction time is approximately 3.7 seconds faster when seeing natural than synthetic reference images (see Fig. 3C). Taken together, these findings indicate that in our setup, participants are not just better overall, but also more confident and substantially faster on natural images.

4.2 Natural images are more helpful across a broad range of layers

Next, we take a more fine-grained look at performance across different layers and branches of the Inception modules (see Fig. 4). Generally, feature map visualizations from lower layers show low-level features such as striped patterns, color or texture, whereas feature map visualizations from

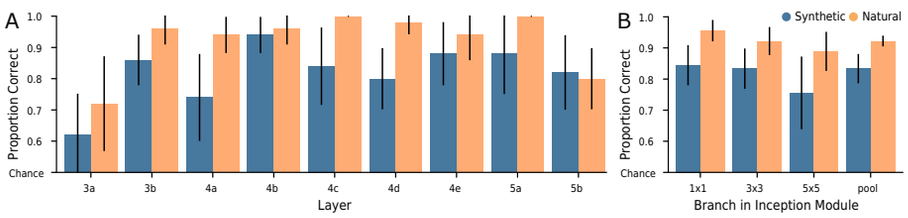


Figure 4: Performance is high across (**A**) a broad range of layers and (**B**) all branches of the Inception modules. The latter differ in their kernel sizes (1×1, 3×3, 5×5, pool). Again, natural images are (mostly) more helpful than synthetic images. Additional plots for the none condition as well as Experiment II can be found in the Appendix in respectively Fig. 18 and Fig. 19.

higher layers tend to show more high-level concepts like (parts of) objects (LeCun et al., 2015; Güçlü & van Gerven, 2015; Goodfellow et al., 2016). We find performance to be reasonably high across most layers and branches: participants are able to match both low-level and high-level patterns (despite not being explicitly instructed what layer a feature map belonged to). Again, natural images are mostly more helpful than synthetic images.

4.3 For expert and lay participants alike: natural images are more helpful

Explanation methods seek to explain aspects of algorithmic decision-making. Importantly, an explanation should not just be amenable to experts but to anyone affected by an algorithm's decision. We here test whether the explanation method of feature visualization is equally applicable to expert and lay participants (see Fig. 5A). Contrary to our prior expectation, we find no significant differences in expert vs. lay performance (RM ANOVA, $p = .44$, for details see Appendix Sec. A.2.2). Hence, extensive experience with CNNs is not necessary to perform well in this forward simulation task. In line with the previous main finding, both experts and lay participants are both better in the natural than in the synthetic condition.

4.4 Even for hand-picked feature visualizations, performance is higher on natural images

Often, explanation methods are presented using carefully selected network units, raising the question whether author-chosen units are representative for the interpretability method as a whole. Olah et al. (2017) identify a number of particularly interpretable feature maps in Inception V1 in their appendix overview. When presenting either these hand-picked visualizations[4] or randomly selected ones, performance for hand-picked feature maps improves slightly (Fig. 5B); however this performance difference is small and not significant for both natural (Wilcoxon test, $p = .59$) and synthetic (Wilcoxon test, $p = .18$) reference images (see Appendix Sec. A.2.4 for further analysis). Consistent with the findings reported above, performance is higher for natural than for synthetic reference images *even on carefully selected hand-picked feature maps*.

4.5 Additional information boosts performance, especially for natural images

Publications on feature visualizations vary in terms of how optimized images are presented: Often, a single maximally activating image is shown (e.g. Erhan et al. (2009); Carter et al. (2019); Olah et al. (2018)); sometimes a few images are shown simultaneously (e.g. Yosinski et al. (2015); Nguyen et al. (2016b)), and on occasion both maximally and minimally activating images are shown in unison (Olah et al. (2017)). Naturally, the question arises as to what influence (if any) these choices have, and whether there is an optimal way of presenting extremely activating images. For this reason, we systematically compare approaches along two dimensions: the number of reference images (1 vs. 9) and the availability of minimally activating images (only Max vs. Min+Max). The results can

[4]All our hand-picked feature maps are taken from the pooling branch of the Inception module. As the appendix overview in Olah et al. (2017) does not contain one feature map for each of these, *we* select interpretable feature maps for the missing layers mixed5a and mixed5b ourselves.

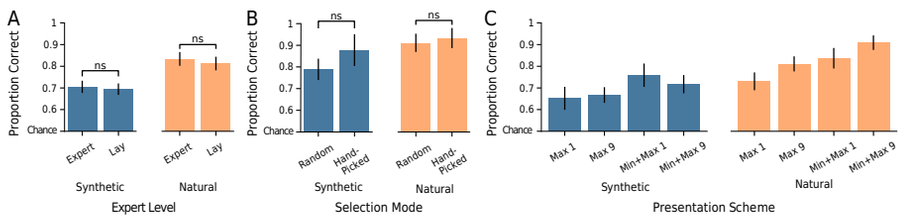


Figure 5: We found no evidence for large effects of expert level or feature map selection. However, performance does improve with additional information. **A: Expert level.** Both experts and lay participants perform equally well (RM ANOVA, $p = .44$), and consistently better on natural than on synthetic images. **B: Selection mode.** There is no significant performance difference between hand-picked feature maps selected for interpretability and randomly selected ones (Wilcoxon test, $p = .18$ for synthetic and $p = .59$ for natural reference images). **C: Presentation scheme.** Presenting both maximally and minimally activating images simultaneously (Min+Max) and presenting nine instead of one single reference image tend to improve performance, especially for natural reference images. "ns" highlights non-significant differences.

be found in Fig. 5C. When just a single maximally activating image is presented (condition Max 1), natural images already outperform synthetic images ($73 \pm 4\%$ vs. $64 \pm 5\%$). With additional information along either dimension, performance improves both for natural as well as for synthetic images. The stronger boost in performance, however, is observed for natural reference images. In fact, performance is higher for natural than for synthetic reference images in all four conditions. In the Min+Max 9 condition, a replication of the result from Experiment I shown in Fig. 3A, natural images now outperform synthetic images by an even larger margin (91 ± 3 vs. $72 \pm 4\%$).

4.6 Subjectively, interpretability of feature visualizations varies greatly

While our data suggests that feature visualizations are indeed helpful for humans to predict CNN activations, we want to emphasize again that our design choices aim at an upper bound on their informativeness. Another important aspect of evaluating an explanation method is the subjective impression. Besides recording confidence ratings and reaction times, we collect judgments on *intuitiveness trials* (see Appendix Fig. 14) and oral impressions after the experiments. The former ask for ratings of how intuitive feature visualizations appear for natural images. As Fig. 6A+B show, participants perceive the intuitiveness of synthetic feature visualizations for strongly activating natural dataset images very differently. Further, the comparison of intuitiveness judgments before and after the main experiments reveals only a small significant average improvement for one out of three feature maps (see Fig. 6B+C, Wilcoxon test, $p < .001$ for mixed4b). The interactive conversations paint a similar picture: Some synthetic feature visualizations are perceived as intuitive while others do not correspond to understandable concepts. Nonetheless, four participants report that their first "gut feeling" for interpreting these reference images (as one participant phrased it) is more reliable. Further, a few participants point out that the synthetic visualizations are exhausting to understand. Finally, three participants additionally emphasize that the minimally activating reference images played an important role in their decision-making.

In a by-feature-map analysis (see Appendix A.2.7 for details and images, as well as Supplementary Material 1 for more images), we compare differences and commonalities for feature maps of different performance levels. According to our observations, easy feature maps seem to contain clear object parts or shapes. In contrast, difficult feature maps seem to have diverse reference images, features that do not correspond to human concepts, or contain conflicting information as to which commonalities between query and reference images matter more. Bluntly speaking, we are also often surprised that participants identified the correct image — the reasons for this are unclear to us.

5 Discussion & Conclusion

Feature visualizations such as synthetic maximally activating images are a widely used explanation method, but it is unclear whether they indeed help humans to understand CNNs. Using well-controlled psychophysical experiments with both expert and lay participants, we here conduct the

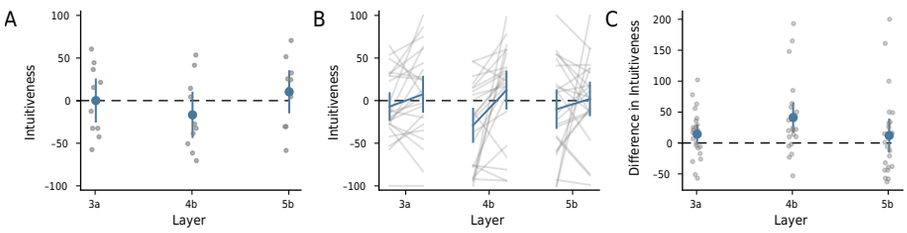


Figure 6: The subjective intuitiveness of feature visualizations varies greatly (see **A** for the ratings from the beginning of Experiment I and **B** for the ratings at the beginning and end of Experiment II). The means over all participants yield a neutral result, i.e. the visualizations are neither un- nor intuitive, and the improvement of subjective intuitiveness before and after the experiment is only significant for one feature map (mixed4b). **C:** On average, participants found feature visualizations slightly more intuitive after doing the experiment as the differences larger than zero show. In all three subfigures, gray dots and lines show data per participant.

very first investigation of intermediate synthetic feature visualizations by Olah et al. (2017): Can participants predict which of two query images leads to a strong activation in a feature map, given extremely activating visualizations? Specifically, we shed light on the following questions:

(1.) *How informative are synthetic feature visualizations — and how do they compare to a natural image baseline?* We find above-chance performance given synthetic feature visualizations, but to our own surprise, synthetic feature visualizations are systematically *less* informative than the simple baseline of strongly activating natural images. Interestingly, many synthetic feature visualizations contain regularization mechanisms to introduce more "natural structure" (Olah et al., 2017), sometimes even called a "natural image prior" (Mahendran & Vedaldi, 2015; Offert & Bell, 2020). This raises the question: Are natural images maybe all you need? One might posit that extremely activating natural (reference) images would have an unfair advantage because we also test on extremely activating natural (query) images. However, our task design ultimately reflects that XAI is mainly concerned with explaining how units behave on *natural* inputs. Furthermore, the fact that feature visualization are not bound to the natural image manifold is often claimed as an advantage because it supposedly allows them to capture more precisely which features a unit is sensitive to (Olah et al., 2017). Our results, though, demonstrate that this is not the case if we want to understand the behavior of units on natural inputs.

(2.) *Do you need to be a CNN expert in order to understand feature visualizations?* To the best of our knowledge, our study is the first to compare the performances of expert and lay people when evaluating explanation methods. Previously, publications either focused on only expert groups (Hase & Bansal, 2020; Kumarakulasinghe et al., 2020) or only laypeople (Schmidt & Biessmann, 2019; Alufaisan et al., 2020). Our experiment shows no significant difference between expert and lay participants in our task — both perform similarly well, and even better on natural images: a replication of our main finding. While a few caveats remain when moving an experiment from the well-controlled lab to a crowdsourcing platform (Haghiri et al., 2019), this suggests that future studies may not have to rely on selected expert participants, but may leverage larger lay participant pools.

(3.) *Are hand-picked synthetic feature visualizations representative?* An open question was whether the visualizations shown in publications represent the general interpretability of feature visualizations (a concern voiced by e.g. Kriegeskorte, 2015), even though they are hand-picked (Olah et al., 2017). Our finding that there is no large difference in performance between hand- and randomly-picked feature visualizations suggests that this aspect is minor.

(4.) *What is the best way of presenting images?* Existing work suggests that more than one example (Offert, 2017) and particularly negative examples (Kim et al., 2016) enhance human understanding of data distributions. Our systematic exploration of presentation schemes provides evidence that increasing the number of reference images as well as presenting both minimally *and* maximally activating reference images (as opposed to only maximally activating ones) improve human performance. This finding might be of interest to future studies aiming at peak performance or for developing software for understanding CNNs.

(5.) *How do humans subjectively perceive feature visualizations?* Apart from the high informativeness of explanations, another relevant question is how much trust humans have in them. In our experiment, we find that subjective impressions of how reasonable synthetic feature visualizations are for explaining responses to natural images vary greatly. This finding is in line with Hase & Bansal (2020) who evaluated explanation methods on text and tabular data.

Caveats. Despite our best intentions, a few caveats remain: The forward simulation paradigm is only one specific way to measure the informativeness of explanation methods, but does not allow us to make judgments about their helpfulness in other applications such as comparing different CNNs. Further, we emphasize that all experimental design choices were made with the goal to measure the best possible performance. As a consequence, our finding that synthetic reference images help humans predict a network's strongly activating image may not necessarily be representative of a less optimal experimental set-up with e.g. query images corresponding to less extreme feature map activations. Knobs to further de- or increase participant performance remain (e.g. hyper-parameter choices could be tuned to layers). Finally, while we explored one particular method in depth (Olah et al., 2017); it remains an open question whether the results can be replicated for other feature visualizations methods.

Future directions. We see many promising future directions. For one, the current study uses query images from extreme opposite ends of a feature map's activation spectrum. For a more fine-grained measure of informativeness, we will study query images that elicit more similar activations. Additionally, future participants could be provided with even *more* information—such as, for example, where a feature map is located in the network. Furthermore, it has been suggested that the combination of synthetic and natural reference images might provide synergistic information to participants (Olah et al., 2017), which could again be studied in our experimental paradigm. Finally, further studies could explore single neuron-centered feature visualizations, combinations of units as well as different network architectures.

Taken together, our results highlight the need for thorough human quantitative evaluations of feature visualizations and suggest that example natural images provide a surprisingly challenging baseline for understanding CNN activations.

AUTHOR CONTRIBUTIONS

The initiative of investigating human predictability of CNN activations came from WB. JB, WB, MB and TSAW jointly combined it with the idea of investigating human interpretability of feature visualizations. JB led the project. JB, RSZ and JS jointly designed and implemented the experiments (with advice and feedback from TSAW, RG, MB and WB). The data analysis was performed by JB and RSZ (with advice and feedback from RG, TSAW, MB and WB). JB designed, and JB and JS implemented the pilot study. JB conducted the experiments (with help from JS). RSZ performed the statistical significance tests (with advice from TSAW and feedback from JB and RG). MB helped shape the bigger picture and initiated intuitiveness trials. WB provided day-to-day supervision. JB, RSZ and RG wrote the initial version of the manuscript. All authors contributed to the final version of the manuscript.

ACKNOWLEDGMENTS

We thank Felix A. Wichmann and Isabel Valera for helpful discussions. We further thank Alexander Böttcher and Stefan Sietzen for support as well as helfpul discussions on technical details. Additionally, we thank Chris Olah for clarifications via `slack.distill.pub`. Moreover, we thank Leon Sixt for valuable feedback on the introduction and related work. From our lab, we thank Matthias Kümmerer, Matthias Tangemann, Evgenia Rusak and Ori Press for helping in piloting our experiments, as well as feedback from Evgenia Rusak, Claudio Michaelis, Dylan Paiton and Matthias Kümmerer. And finally, we thank all our participants for taking part in our experiments.

We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting JB, RZ and RG. We acknowledge support from the German Federal Ministry of Education and Research (BMBF) through the Competence Center for Machine Learning (TUE.AI, FKZ 01IS18039A) and the Bernstein Computational Neuroscience Program Tübingen (FKZ: 01GQ1002), the Cluster of Excellence Machine Learning: New Perspectives for Sciences (EXC2064/1), and the German Research Foundation (DFG; SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms, TP3, project number 276693517).

REFERENCES

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Sravanti Addepalli, Dipesh Tamboli, R Venkatesh Babu, and Biplab Banerjee. Saliency-driven class impressions for feature visualization of deep neural networks. *arXiv preprint arXiv:2007.15861*, 2020.

Ahmed Alqaraawi, Martin Schuessler, Philipp Weiß, Enrico Costanza, and Nadia Berthouze. Evaluating saliency map explanations for convolutional neural networks: a user study. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pp. 275–285, 2020.

Yasmeen Alufaisan, Laura R Marusich, Jonathan Z Bakdash, Yan Zhou, and Murat Kantarcioglu. Does explainable artificial intelligence improve human decision-making? *arXiv preprint arXiv:2006.11194*, 2020.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 2020.

Felix Biessmann and Dionysius Irza Refiano. A psychophysics approach for quantitative comparison of interpretable computer vision models. *arXiv preprint arXiv:1912.05011*, 2019.

Santiago A Cadena, Marissa A Weis, Leon A Gatys, Matthias Bethge, and Alexander S Ecker. Diverse feature visualizations reveal invariances in early layers of deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 217–232, 2018.

Carrie J Cai, Emily Reif, Narayan Hegde, Jason Hipp, Been Kim, Daniel Smilkov, Martin Wattenberg, Fernanda Viegas, Greg S Corrado, Martin C Stumpe, et al. Human-centered tools for coping with imperfect algorithms during medical decision-making. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, 2019.

Nick Cammarata, Gabriel Goh, Shan Carter, Ludwig Schubert, Michael Petrov, and Chris Olah. Curve detectors. *Distill*, 2020. doi: 10.23915/distill.00024.003. https://distill.pub/2020/circuits/curve-detectors.

Shan Carter, Zan Armstrong, Ludwig Schubert, Ian Johnson, and Chris Olah. Activation atlas. *Distill*, 2019. doi: 10.23915/distill.00015. https://distill.pub/2019/activation-atlas.

Diogo V Carvalho, Eduardo M Pereira, and Jaime S Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8):832, 2019.

Arjun Chandrasekaran, Deshraj Yadav, Prithvijit Chattopadhyay, Viraj Prabhu, and Devi Parikh. It takes two to tango: Towards theory of ai's mind. *arXiv preprint arXiv:1704.00717*, 2017.

Eric Chu, Deb Roy, and Jacob Andreas. Are visual explanations useful? a case study in model-in-the-loop prediction. *arXiv preprint arXiv:2007.12248*, 2020.

Dennis Collaris and Jarke J van Wijk. Explainexplore: Visual exploration of machine learning explanations. In *2020 IEEE Pacific Visualization Symposium (PacificVis)*, pp. 26–35. IEEE, 2020.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Jürgen Dieber and Sabrina Kirrane. Why model why? assessing the strengths and limitations of lime. *arXiv preprint arXiv:2012.00093*, 2020.

Jonathan Dinu, Jeffrey Bigham, and J Zico Kolter. Challenging common interpretability assumptions in feature attribution explanations. *arXiv preprint arXiv:2012.02748*, 2020.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

Imme Ebert-Uphoff and Kyle Hilburn. Evaluation, tuning and interpretation of neural networks for working with images in meteorological applications. *Bulletin of the American Meteorological Society*, pp. 1–49, 2020.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

Thomas Fel and David Vigouroux. Representativity and consistency measures for deep neural network explanations. *arXiv preprint arXiv:2009.04521*, 2020.

Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8730–8738, 2018.

GDPR. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46. *Official Journal of the European Union (OJ)*, 59(1-88):294, 2016.

Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, pp. 9277–9286, 2019.

Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pp. 80–89. IEEE, 2018.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

Bryce Goodman and Seth Flaxman. European Union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017.

Fabio M. Graetz. How to visualize convolutional features in 40 lines of code, Jan 2019. URL `https://towardsdatascience.com/how-to-visualize-convolutional-features-in-40-lines-of-code-70b7d87b0030`.

Umut Güçlü and Marcel AJ van Gerven. Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience*, 35(27):10005–10014, 2015.

Siavash Haghiri, Patricia Rubisch, Robert Geirhos, Felix Wichmann, and Ulrike von Luxburg. Comparison-based framework for psychophysics: Lab versus crowdsourcing. *arXiv preprint arXiv:1905.07234*, 2019.

Peter Hase and Mohit Bansal. Evaluating explainable ai: Which algorithmic explanations help users predict model behavior? *arXiv preprint arXiv:2005.01831*, 2020.

Fred Hohman, Haekyu Park, Caleb Robinson, and Duen Horng Polo Chau. Summit: Scaling deep learning interpretability by visualizing activation and attribution summarizations. *IEEE transactions on visualization and computer graphics*, 26(1):1096–1106, 2019.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 9737–9748, 2019.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *Advances in Neural Information Processing Systems*, pp. 125–136, 2019.

JASP Team. JASP (Version 0.13.1), 2020. URL `https://jasp-stats.org/`.

Jeya Vikranth Jeyakumar, Joseph Noor, Yu-Hsi Cheng, Luis Garcia, and Mani Srivastava. How can i explain this to you? an empirical study of deep neural network explanation methods. *Advances in Neural Information Processing Systems*, 33, 2020.

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*, 2015.

Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in neural information processing systems*, pp. 2280–2288, 2016.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.

Jean-Philippe Kröll, Simon B Eickhoff, Felix Hoffstaedter, and Kaustubh R Patil. Evolving complex yet interpretable representations: application to alzheimer's diagnosis and prognosis. In *2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8. IEEE, 2020.

Nesaretnam Barr Kumarakulasinghe, Tobias Blomberg, Jintai Liu, Alexandra Saraiva Leao, and Panagiotis Papapetrou. Evaluating local interpretable model-agnostic explanations on clinical machine learning classification models. In *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, pp. 7–12. IEEE, 2020.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1–8, 2019.

Matthew L Leavitt and Ari Morcos. Towards falsifiable interpretability research. *arXiv preprint arXiv:2010.12016*, 2020.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Yi-Shan Lin, Wen-Chuan Lee, and Z Berkay Celik. What do you see? evaluation of explainable artificial intelligence (xai) interpretability through neural backdoors. *arXiv preprint arXiv:2009.10639*, 2020.

Zachary C Lipton. The mythos of model interpretability. *Queue*, 16(3):31–57, 2018.

Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5188–5196, 2015.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018.

Ari S Morcos, David GT Barrett, Neil C Rabinowitz, and Matthew Botvinick. On the importance of single directions for generalization. *arXiv preprint arXiv:1803.06959*, 2018.

Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, 2015.

W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Interpretable machine learning: definitions, methods, and applications. *arXiv preprint arXiv:1901.04592*, 2019.

Karthikeyan Natesan Ramamurthy, Bhanukiran Vinzamuri, Yunfeng Zhang, and Amit Dhurandhar. Model agnostic multilevel explanations. *Advances in Neural Information Processing Systems*, 33, 2020.

An-phi Nguyen and María Rodríguez Martínez. On quantitative aspects of model interpretability. *arXiv preprint arXiv:2007.07584*, 2020.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in neural information processing systems*, pp. 3387–3395, 2016a.

Anh Nguyen, Jason Yosinski, and Jeff Clune. Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *arXiv preprint arXiv:1602.03616*, 2016b.

Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4467–4477, 2017.

Fabian Offert. " i know it when i see it". visualization and intuitive interpretability. *arXiv preprint arXiv:1711.08042*, 2017.

Fabian Offert and Peter Bell. Perceptual bias and technical metapictures: critical machine vision as a humanities challenge. *AI & SOCIETY*, pp. 1–12, 2020.

Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11):e7, 2017.

Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. The building blocks of interpretability. *Distill*, 3(3):e10, 2018.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. An overview of early vision in inceptionv1. *Distill*, 2020a. doi: 10.23915/distill.00024.002. https://distill.pub/2020/circuits/early-vision.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020b. doi: 10.23915/distill.00024.001. https://distill.pub/2020/circuits/zoom-in.

OpenAI. OpenAI Microscope. `https://microscope.openai.com/models`, 2020. (Accessed on 09/12/2020).

Jonathan Peirce, Jeremy R Gray, Sol Simpson, Michael MacAskill, Richard Höchenberger, Hiroyuki Sogo, Erik Kastman, and Jonas Kristoffer Lindeløv. Psychopy2: Experiments in behavior made easy. *Behavior research methods*, 51(1):195–203, 2019.

Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. Learning to generate reviews and discovering sentiment. *arXiv preprint arXiv:1704.01444*, 2017.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Anchors: High-precision model-agnostic explanations. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J Anders, and Klaus-Robert Müller. Toward interpretable machine learning: Transparent deep neural networks and beyond. *arXiv preprint arXiv:2003.07631*, 2020.

Philipp Schmidt and Felix Biessmann. Quantifying interpretability and trust in machine learning systems. *arXiv preprint arXiv:1901.08558*, 2019.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Hua Shen and Ting-Hao'Kenneth' Huang. How useful are the machine-generated interpretations to general users? a human evaluation on guessing the incorrectly predicted labels. *arXiv preprint arXiv:2008.11721*, 2020.

Rui Shi, Tianxing Li, and Yasushi Yamaguchi. Group visualization of class-discriminative features. *Neural Networks*, 2020.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

Erico Tjoa and Cuntai Guan. Quantifying explainability of saliency methods in deep neural networks. *arXiv preprint arXiv:2009.02899*, 2020.

Julian Tritscher, Markus Ring, Daniel Schlr, Lena Hettinger, and Andreas Hotho. Evaluation of post-hoc xai approaches through synthetic tabular data. In *International Symposium on Methodologies for Intelligent Systems*, pp. 422–430. Springer, 2020.

Zijie J Wang, Robert Turko, Omar Shaikh, Haekyu Park, Nilaksh Das, Fred Hohman, Minsuk Kahng, and Duen Horng Chau. Cnn explainer: Learning convolutional neural networks with interactive visualization. *arXiv preprint arXiv:2004.15004*, 2020.

Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pp. 818–833. Springer, 2014.

Quan-shi Zhang and Song-Chun Zhu. Visual interpretability for deep learning: a survey. *Frontiers of Information Technology & Electronic Engineering*, 19(1):27–39, 2018.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014.

Luisa M Zintgraf, Taco S Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *arXiv preprint arXiv:1702.04595*, 2017.

A APPENDIX

A.1 DETAILS ON METHODS

A.1.1 HUMAN EXPERIMENTS

In our two human psychophysical studies, we ask humans to predict a feature map's strongly activating image ("forward simulation task", Doshi-Velez & Kim 2017). Answers to the two-alternative forced choice paradigm are recorded together with the participants' confidence level (1: not confident, 2: somewhat confident, 3: very confident, see Fig. 7). Time per trial is unlimited and we record reaction time. After each trial, feedback is given (see Fig. 7). A progress bar at the bottom of the screen indicates how many trials of a block are already completed. As reference images, either synthetic, natural or no reference images are given. The synthetic images are the feature visualizations from the method of Olah et al. (2017). Trials of different reference images are arranged in blocks. Synthetic and natural reference images are alternated, and, in the case of Experiment I, framed by trials without reference images (see Fig. 8A, B). The order of the reference image types is counter-balanced across subjects.

The main trials in the experiments are complemented by practice, catch and intuitiveness trials. To avoid learning effects, we use different feature maps for each trial type per participant. Specifically, *practice trials* give participants the opportunity to familiarize themselves with the task. In order to monitor the attention of participants, *catch trials* appear randomly throughout blocks of main trials. Here, the query images are a copy of one of the reference images, i.e., there is an obvious correct answer (see Fig. 15). This control mechanism allows us to decide whether trial blocks should be excluded from the analysis due to e.g. fatigue. To obtain the participant's subjective impression of the helpfulness of maximally activating images, the experiments are preceded (and also succeeded in the case of Experiment II) by three *intuitiveness trials* (see Fig. 14). Here, participants judge in a slightly different task design how intuitive they consider the synthetic stimuli for the natural stimuli. For more details on the intuitiveness trials, see below.

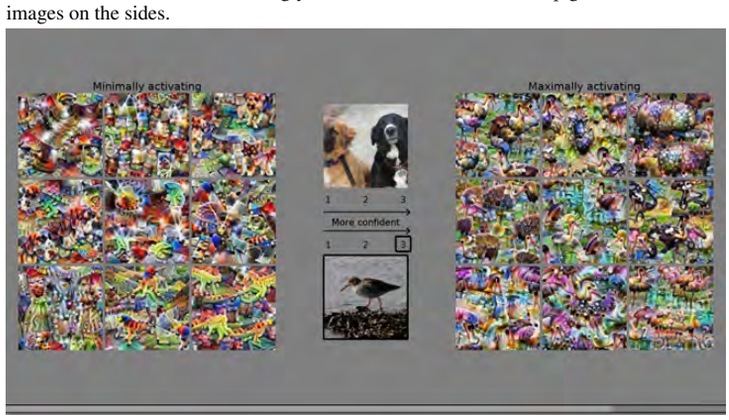
At the end of the experiment, all expert participants in Experiment I and all lay (but not expert) participants in Experiment II are asked about their strategy and whether it changed over time. The information gained through the first group allows us to understand the variety of cues used and paves the way to identify interesting directions for follow-up experiments. The information gained through the second group allowed comparisons to experts' impressions reported in Experiment I.

Experiment I The first experiment focuses on comparing performance of synthetic images to two baselines: natural reference images and no reference images (see Fig. 8A). Screenshots of trials are shown in Fig. 12. In total, 45 feature maps are tested: 36 of these are uniformly sampled from the feature maps of each of the four branches for each of the nine Inception modules. The other nine feature maps are uniformly hand-picked for interpretability from the Inception modules' pooling branch based on the appendix overview selection provided by Olah et al. (2017) or based on our own choices. In the spirit of a *general* statement about the explainability method, different participants see different natural reference and query images, and each participant sees different natural query images for the same feature maps in different reference conditions. To check the consistency of participants' responses, we repeat six randomly chosen main trials for each of the three tested reference image types at the end of the experiment.

Experiment II The second experiment (see Fig. 8B) is about testing expert vs. lay participants as well as comparing different presentation schemes[5] (Max 1, Min+Max 1, Max 9 and Min+Max 9, see Fig. 8E). Screenshots of trials are shown in Fig. 13. In total, 80 feature maps are tested: They are uniformly sampled from every second layer with an Inception module of the network (hence a total of 5 instead of 9 layers), and from all four branches of the Inception modules. Given the focus on four different presentation schemes in this experiment, we repeat the sampling method four times without overlap. In terms of reference image types, only synthetic and natural images are tested. Like in Experiment I, different participants see different natural reference and query images.

[5] In pilot experiments, we learned that participants preferred 9 over 4 reference images, hence this "default" choice in Experiment I.

(a) Screen at the beginning of a trial. The question is which of the two natural images at the center of the screen also strongly activates the CNN feature map given the reference images on the sides.



(b) Screen including a participant's answer visualized by black boxes around the image and the confidence level. A participant indicates which natural image at the center would also be a strongly activating image by clicking on the number corresponding to his/her confidence level (1: not confident, 2: somewhat confident, 3: confident). The time until a participant selects an answer is recorded ("reaction time").

(c) Screen including a participant's answer (black boxes) and feedback on which image is indeed also a strongly activating image (green box).

Figure 7: Forward Simulation Task. The progress bar at the bottom of the screen indicates the progress within one block of trials.

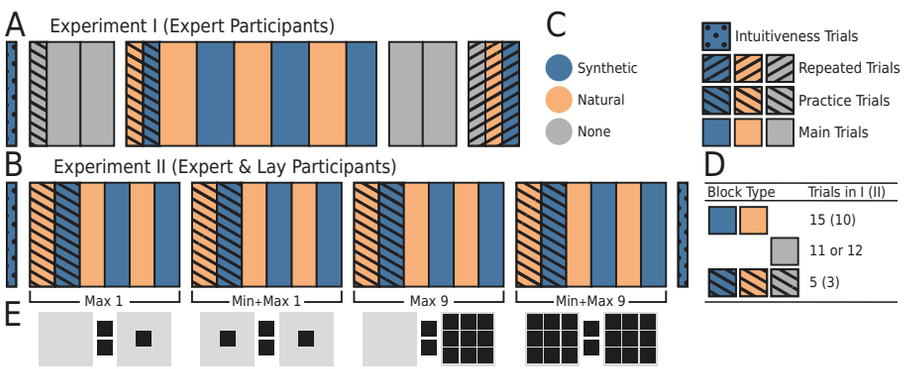


Figure 8: Detailed structure of the two experiments with different foci. **A: Experiment I.** Here, the focus is on comparing performance of synthetic and natural reference images to the most simple baseline: no reference images ("None"). To counter-balance conditions, the order of natural and synthetic blocks is alternated across participants. For each of the three reference image types (synthetic, natural and none), 45 relevant trials are used plus additional catch, practice and repeated trials. **B: Experiment II.** Here, the focus is on testing expert and lay participants as well as comparing different presentation schemes (Max 1, Min+Max 1, Max 9 and Min+Max 9, see **E** for illustrations). Both the order of natural and synthetic blocks as well as the four presentation conditions are counterbalanced across participants. To maintain a reasonable experiment length for each participant, only 20 relevant trials are used per reference image type and presentation scheme, plus additional catch and practice trials. **C:** Legend. **D:** Number of trials per block type (i.e. reference image type and main vs. practice trial) and experiment. Catch trials are not shown in the figure; there was a total of 3 (2) catch trials per each synthetic and natural main block in Experiment I (II). **E:** Illustration of presentation schemes. In Experiment II, all four schemes are tested, in Experiment I only Min+Max 9 is tested.

However, expert and lay participants see the same images. For details on the counter-balancing of all conditions, please refer to Tab. 1.

Intuitiveness Trials In order to obtain the participants' subjective impression of the helpfulness of maximally activating images, we add trials at the beginning of the experiments, and also at the end of Experiment II. The task set-up is slightly different (see Fig. 14): Only maximally activating (i.e. no minimally activating) images are shown. We ask participants to rate how intuitive they find the explanation of the entirety of the synthetic images for the entirety of the natural images. Again, all images presented in one trial are specific to one feature map. By moving a slider to the right (left), participants judge the explanation method as intuitive (not intuitive). The ratings are recorded on a continuous scale from -100 (not intuitive) to $+100$ (intuitive). All participants see the same three trials in a randomized order. The trials are again taken from the hand-picked (i.e. interpretable) feature maps of the appendix overview in Olah et al. (2017). In theory, this again allows for the highest intuitiveness ratings possible. The specific feature maps are from a low, intermediate and high layer: feature map 43 of mixed3a, feature map 504 of mixed4b and feature map 17 of mixed 5b.

Participants Our two experiments are within-subject studies, meaning that every participant answers trials for all conditions. This design choice allows us to test fewer participants. In Experiment I, 10 expert participants take part (7 male, 3 female, age: 27.2 years, SD = 1.75). In Experiment II, 23 participants take part (of which 10 are experts; 14 male, 9 female, age: 28.1 years, SD = 6.76). Expert participants qualify by being familiar or having worked with convolutional neural networks and most of them even with feature visualization techniques. All participants are naive with respect to the aim of the study. Expert (lay) participants are paid 15€ (10 €), per hour for participation. Before the experiment, all participants give written informed consent for participating. All participants have normal or corrected to normal vision. All procedures conform to Standard

8 of the American Psychological 405 Association's "Ethical Principles of Psychologists and Code of Conduct" (2016). Before the experiment, the first author explains the task to each participant and ensures complete understanding. For lay participants, the explanation is simplified: Maximally (minimally) activating images are called "favorite images" ("non-favorite images") of a "computer program" and the question is explained as which of the two query images would also be a "favorite" image to the computer program.

Apparatus Stimuli are displayed on a VIEWPixx 3D LCD (VPIXX Technologies; spatial resolution 1920×1080 px, temporal resolution $120\,\mathrm{Hz}$). Outside the stimulus image, the monitor is set to mean gray. Participants view the display from $60\,\mathrm{cm}$ (maintained via a chinrest) in a darkened chamber. At this distance, pixels subtend approximately $0.024°$ degrees on average ($41\,\mathrm{ps}$ per degree of visual angle). Stimulus presentation and data collection is controlled via a desktop computer (Intel Core i5-4460 CPU, AMD Radeon R9 380 GPU) running Ubuntu Linux (16.04 LTS), using PsychoPy (Peirce et al., 2019, version 3.0) under Python 3.6.

A.1.2 Stimuli Selection

Model Following the existing work on feature visualization by Olah et al. (2017; 2018; 2020b;a), we use an Inception V1 network[6] (Szegedy et al., 2015) trained on ImageNet (Deng et al., 2009; Russakovsky et al., 2015). Note that the Inception V1 network used in previously mentioned work slightly deviates from the original network architecture: The 3×3 branch of Inception module mixed4a only holds 204 instead of 208 feature maps. To stay as close as possible to the aforementioned work, we also use their implementation and trained weights of the network[7]. We investigate feature visualizations for all branches (i.e. kernel sizes) of the Inception modules and sample from layers mixed3a to mixed5b before the ReLU non-linearity.

Synthethic Images from Feature Visualization The synthetic images throughout this study are the optimization results of the feature visualization method from Olah et al. (2017). We use the channel objective to find synthetic stimuli that maximally (minimally) activate the spatial mean of a given feature map of the network. We perform the optimization using lucid 0.3.8 and TensorFlow 1.15.0 (Abadi et al., 2015) and use the hyperparameter as specified in Olah et al. (2017). For the experimental conditions with more than one minimally/maximally activating reference image, we add a diversity regulariztion across the samples. In hindsight, we realized that we generated 10 synthetic images in Experiment I, even though we only needed and used 9 per feature map.

Selection of Natural Images The natural stimuli are selected from the validation set of the ImageNet ILSVRC 2012 (Russakovsky et al., 2015) dataset. To choose the maximally (minimally) activating natural stimuli for a given feature map, we perform three steps, which are illustrated in Fig. 9 and explained in the following: First, we calculate the activation of said feature map for all pre-processed images (resizing to 256×256 pixels, cropping centrally to 224×224 pixels and normalizing) and take the spatial average to get a scalar representing the excitability of the given feature map caused by the image. Second, we order the images according to the collected activation values and select the $(N_{stimuli}+1) \cdot N_{batches}$ maximally (respectively minimally) activating images. Here, $N_{stimuli}$ corresponds to the number of reference images used (either 1 or 9, see Fig. 8, **E**), the $+1$ comes from the query image, and $N_{batches} = 20$ determines the maximum number of participants we can test with our setup. Third, we distribute the selected images into $N_{stimuli}+1$ blocks. Within each block, we randomly shuffle the order of the images. Lastly, we create $N_{batches}$ batches of data by selecting one image from each of the blocks for every batch.[8]

[6] This network is considered very interpretable (Olah et al., 2018), yet other work also finds deeper networks more interpretable (Bau et al., 2017). More recent work, again, suggests that "analogous features [...] form across models [...]," i.e. that interpretable feature visualizations appear "universally" for different CNNs (Olah et al., 2020b; OpenAI, 2020).

[7] `github.com/tensorflow/lucid/tree/v0.3.8/lucid`

[8] After having performed Experiment I and II, we realized a minor bug in our code: Instead of moving every 20th image into the same batch for one participant, we moved every 10th image into the same batch for one participant. This means that we only use a total of 110 different images, instead of 200. The minimal query image is still always selected from the 20 least activating images; the maximal query image is selected from the 91st to 110th maximally activating images - and we do not use the 111th to 200th maximally activating images.

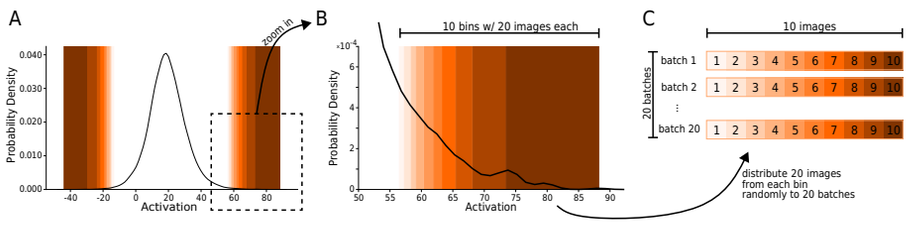


Figure 9: Sampling of natural images. **A:** Distribution of activations. For an example channel (mixed3a, kernel size 1×1, feature map 25), the smoothed distribution of activations for all $50,000$ ImageNet validation images is plotted. The natural stimuli for the experiment are taken from the tails of the distribution (shaded background). **B:** Zoomed-in tail of activations distribution. In the presentation schemes with 9 images, 10 bins with 20 images each are created (10 because of 9 reference plus 1 query image). **C:** In order to obtain 20 batches with 10 images each, the 20 images from one bin are randomly distributed to the 20 batches. This guarantees that each batch contains a fair selection of extremely activating images. The query images are *always* sampled from the most extreme bins in order to give the best signal possible. In the case of the presentation schemes with 1 reference image, the number of bins in B is reduced to 2 and the number of images per batch in C is also reduced to 2.

Subject	**Order of presentation schemes (0-3) and batch-blocks (A-D)**	**Batches**		**Order of synthetic and natural**
		Practice	**Main**	
1 2 3 4	0 (A) 1 (B) 2 (C) 3 (D) 0 (B) 2 (D) 1 (C) 3 (A) 3 (B) 1 (D) 2 (A) 0 (C) 3 (C) 2 (B) 1 (A) 0 (D)		natural: 1 synthetic: 2	natural - synthetic
5 6 7 8	see subject 1-4	0	natural: 3 synthetic: 4	synthetic - natural
9 10 11 12	see subject 1-4		natural: 5 synthetic: 6	natural - synthetic
13	see subject 1-4		natural: 7 synthetic: 8	synthetic - natural

Table 1: **Counter-balancing of conditions in Experiment II.** In total, 13 naive and 10 lay participants are tested. Each "batch block" contains 20 feature maps (sampled from five layers and all Inception module branches). Batches indicate which batch number the natural query (and reference images) are taken from.

The reasons for creating several batches of extremely activating natural images are two-fold: (1) We want to get a *general* impression of the interpretability method and would like to reduce the dependence on single images, and (2) in Experiment I, a participant has to see different query images in the three different reference conditions. A downside of this design choice is an increase in variability. The precise allocation was done as follows: In Experiment I, the natural query images of the none condition were always allocated the batch with $batch\_nr = subject\_id$, the query and reference images of the natural condition were allocated the batch with $batch\_nr = subject\_id+1$, and the natural query images of the synthetic condition were allocated the batch with $batch\_nr = subject\_id+2$. The allocation scheme in Experiment II can be found in Table 1.

Selection of Feature Maps The selection of feature maps used in Experiment I is shown in Table 2; the selection of feature maps used in Experiment II is shown in Table 3.

Layer	Branch	Feature Map	Layer	Branch	Feature Map
mixed3a	1×1	25	mixed4d	1×1	95
	3×3	189		3×3	342
	5×5	197		5×5	451
	Pool	227		Pool	483
	Pool*	230		Pool*	516
mixed3b	1×1	64	mixed4e	1×1	231
	3×3	178		3×3	524
	5×5	390		5×5	656
	Pool	430		Pool	816
	Pool*	462		Pool*	809
mixed4a	1×1	68	mixed5a	1×1	229
	3×3	257		3×3	278
	5×5	427		5×5	636
	Pool	486		Pool	743
	Pool*	501		Pool*	720
mixed4b	1×1	45	mixed5b	1×1	119
	3×3	339		3×3	684
	5×5	438		5×5	844
	Pool	491		Pool	1007
	Pool*	465		Pool*	946
mixed4c	1×1	94			
	3×3	247			
	5×5	432			
	Pool	496			
	Pool*	449			

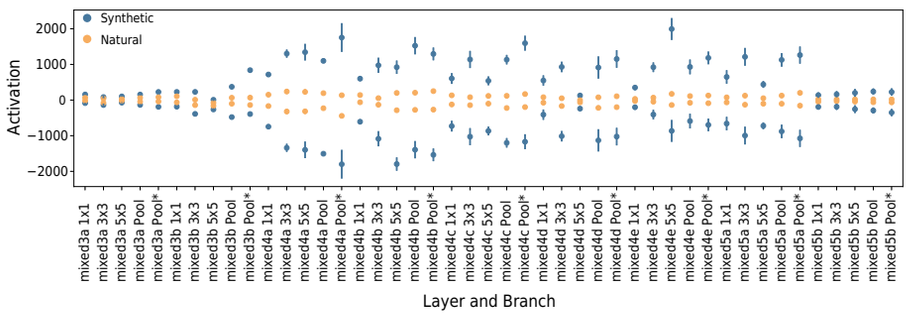
Table 2: Feature maps analyzed in Experiment I. For each of the 9 layers with an Inception module, one randomly chosen feature map per branch (1×1, 3×3, 5×5 and pool) and one additional hand-picked feature map (highlighted with *) are used.

A.1.3 Different Activation Magnitudes

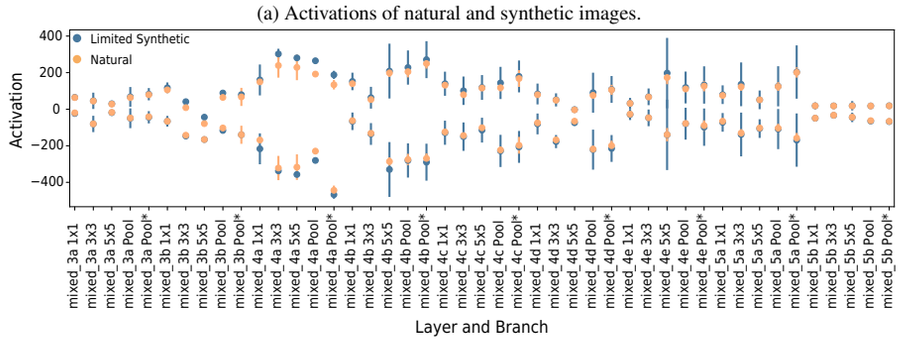
We note that the elicited activations of synthetic images are almost always about one magnitude larger than the activations of natural images (see Fig. 10a). This constitutes an inherent difference in the synthetic and natural reference image condition. A simple approach to make the two conditions more comparable is to limit the optimization process such that the resulting feature visualizations elicit activations similar to that of natural images. This can be achieved by halting the optimization process once the activations approximately match. By following that procedure one finds limited synthetic images which are indistinguishable from natural images in terms of their activations (see Fig. 10b). Importantly though, these images are visually not more similar to natural images, have a much lower color contrast than normal feature visualizations, and above all hardly resemble meaningful features (see Fig. 11).

A.1.4 Data Analysis

Significance Tests All significance tests are performed with JASP (JASP Team, 2020, version 0.13.1). For the analysis of the distribution of confidence ratings (see Fig. 3B), we use contingency tables with χ^2-tests. For testing pairwise effects in accuracy, confidence, reaction time and intuitiveness data, we report Wilcoxon signed-rank tests with uncorrected p-values (Bonferroni-corrected critical alpha values with family-wise alpha level of 0.05 reported in all figures where relevant). These non-parametric tests are preferred for these data because they do not make distributional assumptions like normally-distributed errors, as in e.g. paired t-tests. For testing marginal effects (main effects of one factor marginalizing over another) we report results from repeated measures ANOVA (RM ANOVA), which does assume normality.



(a) Activations of natural and synthetic images.



(b) Activations of natural and limited synthetic images.

Figure 10: Mean activations and standard deviations (not two standard errors of the mean!) of the minimally (below 0) and maximally (above 0) activating synthetic and natural images used in Experiment I. Note that there are 10 (i.e. accidentally not 9) synthetic images and $20 \cdot 10 = 200$ natural images (because of 20 batches) in Experiment I for both minimally and maximally activating images. Please also note that the standard deviations for the selected natural images are invisible because they are so small. Limited synthetic images refer to feature visualizations which are the result of stopping the optimization process early with the goal of matching the activation level of natural stimuli.

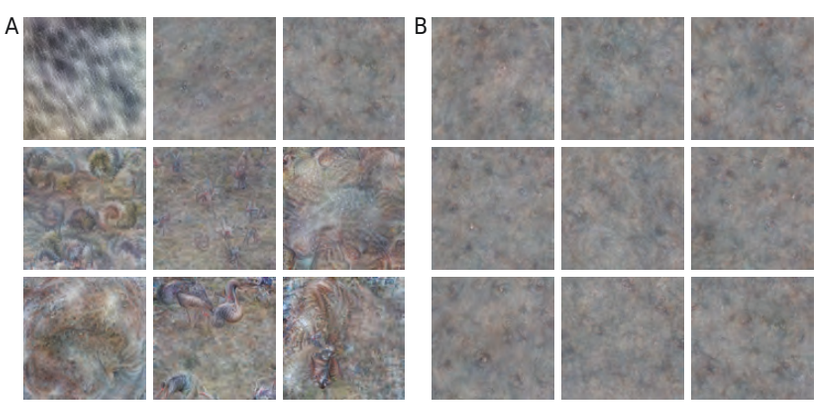


Figure 11: Limited feature visualizations, which are the result of stopping the optimization process early with the goal of matching the activation level of the chosen extreme natural stimuli. **A**: Feature visualizations for mixed_4a pool* feature map of Experiment I. **B**: Feature visualizations for all nine pool* feature maps of Experiment I.

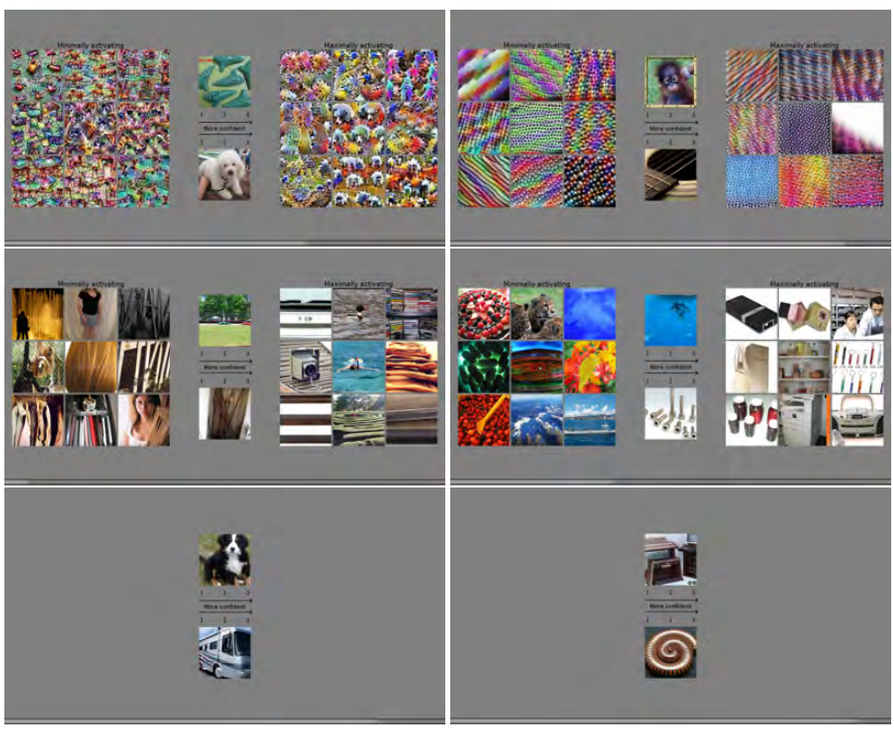


Figure 12: Experiment I: Example trials of the three reference images conditions: synthetic reference images (first row), natural reference images (second row) or no reference images (third row). The query images in the center are always natural images.

Figure 13: Experiment II: Example trials of the four presentation schemes: Max 1, Min+max 1, Max 9, Min+Max 9. The left column contains synthetic reference images, the right column contains natural reference images.

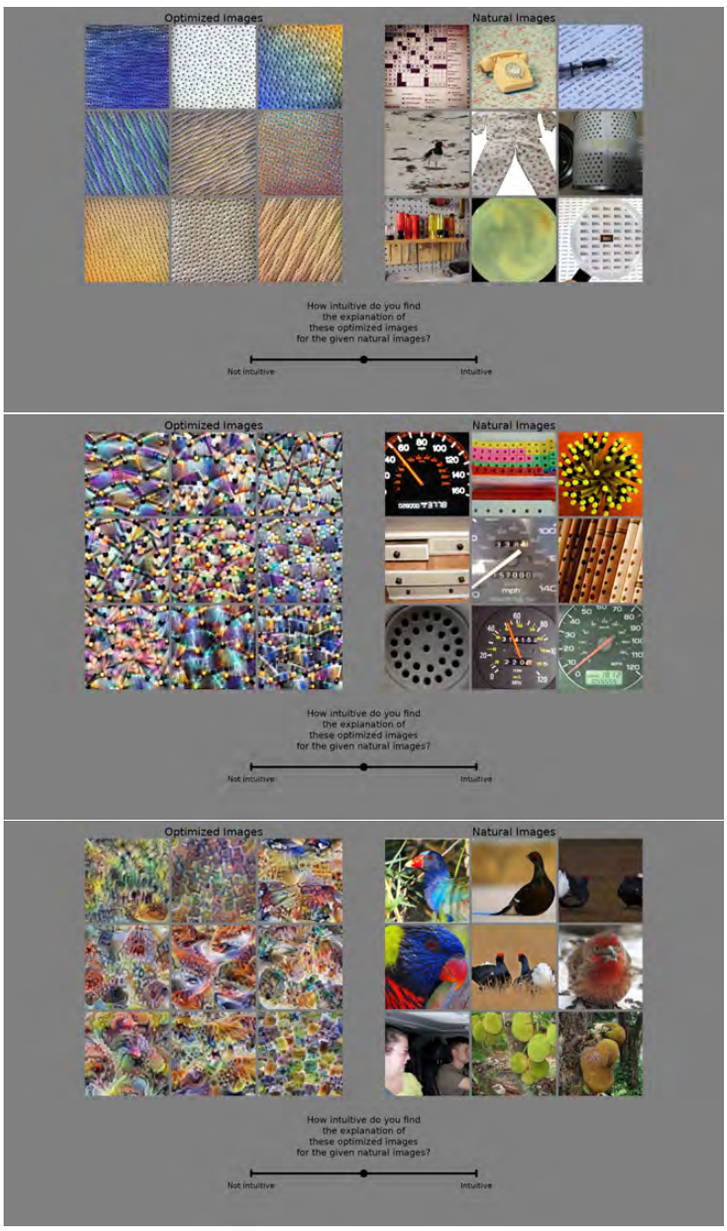


Figure 14: Trials for intuitiveness judgment. The tested feature maps are from layer mixed3a (channel 43), mixed4b (channel 504) and mixed 5b (channel 17). They are the same in Experiment I and in Experiment II.

Layer	Branch	Feature Map for Batch Block (A-D)			
		A	B	C	D
mixed3a	1×1	25	14	12	53
	3×3	189	97	171	106
	5×5	197	203	212	204
	Pool	227	238	232	247
mixed4a	1×1	68	33	45	17
	3×3	257	355	321	200
	5×5	427	425	429	423
	Pool	486	497	478	506
mixed4c	1×1	94	53	59	95
	3×3	247	237	357	209
	5×5	432	402	400	416
	Pool	496	498	473	497
mixed4e	1×1	231	83	6	89
	3×3	524	323	401	373
	5×5	656	624	642	620
	Pool	816	755	724	783
mixed5b	1×1	119	14	266	300
	3×3	684	592	657	481
	5×5	844	829	839	875
	Pool	1007	913	927	903

Table 3: Feature maps analyzed in Experiment II. Four sets of feature maps (batch blocks A to D) are sampled: For every second layer with an Inception module (5 layers in total), one feature map is randomly selected per branch of the Inception module (1×1, 3×3, 5×5 and pool). For the practice, catch and intuitiveness trials additional randomly chosen feature maps are used.

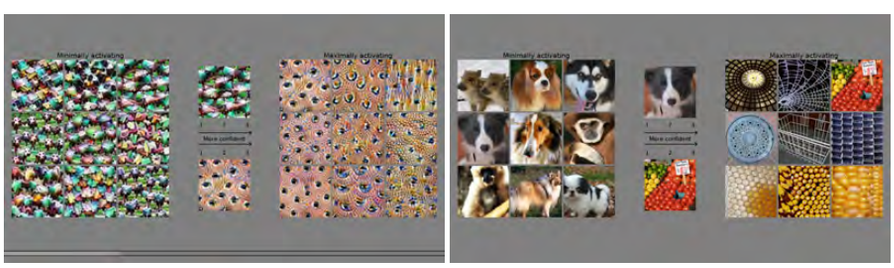


Figure 15: Catch trials. An image from the reference images is copied as a query image, which makes the answer obvious. The purpose of these trials is to integrate a mechanism into the experiment which allows us to check post-hoc whether a participant was still paying attention.

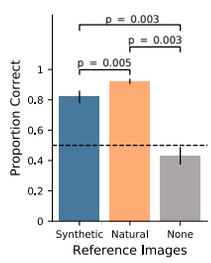
A.2 Details on results

A.2.1 Complementing figures for main results

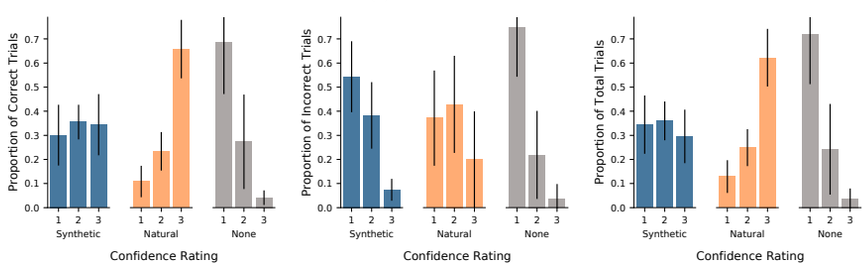
Figures 16 - 21 complement the results and figures presented in Section 4. Here, all experimental conditions are shown.

A.2.2 Details on performance of expert and lay participants

As reported in the main body of the paper, a mixed-effects ANOVA revealed no significant main effect of expert level ($F(1, 21) = 0.6$, $p = 0.44$, between-subjects effect). Further, there is no significant interaction with the reference image type ($F(1, 21) = 0.4, p = 0.53$), and both expert and lay participants show a significant main effect of the reference image type ($F(1, 21) = 230.2, p < 0.001$).



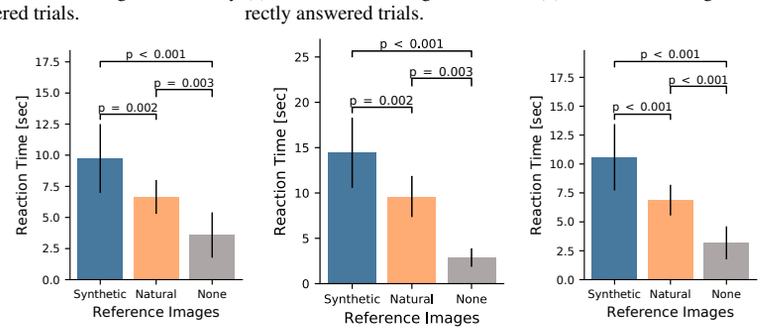
(a) Performance.



(b) Confidence ratings on correctly answered trials.

(c) Confidence ratings on incorrectly answered trials.

(d) Confidence ratings on all trials.

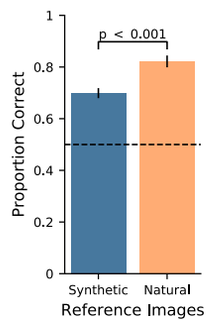


(e) Reaction time on correctly answered trials.

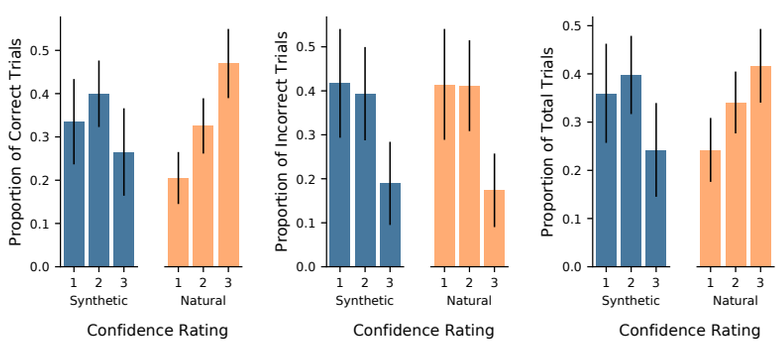
(f) Reaction time on incorrectly answered trials.

(g) Reaction time on all trials.

Figure 16: Task performance (a), distribution of confidence ratings (b-d) and reaction times (e-g) of Experiment I. The p-values are calculated with Wilcoxon sign-rank tests. Note that unlike in the main paper, these figures consistently include the "None" condition. For explanations, see Sec. 4.1.



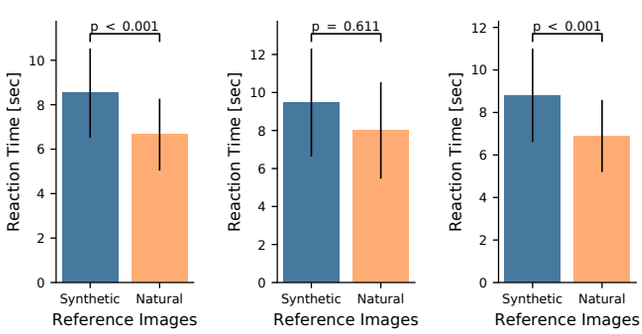
(a) Performance.



(b) Confidence ratings on correctly answered trials.

(c) Confidence ratings on incorrectly answered trials.

(d) Confidence ratings on all trials.

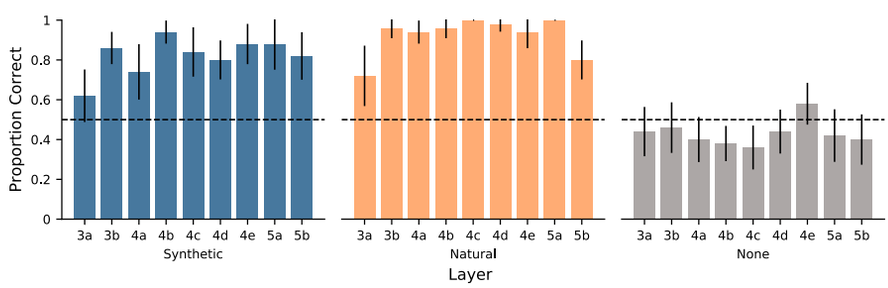


(e) Reaction time on correctly answered trials.

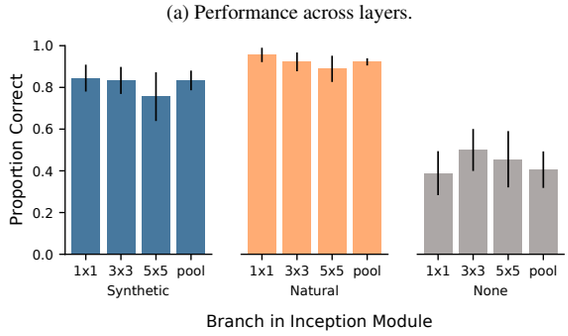
(f) Reaction time on incorrectly answered trials.

(g) Reaction time on all trials.

Figure 17: Task performance (a), distribution of confidence ratings (b-d) and reaction times (e-g) of Experiment II, averaged over expert level and presentation schemes. The p-values are calculated with Wilcoxon sign-rank tests. The results replicate our findings of Experiment I. For explanations on the latter, see Sec. 4.1.

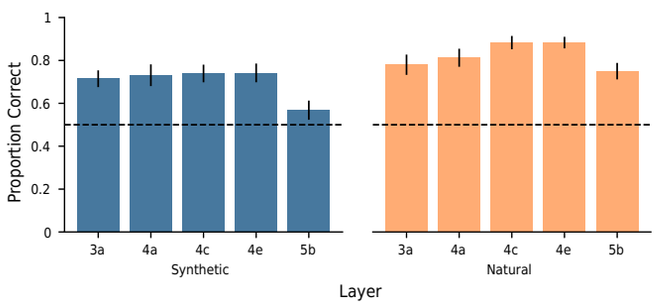


(a) Performance across layers.

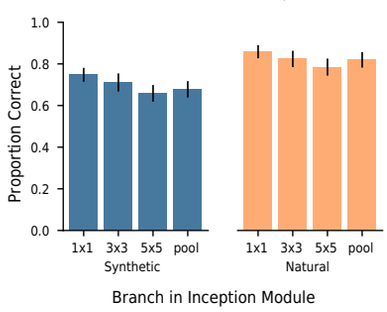


(b) Performance across branches.

Figure 18: High performance across (a) layers and (b) branches of the Inception modules in Experiment I. Note that unlike in the main paper these figures consistently include the "None" condition. For explanations, see Sec. 4.2.



(a) Performance across layers.



(b) Performance across branches in Inception module.

Figure 19: High performance across (a) layers and (b) branches of the Inception modules in Experiment II. Note that only every second layer is tested here (unlike in Experiment I). The results replicate our findings of Experiment I. For explanations, see Sec. 4.2

A.2.3 DETAILS ON PERFORMANCE OF EXPERTS SPLIT BY DIFFERENT LEVELS OF EXPERTISE

Even though Experiment II does not show a significant performance difference for lay and expert participants, it is an open question whether the level of expertise or the background of experts matters. For the data from experts, we hence further divide participants into subgroups according to their expertise (see Fig. 20a-f) and background level (see Fig. 20g-h). Expertise level 1 means that participants are familiar with CNNs, but not feature visualizations; expertise level 2 means that participants have heard of or read about feature visualizations; and expertise level 3 means that participants have used feature visualizations themselves. We note that we also accepted feature visualizations methods other than the one by Olah et al. (2017), e.g. DeepDream (Mordvintsev et al., 2015) for level 2 and 3. Regarding background, we distinguished computational neuroscientists from researchers working on computer vision and / or machine learning. We note that some subgroups only hold one participant and hence may not be representative.

Our data shows varying trends for the three expert levels (see Fig. 20a-f): For synthetic images, performance decreases with increasing expertise in Experiment I, but increases for Experiment II. For natural images, performance first increases for participants of expertise level 2, and then slightly decreases for participants with expertise level 3 - a trend that holds for both Experiment I and II. In the none condition of Experiment I, performance is highest for the participant of expertise level 1, but decreases for participants of expertise level 2, and again slightly increases for expertise level 3.

Regarding expert's different backgrounds, our hypothesis is that many of the computational neuroscientists are very familiar with maximally exciting images for monkeys or rodents, and hence might perform better than pure computer vision / machine learning experts. Fig. 20g-h suggest that this is not the case: The bars for all three reference image types are very similar.

Not finding clear trends in our data between different expertise levels or experts is not surprising as there is even no significant difference between participants whose professional backgrounds are much further apart: lay people vs. people familiar with CNNs.

A.2.4 DETAILS ON PERFORMANCE OF HAND- AND RANDOMLY-PICKED FEATURE MAPS

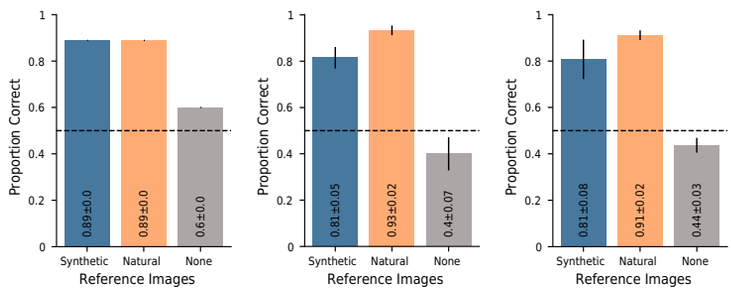
As described in the main body of the paper, pairwise Wilcoxon sign-rank tests reveal no significant differences between hand-picked and randomly-selected feature maps within each reference image type ($Z(9) = 27.5$, $p = 0.59$ for natural reference images and $Z(9) = 41$ $p = 0.18$ for synthetic references). However, marginalizing over reference image type using a repeated measures ANOVA reveals a significant main effect of the feature map selection mode: $F(1, 9) = 6.14, p = 0.035$. Therefore, while there may be a small effect of hand-picking feature maps, our data indicates that this effect, if present, is small.

A.2.5 REPEATED TRIALS

To check the consistency of participants' responses, we repeat six main trials for each of the three tested reference image types at the end of the experiment. Specifically, the six trials correspond to the three highest and three lowest absolute confidence ratings. Results are shown in Fig. 21. We observe consistency to be high for both the synthetic and natural reference image types, and moderate for no reference images (see Fig. 21A). In absolute terms, the largest increase in performance occurs for the none condition; for natural reference images there was also a small increase; for synthetic reference images, there was a slight decrease (see Fig. 21B and C). In the question session after the experiments, many participants reported remembering the repeated trials from the first time.

A.2.6 QUALITATIVE FINDINGS

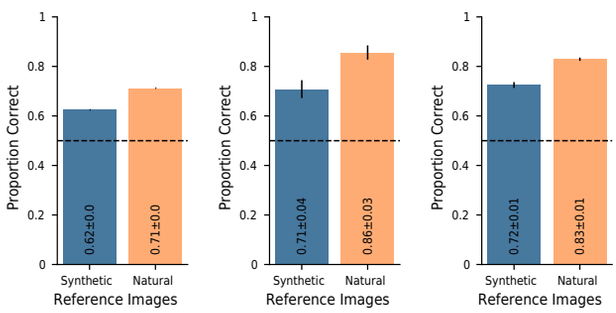
In a qualitative interview conducted after completion of the experiment, participants reported to use a large variety of strategies. Colors, edges, repeated patterns, orientations, small local structures and (small) objects were commonly mentioned. Most but not all participants reported to have adapted their decision strategy throughout the experiment. Especially lay participants from Experiment II emphasized that the trial-by-trial feedback was helpful and that it helped to learn new strategies. As already described in the main text, participants reported that the task difficulty varied greatly; while some trials were simple, others were challenging. A few participants highlighted that the comparison between minimally and maximally activating images was a crucial clue and allowed employing the



(a) Expertise level 1: one participant.

(b) Expertise level 2: six participants.

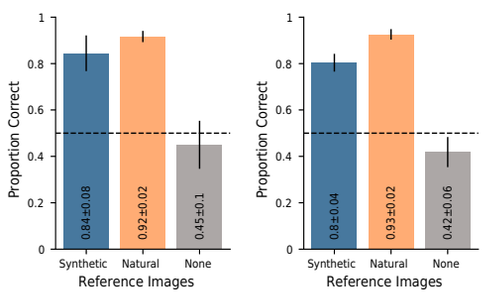
(c) Expertise level 3: three participants.



(d) Expertise level 1: one participant.

(e) Expertise level 2: six participants.

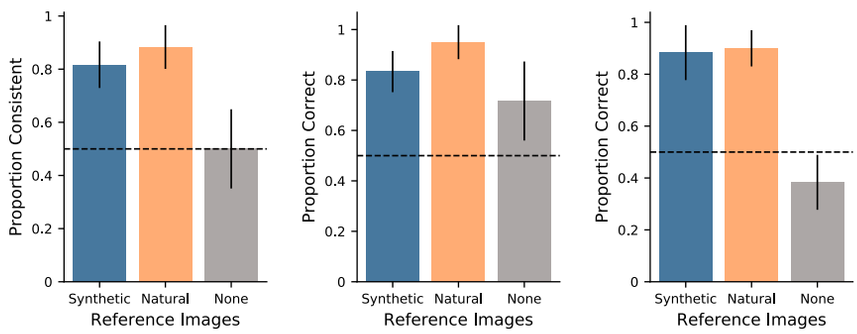
(f) Expertise level 3: three participants.



(g) Computational Neuroscience background: six participants.

(h) Computer Vision / Machine Learning background: four participants.

Figure 20: Performance of experts split by different levels of expertise: The first (second) row shows the data of Experiment I (II) split up by different levels of familiarity with CNNs and feature visualizations. The third row shows the data of Experiment I split up by different backgrounds.



(a) Proportion of trials that were answered the same upon repetition.

(b) Performance for repeated trials upon repetition.

(c) Performance for repeated trials when first shown.

Figure 21: Repeated trials in Experiment I.

exclusion criterion: If the minimally activating query image was easily identifiable, the choice of the maximally activating query image was trivial. This aspect motivated us to conduct an additional experiment where the presentation scheme was varied (Experiment II).

A.2.7 By-Feature-Map Analysis

For Experiment I, we look at each feature map separately and analyze which feature maps participants find easy and which they find difficult. Further, we investigate commonalities and differences between feature maps. We note that the data for this analysis relies on only 10 responses for each feature map and hence may be noisy.

In Fig. 22, we show the number of correct answers split up by reference image type. The patterns look similar to the trend in Fig. 4: Across most layers, there is no clearly identifiable trend that feature maps of a certain network depth would be easier or more difficult; only the lowest (3a) and the highest layer (5b) seem slightly more difficult for both the synthetic and the natural reference images.

Easy Feature Maps When feature maps are easy (synthetic: $10/10$, natural: $10/10$ correct responses), their features seem to correspond to clear object parts (e.g. dogs vs. humans, food vs. cats), or shapes (e.g. round vs. edgy (see Supplementary Material Fig. 2- 5)). In Fig. 23, we show the query as well as natural and synthetic reference images for one such easy feature map for one participant. For the images shown to two more participants, see Supplementary Material Fig. 1. Other relatively easy feature maps (where eight to ten participants choose the correct query image for both reference image types) additionally contained other low level cues such as color or texture (see Supplementary Material Fig. 4-5).

Difficult Feature Maps The most difficult feature maps for synthetic and natural reference images are displayed in Fig. 24. Only four participants predicted the correct query image. Interestingly, the other reference image type was much more easily predictable for both feature maps: Nine out of ten participants correctly simulated the network's decision. Our impression is that the reason for these feature maps being so difficult in one reference condition is the diversity in the images. In the case of synthetic reference images, we also consider identifying a concept difficult and consequently are unsure what to compare.

From studying several feature maps, our impression is that one or more of the following aspects make feature maps difficult to interpret:

- Reference images are diverse (see Fig. 24a for synthetic reference images and d for natural reference images)

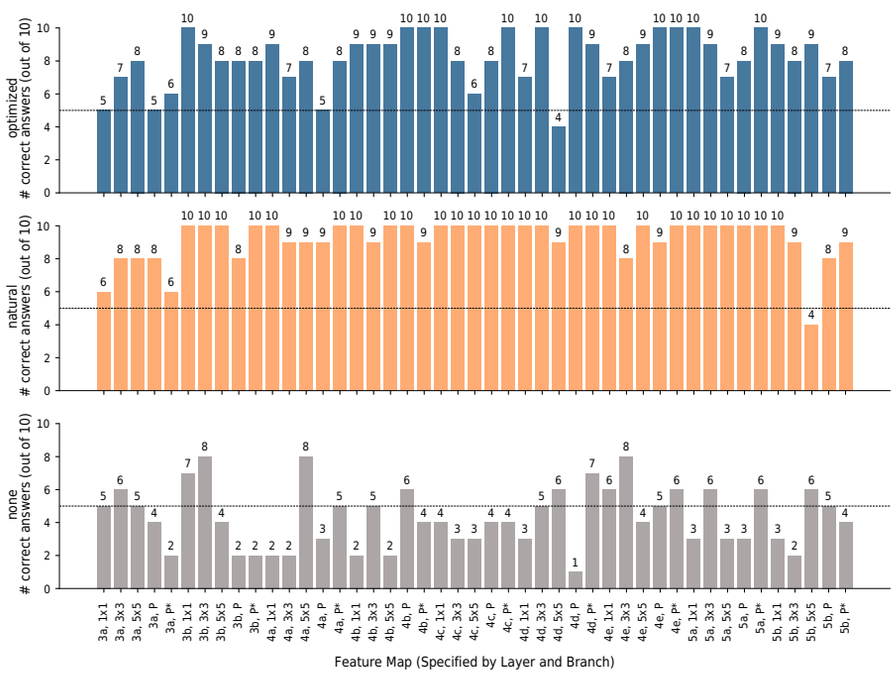


Figure 22: Data for Experiment I split up by feature maps: For each reference image type, the number of correct answers (out of ten) is shown. There is no clear trend that certain feature maps would be easier or more difficult.

Figure 23: An easy feature map (here: 5a, pool*) from Experiment I where all participants answered correctly for both synthetic and natural reference images. The shown stimuli were shown to participant 1, for stimuli shown to participant 2 and 3, see Supplementary Material Fig 1.

- The common feature(s) seem to not correspond to common human concepts (see Fig. 24a and c)
- Conflicting information, i.e. commonalities can be found between one query image and both the minimal and maximal reference images (see Fig. 25a: eyes and extremity-like structure in synthetic min reference images vs. eyes and earth-colors in synthetic max reference images - both could be considered similar to the max query image of a frog)
- Very small object parts such as eyes or round, earth-colored shapes seem to be the decisive features (see Fig. 25a and b)
- Low level cues such as the orientation of lines appear random in the synthetic reference images[9] (see Fig. 26a)

Finally, when we speak bluntly, we are often surprised that participants identified the correct image — the reasons for this are unclear to us (see for example Supplementary Material Fig. 6-7).

A.2.8 High quality data as shown by high performance on catch trials

We integrate a mechanism to probe the quality of our data: In *catch trials*, the correct answer is trivial and hence incorrect answers might suggest the exclusion of specific trial blocks (for details, see Sec. A.1.1). Fortunately, very few trials are missed: In Experiment I, only two (out of ten) participants miss one trial each (i.e. a total of 2 out of 180 catch trials were missed); in Experiment II, five participants miss one trial and four participants miss two trials (i.e. a total of 13 out of 736 catch

[9]We expected lower layers to be easier than higher layers for synthetic reference images, but our data showed that this was not the case (see Fig. 22. We can imagine that the diversity term as well as the non-custom hyper-parameters contribute to these sub-optimal images.

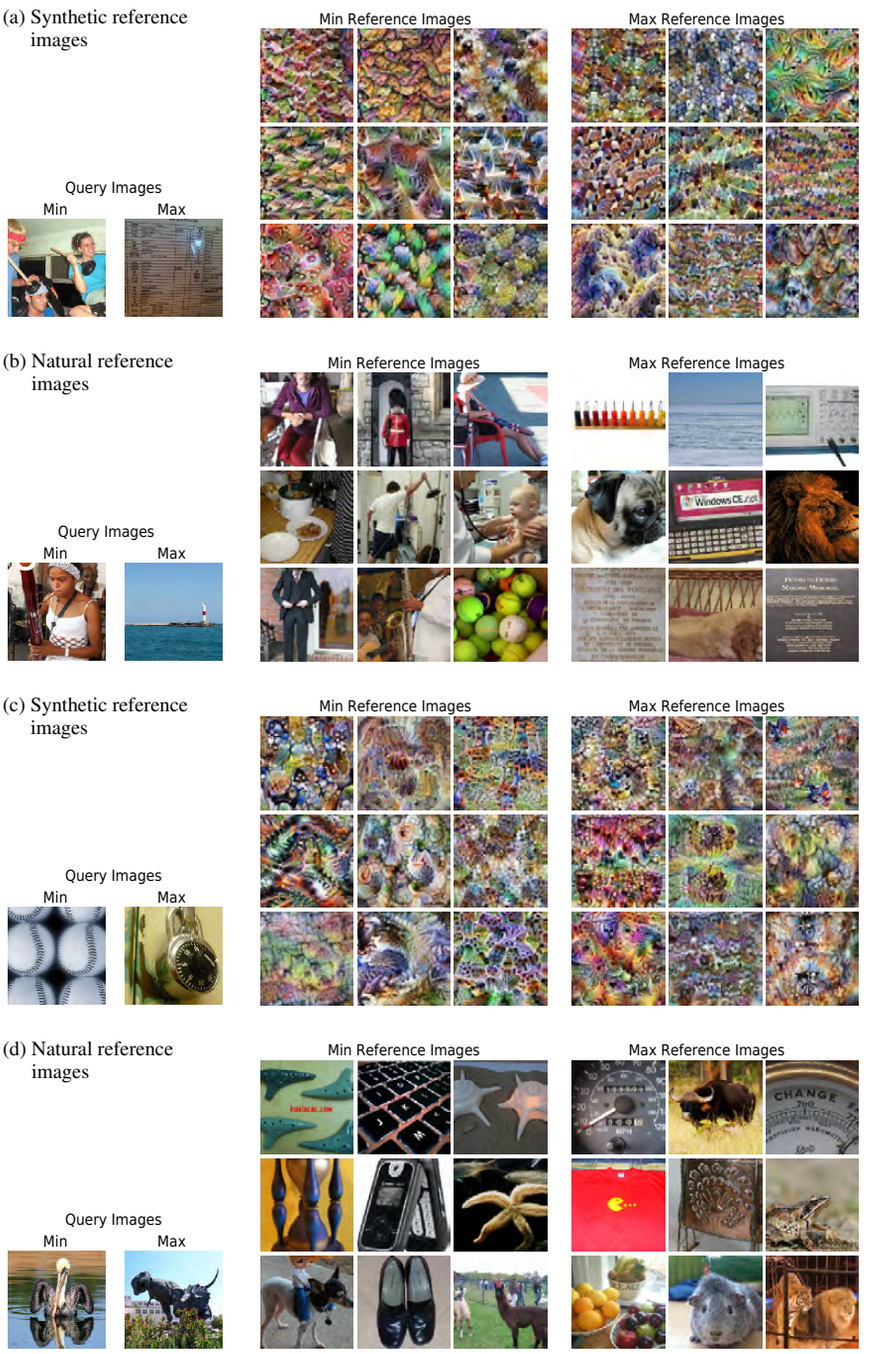


Figure 24: Two difficult feature maps (4d, 5x5 in a and b; 5b, 5x5 in c and d) from Experiment I where only four participants answered correctly for synthetic (a and b) and natural (c and d) reference images. The displayed stimuli were shown to participant 1, for stimuli shown to participant 2 (3), see Supplementary Material Fig. 8 (9).

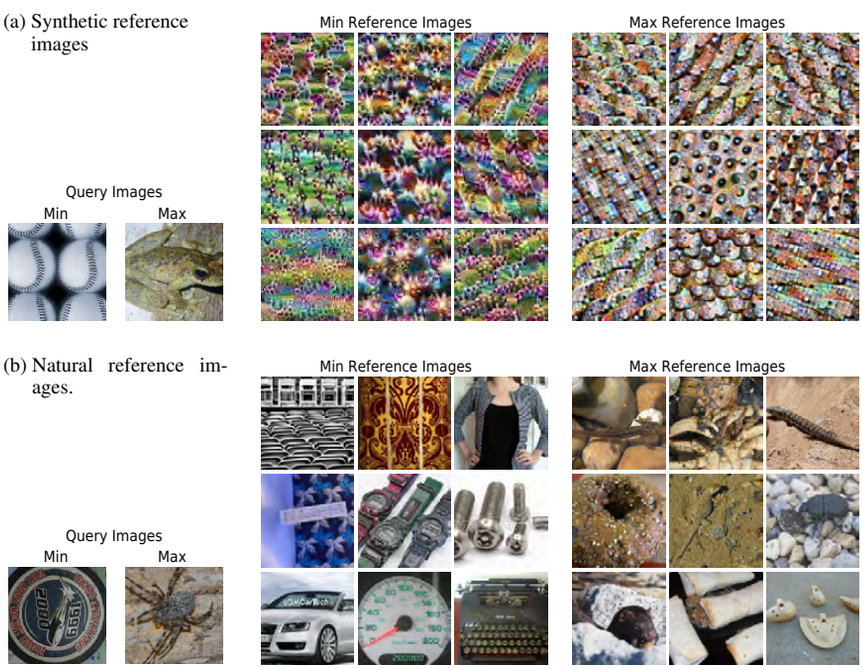


Figure 25: A feature map (here: 4a, Pool) from Experiment I where the feature is small (eyes) and a participant might perceive conflicting information (eyes and extremity-like structure in min reference images vs. eyes and earth-colors in max reference images). In this specific example, eight (nine) out of ten participants gave the correct answer for this feature map given synthetic (natural) reference images. The displayed stimuli were shown to participant 1, for stimuli shown to participant 2 and 3, see Supplementary Material Fig. 10.

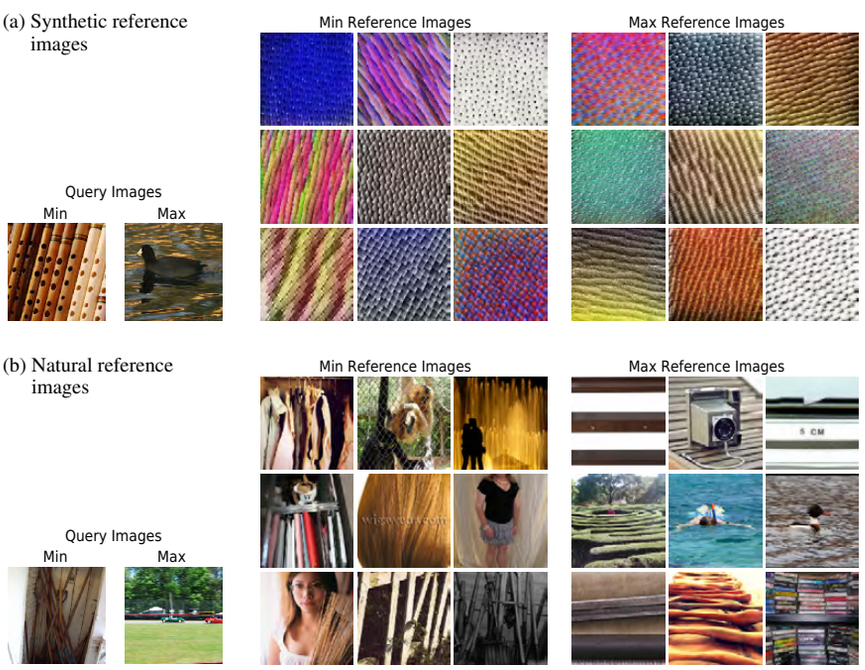


Figure 26: A feature map from a low layer (here: 3a, 3x3) from Experiment I where the feature seems to be a low level cue (horizontal vs. vertical striped) that is surprisingly clear in the natural, but surprisingly unclear in the synthetic reference images. In this specific example, seven (eight) out of ten subjects gave the correct answer for this feature map given synthetic (natural) reference images. The displayed stimuli were shown to participant 1, for stimuli shown to participant 2 and 3, see Supplementary Material Fig. 11.

trials were missed). As this indicates that our data is of high quality, we do not perform the analysis with excluded trials as we expect to find the same results.

[8] Baseline condition.

[9] Metrics of explanation quality computed without human judgment are inconclusive and do not correspond to human rankings.

[10] Task has an additional "I don't know"-option for confidence rating.

[11] Comparison is only performed between methods but no absolute measure of interpretability for a method is obtained.

A.3 DETAILS ON RELATED WORK

Paper	Analyzes Intermediate Features?	Explanation Methods Analyzed	Explanation helpful?	Results Confidence/Trust
Ours	**yes**	• Feature Visualization • natural images[8] • no explanation[8]	yes	• high variance in confidence ratings • natural images are more helpful
Biessmann & Refiano (2019)	no	• LRP • Guided Backprop • simple gradient[8]	yes	• highest confidence for guided backprop[9]
Chu et al. (2020)	no	• prediction + gradients • prediction[8] • no information[8]	no	• faulty explanations do not decrease trust
Shen & Huan (2020)	no	• Extremal Perturb • GradCAM • SmoothGrad • no explanation[8]	no	• -
Jeyakumar et al. (2020)	no	• LIME • Anchor • SHAP • Saliency Maps • Grad-CAM++ • Ex-Matchina	unclear[11]	• -
Alqaraawi et al. (2020)	no	• LRP • classification scores • no explanation[8]	yes	• confidence similar across conditions
Chandra-sekaran et al. (2017)	no	• prediction confidence • attention maps • Grad-CAM • no explanation[8]	no	• -
Schmidt & Biessmann (2019)	no	• LIME • custom method • random/no explanation[8]	yes	• humans trust own judgement regardless explanations, except in one condition
Hase & Bansal (2020)	no	• LIME • Prototype • Anchor • Decision Boundary • combination of all 4	partly	• high variance in helpfulness • helpfulness cannot predict user per-formance
Kumaraku-lasinghe et al. (2020)	no	• LIME	yes	• fairly high trust and reliance
Ribeiro et al. (2018)	no	• LIME • Anchor • no explanation[8]	yes	• high confidence for Anchor • low for LIME & no explanation
Alufaisan et al. (2020)	no	• prediction + Anchor • prediction[8] • no information[8]	partly	• explanations do not increase confidence
Ramamurthy et al. (2020)	no	• MAME • SP-LIME • Two Step	• unclear[11]	• users can adjust MAME which increased trust
Dieber & Kirrane (2020)	no	• LIME	partly	• -
Dinu et al. (2020)	no	• SHAP • ridge • lasso • random explanation[8]	partly	• no statement on confidence ratings

Paper	Experimental Setup			
	Dataset	**Task**	**Participants**	**Collected Data**
Ours	• natural images (ImageNet)	• CNN activation classification	• experts • laypeople	• decision • confidence • reaction time • post-hoc evaluation
Biessmann & Refiano (2019)	• face images (Cohn-Kanade)	• 2-way classification[10]	• laypeople	• decision • confidence • reaction time
Chu et al. (2020)	• face images (APPA-REAL)	• age regression	• laypeople	• decision • trust • reaction time • post-hoc evaluation
Shen & Huan (2020)	• natural images (ImageNet)	• model error identification	• laypeople	• decision
Jeyakumar et al. (2020)	• natural images (CIFAR-10) • text (Sentiment140) • audio (Speech Commands) • sensory data (MIT-BIH Arrhythmia)	• preference for one out of two explanation methods	• laypeople	• decision
Alqaraawi et al. (2020)	• natural images (Pascal VOC)	• classification	• technical background (neither lay nor expert)	• decision • confidence • free answer on features
Chandra-sekaran et al. (2017)	• VQA (visualqa.org)	• model error identification • regression	• laypeople	• decision
Schmidt & Biessmann (2019)	• book categories • Movie reviews (IMDb)	• 9-/2-way classification	• laypeople	• decision • reaction time • trust
Hase & Bansal (2020)	• movie reviews (Movie Review) • tabular (Adult)	• 2-way classification	• experts	• decision • helpfulness rating • explanation helpfulness
Kumaraku-lasinghe et al. (2020)	• tabular (Patient data)	• 2-way classification	• experts	• decision • feature ranking • satisfaction • questionnaire
Ribeiro et al. (2018)	• tabular (Adult, rcdv)	• 2-way classification[10] • VQA	• experts	• decision • reaction time • confidence
Alufaisan et al. (2020)	• tabular (COMPAS, Census Income)	• 2-way classification	• laypeople	• decision • confidence • reaction time
Ramamurthy et al. (2020)	• tabular (HELOC, pump failure)	• 2-way classification	• experts • laypeople	• decision
Dieber & Kirrane (2020)	• tabular (Rain in Australia)	• interview	• laypeople • experts	• how interpretable LIME output is
Dinu et al. (2020)	• tabular (Airbnb price listings)	• interview	• laypeople	• decision: which model would perform better in practice • confidence

Table 4: Overview of publications that evaluate explanation methods in human experiments. Note that the table already starts on the previous page and that the footnotes are displayed on page 39.

A.2 How Well Do Feature Visualizations Support Causal Understanding of CNN Activations?

The following 31 pages were published as:

Roland S. Zimmermann*, Judy Borowski*, Robert Geirhos, Matthias Bethge, Thomas Wallis, and Wieland Brendel. "How Well do Feature Visualizations Support Causal Understanding of CNN Activations?" *NeurIPS (2021)*

A summary is given in Section 2.1.2 on page 31. * Equal contribution.

Abstract

A precise understanding of why units in an artificial network respond to certain stimuli would constitute a big step towards explainable artificial intelligence. One widely used approach towards this goal is to visualize unit responses via activation maximization. These synthetic feature visualizations are purported to provide humans with precise information about the image features that cause a unit to be activated - an advantage over other alternatives like strongly activating natural dataset samples. If humans indeed gain causal insight from visualizations, this should enable them to predict the effect of an intervention, such as how occluding a certain patch of the image (say, a dog's head) changes a unit's activation. Here, we test this hypothesis by asking humans to decide which of two square occlusions causes a larger change to a unit's activation. Both a large-scale crowdsourced experiment and measurements with experts show that on average the extremely activating feature visualizations by Olah et al. (2017) indeed help humans on this task ($68 \pm 4\,\%$ accuracy; baseline performance without any visualizations is $60 \pm 3\,\%$). However, they do not provide any substantial advantage over other visualizations (such as e.g., dataset samples), which yield similar performance ($66 \pm 3\,\%$ to $67 \pm 3\,\%$ accuracy). Taken together, we propose an objective psychophysical task to quantify the benefit of unit-level interpretability methods for humans, and find no evidence that a widely-used feature visualization method provides humans with better "causal understanding" of unit activations than simple alternative visualizations.

How Well do Feature Visualizations Support Causal Understanding of CNN Activations?

Roland S. Zimmermann[*1] **Judy Borowski**[*1]

Robert Geirhos[1] **Matthias Bethge**[†1] **Thomas S. A. Wallis**[†2] **Wieland Brendel**[†1]

[1] Tübingen AI Center, University of Tübingen, Germany.
[2] Institute of Psychology and Centre for Cognitive Science, Technical University of Darmstadt, Germany.
[*] Shared first authorship, determined by coin flip. firstname.lastname@uni-tuebingen.de
[†] Joint supervision.

Abstract

A precise understanding of why units in an artificial network respond to certain stimuli would constitute a big step towards explainable artificial intelligence. One widely used approach towards this goal is to visualize unit responses via activation maximization. These synthetic feature visualizations are purported to provide humans with precise information about the image features that *cause* a unit to be activated — an advantage over other alternatives like strongly activating natural dataset samples. If humans indeed gain causal insight from visualizations, this should enable them to predict the effect of an intervention, such as how occluding a certain patch of the image (say, a dog's head) changes a unit's activation. Here, we test this hypothesis by asking humans to decide which of two square occlusions causes a larger change to a unit's activation. Both a large-scale crowdsourced experiment and measurements with experts show that on average the extremely activating feature visualizations by Olah et al. [40] indeed help humans on this task ($68 \pm 4\,\%$ accuracy; baseline performance without any visualizations is $60 \pm 3\,\%$). However, they do not provide any substantial advantage over other visualizations (such as e.g. dataset samples), which yield similar performance ($66 \pm 3\,\%$ to $67 \pm 3\,\%$ accuracy). Taken together, we propose an objective psychophysical task to quantify the benefit of unit-level interpretability methods for humans, and find no evidence that a widely-used feature visualization method provides humans with better "causal understanding" of unit activations than simple alternative visualizations.

1 Introduction

It is hard to trust a black-box algorithm, and it is hard to deploy an algorithm if one does not trust its output. Many of today's best-performing machine learning models, deep convolutional neural networks (CNNs), are also among the most mysterious ones with regards to their internal information processing. CNNs typically consist of dozens of layers with hundreds or thousands of units that distributively process and aggregate information until they reach their final decision at the topmost layer. Shedding light onto the inner workings of deep convolutional neural networks has been a long-standing quest that has so far produced more questions than answers.

One of the most popular tools for explaining the behavior of individual network units is to visualize unit responses via activation maximization [16, 33, 38, 35, 39, 36, 54, 15]. The idea is to start with an image (typically random noise) and iteratively change pixel values to maximize the activation

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

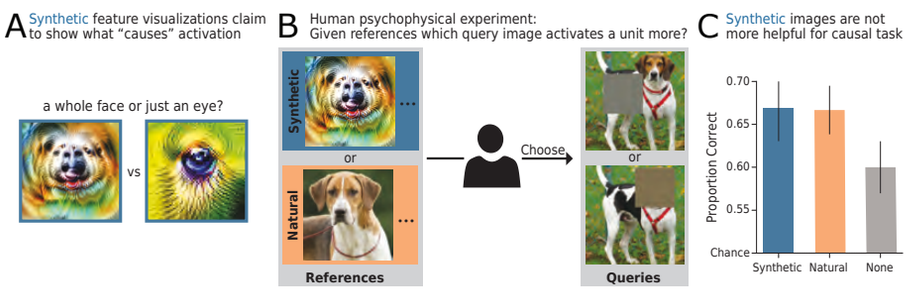


Figure 1: How useful are feature visualizations to interpret the effects of interventions? **A: "Causal" synthetic feature visualizations. B: Human experiment.** Given strongly activating reference images (e.g. synthetic or natural), a human participant chooses which out of two manipulated images activates a unit more. Note that this trial is made up — real trials are often more difficult. **C: Core result.** While participants are above chance for all visualization types, synthetic images only provide a substantial advantage over *no* references and not over other alternatives such as natural references.

of a particular network unit via gradient ascent. The resulting synthetic images, called *feature visualizations*, often show interpretable structures, and are believed to isolate and highlight exactly those features that "cause" a unit's response [40, 50]. Some of the synthetic feature visualizations appear quite intuitive and precise. As shown in Fig. 1A, they might facilitate distinguishing whether, for example, a unit responds to just an eye or a whole dog's face.

However, other aspects cast a more critical light on feature visualization's "causality": Generating these synthetic images typically involves regularization mechanisms [36, 33, 38, 35], which may influence how faithfully they visualize what "causes" a network unit's activation. Furthermore, to obtain a complete description of a mathematical function, one generally needs more information than just knowing its extrema. In view of this, it is an open question how well a unit can be characterized by simply visualizing the arguments of its maxima. Finally, a crucial unknown factor is whether *humans* are able to obtain a causal understanding of CNN activations from these synthetic visualizations.

Given these points, we develop a psychophysical experiment to test whether feature visualizations by Olah et al. [40] indeed allow humans to gain a causal understanding of a unit's behavior. Our task is based on the reasoning that being able to predict the effect of an intervention is at the heart of causal understanding. Understanding the causal relation between variables implies an understanding of how changes in one variable affect another one [45]. In our proposed experiment, this means that participants can predict the effect of an intervention — in form of an image manipulation — if they know the causal relation between image features and a unit's activations. Our experiment tests whether synthetic feature visualizations indeed provide information about such causal relations. Specifically, we ask humans which of two manipulated images activates a CNN unit more strongly. The interventions we test are obtained by placing an occlusion patch at two different locations in an image. Taken together, this experiment probes the purported explanation method's advantage of causality in a counterfactual-inspired prediction set-up [14].

Besides feature visualizations, other visualization methods have been used to gain an understanding of the inner workings of CNNs. In this experiment, we additionally test alternatives based on natural dataset examples and compare them with feature visualizations. This is particularly interesting because dataset examples are often assumed to provide less "causal" information about a unit's response as they might contain misleading correlations [40]. To continue the example above, dog eyes usually co-occur with dog faces; thus, separating the influence of one image feature from the other one using natural exemplars might be challenging.

Our data shows that:

- Synthetic feature visualizations provide humans with some helpful information about the most important patch in an image — but not much more information than no visualizations at all.
- Dataset samples as well as other combinations and types of visualizations are similarly helpful.
- How easily the most important patch is identifiable depends on the unit, the images as well as the relative activation strength attributed to the patch.

2 Related Work

Feature visualizations are a widely used method to understand the learned representations and decision-making mechanisms of CNNs [33, 38, 35, 39, 36, 54, 15, 40, 37]. As such, several works leverage this method to study InceptionV1 [42, 41, 8, 43, 50, 9, 58, 59, 46] and other networks [6, 21, 20]; others create interactive tools [61, 44, 52] or introduce analysis frameworks [65]. In contrast, some researchers question whether this synthetic visualization technique, first introduced by Erhan et al. [16], is too intuition-driven [27], and how representative the appealing visualizations in publications are [26]. Further, as already mentioned above, the engineering of the loss function may influence their faithfulness [36, 33, 38, 35]. Another challenge is generating *diverse* feature visualizations to represent the different aspects that one single unit may respond to [42, 36]. Finally, our recent human evaluation study [5] found that while these synthetic images do provide humans with helpful information in a forward simulation-inspired task, simple natural dataset examples are even more helpful.

Human evaluation studies are extensively used to quantify various aspects of interpretability. As an alternative to pure mathematical approximations [2, 66, 57, 63], researchers not only evaluate the understandability of explanation methods in psychophysical studies [7, 34, 5], but also trust in these methods [28, 64]) as well as the human cognitive load necessary for parsing explanations [1] or whether humans would follow an explained model decision [47, 13, 48]. A recent study even demonstrates that metrics of the explanation quality computed *with* human judgment are more insightful than those without [4].

Counterfactuals are a popular paradigm for both *creating* as well as *evaluating* explanation methods. Intuitively, they provide answers to the question "what should I change to achieve a different outcome?" — in the context of machine learning explanation methods, usually the smallest, realistic change to a data point is of interest. As examples, counterfactual explanation methods have been developed for vision- [22] and language-based [62] models as well as for model-agnostic scenarios [51]. Further, they are set into context of the EU General Data Protection Regulation [60]. Ustun et al. [56] investigate feasible and least-cost counterfactuals, while Mahajan et al. [32] and Karimi et al. [25] take feature interactions into account. To *evaluate* — rather than create — explanation methods, researchers often follow the "counterfactual simulation" task introduced by Doshi-Velez and Kim [14]: Humans are given an input, an output, and an explanation and are then asked "what must be changed to change the method's [model's] prediction to a desired output?" Doshi-Velez and Kim [14]. Based on this task, Lucic et al. [30] test their new explanation method and Hase and Bansal [24] compare different explanation methods to each other.

Figure 2: **Schematic visualization of an example trial** in our psychophysical experiment. For a certain network unit, participants are shown several maximally activating images. While the ones on the left serve as reference images, the ones on the right serve as query images: The top one is a natural maximally activating image and the bottom ones are copies of said image with square occlusions at different locations. The task is to select the image that activates the given network unit more strongly. Participants answer by clicking on the number below the corresponding image according to their confidence level (1: not confident, 2: somewhat confident, 3: very confident). Correct answer: right image.

In this project, we design a counterfactual-inspired task to evaluate how well feature visualizations support causal understanding of CNN activations. This is the first study to apply such a paradigm to understanding the causes of individual units' activations. In order to scale the experiments, we

simplify our task by having participants choose between two intervention *options*, rather than having them freely determine interventions themselves.

3 Methods

We run an extensive psychophysical experiment with more than $12,000$ trials distributed over 323 crowdsourced participants on Amazon Mechanical Turk (MTurk) and two experts (the two first authors).[1] For more details than provided below, please see Appx. Sec. A.1.

Design Principles Overall, our experimental design choices aim at (1) the *best performance possible*, meaning that we select images that make the signal as clear as possible; (2) *generality* over the network, meaning that we randomly sample units of different layers and branches (testing all units would be too costly); and (3) *easy extendability*, meaning that we choose a between-participant design (each participant sees only one reference image condition) so that other visualizations methods can be added to the comparisons in the future.

3.1 Psychophysical Task

If feature visualizations indeed support causal understanding of CNN activations, this should enable humans to predict the effect of an intervention, such as how occluding an image region changes a unit's activation. Based on this idea, we employ a two-alternative forced choice task (chance performance: 50%) where human observers are presented with two different occlusions in an image, and asked to estimate which of them causes a smaller change to the given unit's activation (see Fig. 2 for an example trial). More specifically, participants choose the *query* image that they believe to also elicit a strong activation given a set of 9 *reference* images. Such references could for instance consist of synthetic feature visualizations of a certain unit (purportedly "causal"), or alternative visualizations. To summarize, the task requires humans to first identify the shared aspect in the reference images and to then choose the query image in which that aspect is more visible. Since we do not make any assumptions about whether participants are familiar with machine learning, we avoid asking participants about activations of a unit in the CNN. Instead, we explain that an image would be "favored" by a machine, and the task is to select the image which is "more favored". The complete set of instructions shown to participants can be found in Appx. Fig. 9 and 10. In addition to each participant's image choice, the subjective confidence level and reaction time are also recorded.

3.2 Stimulus Generation

To generate stimuli, we follow Olah et al. [40] and use an InceptionV1 network [53] trained on ImageNet [12, 49]. Throughout this paper, we refer to a CNN's channel as a "unit" and imply taking the spatial average of all neurons in one channel.[2] We test units sampled from 9 layers and 2 Inception module branches (namely 3×3 and POOL). For more details on the generation procedures of the respective stimuli, see Appx. A.1.2.

We use five different types of **reference images**:

- **Synthetic references**: The synthetic images are the optimization results of the feature visualization method by Olah et al. [40] with the channel objective for 9 diverse images.
- **Natural references**: The reference images are the most strongly activating[3] dataset samples from ImageNet [12, 49].
- **Mixed references**: This is a combination of the previous two conditions: the 5 most strongly activating natural and 4 synthetic reference images are used. The motivation is that this condition combines the advantages of both worlds — namely precise information from feature visualizations and easily understandable natural images — and, thus, has the potential to give rise to higher performance in the task. Jointly looking at these two visualization types is common in practice [40].

[1] Code and data are available at github.com/brendel-group/causal-understanding-via-visualizations.

[2] Other papers might refer to a channel as a "feature map", e.g. [5].

[3] To reduce compute requirements, we use a random subset of the training set ($\approx 50\%$).

- **Blurred references**: To increase the informativeness of natural images for this task, we modify them by blurring everything but a single patch. This patch is chosen in the same way as in the maximally activating query image (see below). Consequently, this method cues participants to the most important image feature. In a way, these images can be seen as an approximate inverse of the maximally activating query image and might improve performance on our task.
- **No references**: This is a control condition in which participants do not see any reference images and have to solve the task purely based on query images.

To generate **query images**, we place a square patch of 90×90 pixels of the average RGB color of the occluded pixels into a most strongly activating image chosen from ImageNet. The location of the occlusion patch is chosen such that the activation of the manipulated image is either minimal or maximal among all possible occlusion locations. These images then yield the distractor and target query images respectively.

3.3 Structure of the Psychophysical Experiment

We test the five different reference image types as separate experimental conditions. In each condition, we collect data from a total of 50 different MTurk participants, each assigned to a single Human Intelligence Task (HIT) consisting of an instruction block, a variable number of practice blocks and a main block. The instructions extensively explain a hand-crafted example trial (see Appx. Fig. 9 and 10). The blocks of 4 practice trials each - which are randomly sampled from a pool of 10 trials - have to be repeated until reaching 100% performance; except in the none condition, as there is no obvious ground truth due to the absence of reference images. Finally, 18 main trials follow that are randomly interleaved with a total of 3 obvious catch trials. While feedback is provided during practice trials, no feedback is provided in the other trials. At the end, participants can share comments via an optional free-text field. Across all conditions, all participants see the same query images for the instruction, practice and catch trials. In contrast, the query images differ across participants in the main trials: In each reference image condition, we test 10 different sets of query images, each responded to by 5 different MTurk participants, hence 50 HITs per condition. The order of the main and catch trials per participant is randomly arranged, and identical across conditions. Each MTurk participant takes part in only one reference image condition (i.e. reference images are a between-participants factor). For more details, see Appx. Sec. A.1.4.

3.4 Ensuring High-Quality Data in an Online Experiment

To ensure that the data we collect in our online experiment is of high quality, we take two measures: (1) We integrate hidden checks which were set before data collection. Only if a participant passes all five of them do we include his/her data in our analysis. First, these *exclusion criteria* comprise a performance threshold on the practice trials as well as a maximum number of blocks a participant may attempt. Further, they include a performance threshold for catch trials, a minimum image choice variability as well as a minimum time spent on both the instructions and the whole experiment. For more details, see Appx. Sec. A.1.1. (2) Our previous human evaluation study in a well-controlled lab environment found that natural reference images are more informative than synthetic feature visualizations when choosing which of two different images is more highly activating for a given unit [5]. We replicate this main finding on MTurk based on a subset of the originally tested units (see Appx. A.3) which indicates that the experiment's environment does not influence this task's outcome. Our decision to leverage a crowdsourcing platform is further corroborated by our result in Borowski et al. [5], that there is no significant difference between expert and lay performance.

3.5 Baselines

In order to both set MTurk participants' performance into context as well as evaluate different strategies participants could use to perform our task, we further evaluate a few baselines.

- **Expert Baseline**: The two first authors answer all 18 trials in all 5 reference conditions on all 10 image sets. As they are familiar with the task design and are certainly engaged, this data serves as an upper human bound.
- **Center Baseline**: In natural images from ImageNet, important objects are likely to be closer to the center of the image. If participants were biased to assume that units respond to *objects*, a potential

strategy to decide which occluding patch produces a smaller effect on the unit's activation would therefore be to choose the image with the most eccentric occlusion. The Center Baseline model performs this strategy for all images.

- **Primary Object Baseline**: The Center Baseline is not a perfect measurement of an object-biased strategy because primary objects can appear away from the center. To account for this, the two first authors and the last author manually label all trials, choosing the image for which the occlusion hides as little information as possible from the most prominent object in the scene. In approximately one third of the trials ($58/180$), the authors' confidence ratings are very low (reflecting e.g. the absence of a primary object); in these cases we repeatedly replace the decisions by random binomial choices. Thus, in the results, we report the estimated expected values, but cannot perform a by-trial analysis. For more details, see Appx. Sec. A.1.3.
- **Variance Baseline**: Another assumption participants might make is that a patch in a low-contrast region, e.g. a blue sky, is unlikely to have a large effect on the unit's activation. This baseline selects the query image whose content is less affected by the introduction of the occlusion patch. To simulate this, we calculate the standard deviation over the occluded pixels and choose the one of the lower standard deviation.
- **Saliency Baseline**: As a complement to the baselines above, this baseline selects the query image whose original pixels hidden by the occlusion patch have a lower probability of being looked at by the participants. This simulates that participants select the image with a patch that occludes less prominent information and is estimated with the saliency prediction model DeepGaze IIE [29]. For more details, see Appx. Sec. A.1.3.

4 Results

The results shown in this section are based on 7350 [4] trials from MTurk participants, who passed all exclusion criteria, and experts distributed over five conditions. In all figures, *Synthetic* refers to the purportedly "causal", activation-maximizing feature visualizations, *Natural* to ImageNet samples, *Mixed* to the combined presentation of synthetic and natural images, *Blur* to the blurred images, and *None* to the condition with no reference images at all. Further, error bars indicate two standard errors above and below the participant-mean over network units and image sets, unless stated otherwise.

4.1 No Significant Advantage of Synthetic Feature Visualizations

If feature visualizations provide humans with useful information about the image features causing high unit activations and other visualizations do not, participants' accuracy in our task should be higher given feature visualizations than for all other visualization types or no reference images. This is only partly what we find: On average, accuracy for feature visualizations is slightly higher than when no reference images are given ($67 \pm 4\%$ vs. $60 \pm 3\%$). However, the accuracy for feature visualizations is not significantly higher than for other visualization methods (see Fig. 3A, dark bars). For the latter, MTurk participants reach between $66 \pm 3\,\%$ and $67 \pm 5\,\%$ depending on the visualization type. Statistically, only the condition without reference images is

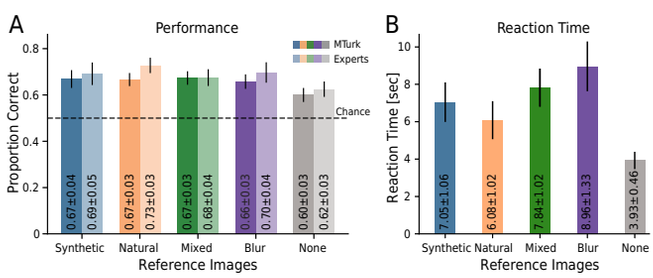


Figure 3: **A: Task accuracy.** On average, humans reach the same performance regime with any visualization method. This holds for both lay participants on MTurk (darker colors) as well as experts (brighter colors). **B: Reaction times.** MTurk participants need several seconds to answer a trial, indicating that they carefully make their decision. For more details see Appx. Fig. 13.

[4] (18 main + 3 catch trials)×50 MTurk participants ×5 conditions + (18 main + 3 catch trials)×20 expert measurements ×5 conditions.

different from all other conditions ($p < 0.05$, Mann-Whitney U test). Taken together, these findings suggest that all visualization methods are similarly helpful for humans in our counterfactual-inspired task, and that they only seem to offer a small improvement over no visualizations at all.

4.1.1 MTurk Participants Carefully Make Their Choices

Similar performances for various conditions such as those found in Fig. 3A might suggest that participants would not give their best when doing our experiment. However, several aspects speak against this: (1) Measurement of the two first authors, i.e. experts who designed and thus clearly understand the task, and certainly engage during the experiment, again show very similar performance (see Fig. 3A, bright bars): This estimated upper bound is just $1 - 6\%$ better than MTurk participant performance. (2) With our strict exclusion criteria, we check for doubtful participant behavior and only include data from participants who pass all five criteria. (3) Reaction times per trial (see Fig. 3B) lie between $\approx 4\,\mathrm{s}$ and $\approx 9\,\mathrm{s}$. This, as well as the fact that participants take longer for the conditions *with* references than for the *None* condition, suggest that they carefully make their decisions. (4) Several MTurk participants' comments in an optional free-text field indicate that they engage in the task: "[...] I did my best", "It was engaging", "interesting task". (5) Trial-by-trial responses between MTurk participants are more similar than expected by chance (see Fig. 4B discussed below), which suggests that humans use the available information.

4.1.2 Simple Baselines Can Reach the Same Above-Chance Performance Regime

Decision-making strategies can be diverse. To set human performance into context, we evaluate several simple strategies as baselines: How high is performance if one always chooses the query image with an unoccluded center (Center Baseline) or primary object (Object Baseline)? Or such that the more varying or salient image region is unoccluded (Variance and Saliency Baseline)? Fig. 4A shows that these strategies have varying performances with the best ones — namely the Object and Variance baselines — reaching $63 \pm 1\,\%$ and $63\,\%$, respectively. Since already these simple heuristics, which do not require reference visualizations, can reach the same performance regime as participants, the additional advantage of visualizations (reaching just up to $4\,\%$ better performance) appears limited.

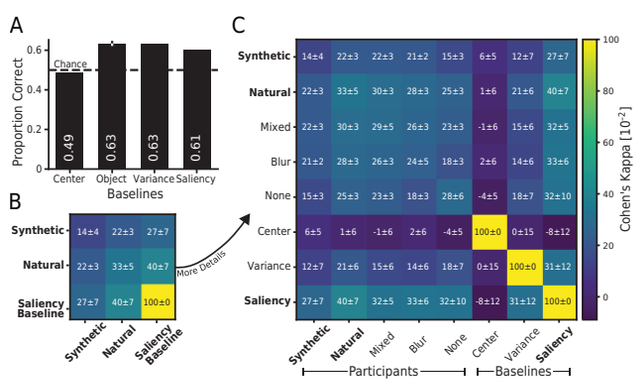


Figure 4: **A: Baseline performances.** Simple baselines can reach above chance level.[5] **B, C: Decision consistency.** The mean and two standard errors of the mean of Cohen's kappa averaged over participants and image sets quantifies the pairwise consistency of decision patterns.[6] While they vary across participants, they are higher between conditions with natural references and highest between the Saliency Baseline and other conditions. For more details, see Appx. Fig 15.

4.2 By-trial Decisions Show Systematic but Fairly Low Agreement

While accuracy is the most common metric to evaluate task performance, it does not suffice to compare two systems' decision-making processes [31, 19, 18]. Instead, a quantitative trial-by-trial error analysis is necessary to ascertain or distinguish strategies. Here, we use Cohen's kappa [10] to

[5] Only the Object Baseline has an error bar because in trials with, e.g. no clear primary object, we replace decisions by random binomial choices. The reported values are the estimated expectation value and standard deviation.

[6] There is no data for the Object Baseline because about one third of the trials do not have a clear answer from the three author responses. For more details, see Appx. A.1.3.

calculate the degree of agreement in classification while taking the expected chance agreement into account. A value of 1 corresponds to perfect agreement, while a value of 0 corresponds to as much agreement as would be expected by chance. Negative values indicate systematic disagreement.

In Fig. 4B and C, we plot consistency between MTurk participants of the same and different reference conditions as well as between MTurk participants and baselines. Since Cohen's kappa only allows for comparisons of two decision makers, we compute this statistic for all possible pairs across image sets, and report the mean over participants and image sets and two standard errors of the mean. All values between participants as well as between participants and baselines are in an intermediate regime (up to 0.40). This suggests that there is systematic agreement, but also quite some room for subjective decisions. Among participant-baseline comparisons, highest agreement is found for the saliency baseline[7] , while lowest agreement is found for the Center Baseline. Within participant to participant comparisons, decision strategies for conditions involving unmodified natural images (*Natural*, *Mixed*) are more similar to each other as well as slightly more similar to other strategies than the *Synthetic*, *Blur* or *None* condition to other strategies. Within the *Synthetic* condition, participants are relatively inconsistent. We hypothesize that due to the fact that humans are more familiar with natural images, they use more consistent information from these types of reference images and, thus, their decisions are more similar.

4.3 Performance Varies across Units, Image Sets and Activation Differences, but Less So for Reference Conditions

Having found that feature visualizations do not offer an overall advantage over other techniques, we now ask: Is performance similar across units, query images and their activation differences?

Units and Image Sets As evident from Fig. 5, performance varies by unit, but usually not much by reference condition: While only one unit (layer 2, POOL) is clearly below chance level, many units reach around average performance and a few units stand out with high performances (e.g. layer 8, POOL). Further, the five reference conditions are relatively close to each other for most units. Finally, on the image set level, we observe fairly high variance - probably partly due to the limited number of participants per image set (see Appx. Fig. 14).

Fig. 6 further illustrates the different difficulty levels as well as the strong unit- and image-dependency: For the shown easy unit (Fig. 6A), the (presumably yellow-black) feature is fairly clearly identifiable and visible in the diverse reference and query images. In contrast, for the shown difficult unit (Fig. 6B), the unit's feature selectivity is unclear not only in the reference but also in the query images.

Activation Differences We hypothesize that our task might be easier if the difference in activations between the two interventions of the query images is larger. In Fig. 7A and B, we plot by-image-set performance against the relative activation differences, i.e. the difference between activations elicited by the two manipulated images normalized by the unperturbed query image's activation. The figure shows that even though we select query images as the most strongly activating images for a unit, the relative activation differences vary widely. Furthermore, human performance indeed tends to increase with higher relative activation difference, confirming our hypothesis. This trend is stronger in the POOL than in the 3×3 branch as quantified by the Spearman's rank correlations in Fig. 7C.

5 Discussion & Conclusions

Explanation methods such as feature visualizations have been criticized as intuition-driven [27], and it is unclear whether they allow humans to gain a precise understanding of which image features "cause" high activation in a unit. Here, we propose an objective psychophysical task to quantify how well these synthetic images support causal understanding of CNN units. Through a time- and cost-intensive evaluation (based on $24,439$ trials taking more than 81 participant hours including all pilot and reported experiments), we put this widespread intuition to a quantitative test. Our data provides no evidence that humans can predict the effect of an image intervention (occlusion) particularly well when supported with feature visualizations. Instead, human performance is only moderately above a

[7] From a different perspective, this result can be seen as a confirmation that the CNN learned to look at the "important" part of the image for downstream classification.

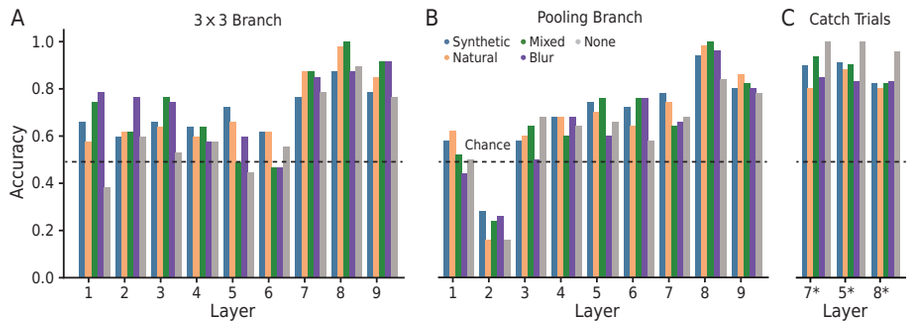


Figure 5: While for some units predicting the effect of an intervention is relatively easy, for most units performance is close to or just above chance. **A** and **B** show the **performance per unit** in the main trials separated by branch (3×3 and POOL respectively) and layer. **C** shows the performance per unit in the hand-picked trials used as catch trials (hence the *), though selected from those MTurk participants who pass the exclusion criteria without the catch trial exclusion criterion. Note that each bar represents averages over participants and image sets.

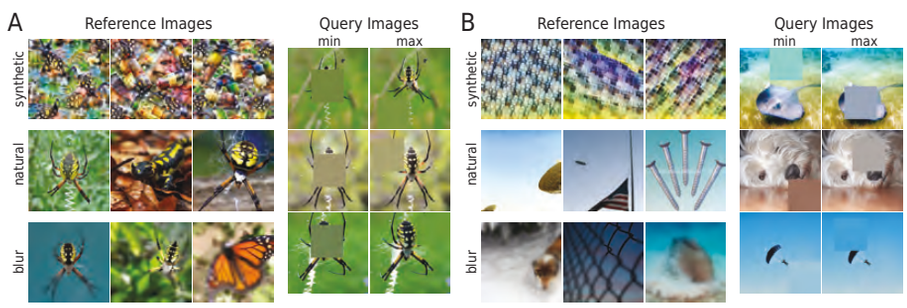
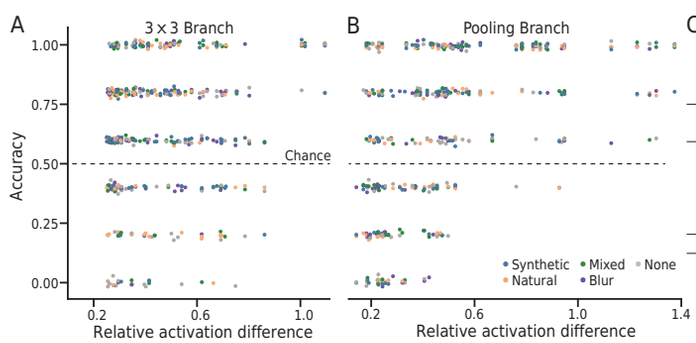


Figure 6: **Example reference and query images** for a unit with high (**A**) and low (**B**) performance from layer 8 and 2 of the POOL branch, respectively.



	Branch	
References	3×3	Pool
Synthetic	0.047	0.487*
Natural	0.162	0.447*
Mixed	0.080	0.488*
Blur	0.159	0.597*
None	0.242*	0.432*
All	0.135*	0.491*

Figure 7: Performance tends to increase with the relative activation difference between query images. This effect is stronger for the POOL branch (**B**) than for the 3×3 branch (**A**) as quantified by Spearman's rank correlations (**C**). Stars signal significance ($p < .05$). Note that each dot in A and B represents the participant-averages, i.e. there is one dot per combination of layers, branch and image set. For an alternative visualization see Appx. Fig. 16.

baseline condition where humans are not shown any visualization at all, and similar to that of other visualization methods such as simple dataset samples. Further, by-trial decisions show systematic but fairly low agreement between participants. Finally, task performance depends on the unit choice, image selections and activation differences between query images. These results add quantitative evidence against the generally-assumed usefulness of feature visualizations for understanding the causes of CNN unit activations.

Our counterfactual-inspired task is the *first* quantitative evaluation of whether feature visualizations support causal understanding of unit activations, but it is certainly not the *only* possible way to evaluate causal understanding. For example, our interventions are constrained to occlusions of a fixed size and shape, imposing an upper limit on the precision with which the occlusions can cover the part of the image that is most responsible for driving a unit's activation. Future work could explore more complex intervention techniques, extend our study to more units of InceptionV1 as well as to different networks, and investigate additional visualization methods. Thanks to the between-participant design, new conditions can be added to the data without the requirement to re-run already collected trials.

Taken together, the empirical results of our quantitative evaluation method indicate that the widely used visualization method by Olah et al. [40] does not provide causal understanding of CNN activations beyond what can be obtained from much simpler baselines. This finding is contrary to wide-spread community intuition and reinforces the importance of testing falsifiable hypotheses in the field of interpretable artificial intelligence [27]. With increasing societal applications of machine learning, the importance of feature visualizations and interpretable machine learning methods is likely to continue to increase. Therefore, it is important to develop an understanding of what we can — and cannot — expect from explainability methods. We think that human benchmarks, like the one presented in this study, help to expose a precise notion of interpretability that is quantitatively measurable and comparable to competing methods or baselines. The paradigm we developed in this work can be easily adapted to account for other notions of causality and, more generally, interpretability as well. For the future, we hope that our task will serve as a challenging test case to steer further development of feature visualizations.

Author Contributions

The idea to test how well feature visualizations support causal understanding of CNN activations was born out of several reviewer and audience comments on our previous paper [5]. The first idea of how to test this in a psychophysical experiment came from TSAW. JB led the project. JB, RSZ, WB and TSAW jointly improved the experimental set-up with input from MB and RG. RSZ led and JB helped with the implementation and execution of the experiment; JB led and RSZ contributed to the generation of stimuli. RSZ and JB both coded the baselines, and TSAW guided the replication experiment with statistical power simulations. The data analysis was performed by RSZ and JB with advice and feedback from RG, TSAW, WB and MB. TSAW and WB provided day-to-day supervision. While JB and RSZ created the first draft of the manuscript, RG and TSAW heavily edited the manuscript and all authors contributed to the final version.

Acknowledgments

We thank Felix A. Wichmann and Isabel Valera for a helpful discussion. We further thank Ludwig Schubert for information on technical details via `slack.distill.pub`. In addition, we thank our colleagues for helpful discussions, and especially Matthias Kümmerer, Dylan Paiton, Wolfram Barfuss, and Matthias Tangemann for valuable feedback on our task, and/or technical support. Moreover, we thank our various reviewers and other researchers for comments on our previous paper inspiring us to investigate causal understanding of visualization methods. And finally, we thank all our participants for taking part in our experiments.

Funding

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting JB, RSZ and RG. This work was supported by the German Federal Ministry of Education and Research (BMBF) through the Competence Center for Machine Learning (TUE.AI, FKZ 01IS18039A) and the Bernstein Computational Neuroscience Program Tübingen (FKZ 01GQ1002), the Cluster of Excellence Machine Learning: New Perspectives for Sciences (EXC2064/1), and the German Research Foundation (DFG, SFB 1233, Robust Vision: Inference Principles and Neural Mechanisms, TP3, project number 276693517). MB and WB acknowledge funding from the MICrONS program of the Intelligence Advanced Research Projects

Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003. WB acknowledges financial support via the Emmy Noether Research Group on The Role of Strong Response Consistency for Robust and Explainable Machine Vision funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1.

References

[1] Ashraf M. Abdul, Christian von der Weth, Mohan S. Kankanhalli, and Brian Y. Lim. COGAM: measuring and moderating cognitive load in machine learning model explanations. In Regina Bernhaupt, Florian 'Floyd' Mueller, David Verweij, Josh Andres, Joanna McGrenere, Andy Cockburn, Ignacio Avellino, Alix Goguey, Pernille Bjøn, Shengdong Zhao, Briane Paul Samson, and Rafal Kocielnik, editors, *CHI '20: CHI Conference on Human Factors in Computing Systems, Honolulu, HI, USA, April 25-30, 2020*, pages 1–14. ACM, 2020. doi: 10.1145/3313831.3376615.

[2] Elvio Amparore, Alan Perotti, and Paolo Bajardi. To trust or not to trust an explanation: using leaf to evaluate local linear xai methods. *PeerJ Computer Science*, 7:e479, 2021.

[3] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

[4] Felix Biessmann and Dionysius Irza Refiano. A psychophysics approach for quantitative comparison of interpretable computer vision models. *arXiv preprint arXiv:1912.05011*, 2019.

[5] Judy Borowski, Roland Simon Zimmermann, Judith Schepers, Robert Geirhos, Thomas S. A. Wallis, Matthias Bethge, and Wieland Brendel. Exemplary natural images explain {cnn} activations better than state-of-the-art feature visualization. In *International Conference on Learning Representations*, 2021.

[6] Santiago A Cadena, Marissa A Weis, Leon A Gatys, Matthias Bethge, and Alexander S Ecker. Diverse feature visualizations reveal invariances in early layers of deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 217–232, 2018.

[7] Carrie J Cai, Jonas Jongejan, and Jess Holbrook. The effects of example-based explanations in a machine learning interface. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pages 258–262, 2019.

[8] Nick Cammarata, Gabriel Goh, Shan Carter, Ludwig Schubert, Michael Petrov, and Chris Olah. Curve detectors. *Distill*, 2020. doi: 10.23915/distill.00024.003. https://distill.pub/2020/circuits/curve-detectors.

[9] Nick Cammarata, Gabriel Goh, Shan Carter, Chelsea Voss, Ludwig Schubert, and Chris Olah. Curve circuits. *Distill*, 6(1):e00024–006, 2021.

[10] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[11] Joshua R de Leeuw and Benjamin A Motz. Psychophysics in a web browser? comparing response times collected with javascript and psychophysics toolbox in a visual search task. *Behavior Research Methods*, 48(1):1–12, 2016.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. doi: 10.1109/CVPR.2009.5206848.

[13] William K Diprose, Nicholas Buist, Ning Hua, Quentin Thurier, George Shand, and Reece Robinson. Physician understanding, explainability, and trust in a hypothetical machine learning risk calculator. *Journal of the American Medical Informatics Association*, 27(4):592–600, 2020.

[14] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

[15] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. *arXiv preprint arXiv:1906.00945*, 2019.

[16] Dumitru Erhan, Yoshua Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009.

[17] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pages 2950–2958. IEEE, 2019. doi: 10.1109/ICCV.2019.00304.

[18] Christina M Funke, Judy Borowski, Karolina Stosio, Wieland Brendel, Thomas SA Wallis, and Matthias Bethge. Five points to check when comparing visual perception in humans and machines. *Journal of Vision*, 21(3):16–16, 2021.

[19] Robert Geirhos, Kristof Meding, and Felix A. Wichmann. Beyond accuracy: quantifying trial-by-trial behaviour of cnns and humans by measuring error consistency. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[20] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 6(3): e30, 2021.

[21] Nicolas Gonthier, Yann Gousseau, and Saïd Ladjal. An analysis of the transfer learning of convolutional neural networks for artistic images. *arXiv preprint arXiv:2011.02727*, 2020.

[22] Yash Goyal, Ziyan Wu, Jan Ernst, Dhruv Batra, Devi Parikh, and Stefan Lee. Counterfactual visual explanations. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2376–2384. PMLR, 2019.

[23] Peter Green and Catriona J MacLeod. Simr: an r package for power analysis of generalized linear mixed models by simulation. *Methods in Ecology and Evolution*, 7(4):493–498, 2016.

[24] Peter Hase and Mohit Bansal. Evaluating explainable AI: Which algorithmic explanations help users predict model behavior? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5540–5552, Online, 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.491.

[25] Amir-Hossein Karimi, Gilles Barthe, Borja Balle, and Isabel Valera. Model-agnostic counterfactual explanations for consequential decisions, 2020.

[26] Nikolaus Kriegeskorte. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual review of vision science*, 1:417–446, 2015.

[27] Matthew L Leavitt and Ari Morcos. Towards falsifiable interpretability research. *arXiv preprint arXiv:2010.12016*, 2020.

[28] Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, page 2119–2128, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605582467. doi: 10.1145/1518701.1519023.

[29] Akis Linardos, Matthias Kümmerer, Ori Press, and Matthias Bethge. Calibrated prediction in and out-of-domain for state-of-the-art saliency modeling, 2021.

[30] Ana Lucic, Hinda Haned, and Maarten de Rijke. Why does my model fail? contrastive local explanations for retail forecasting. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 90–98, 2020.

[31] Wei Ji Ma and Benjamin Peters. A neural network walks into a lab: towards using deep nets as models for human behavior. *arXiv preprint arXiv:2005.02181*, 2020.

[32] Divyat Mahajan, Chenhao Tan, and Amit Sharma. Preserving causal constraints in counterfactual explanations for machine learning classifiers, 2020.

[33] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 5188–5196. IEEE Computer Society, 2015. doi: 10.1109/CVPR.2015.7299155.

[34] Sina Mohseni, Jeremy E Block, and Eric Ragan. Quantitative evaluation of machine learning explanations: A human-grounded benchmark. In *26th International Conference on Intelligent User Interfaces*, pages 22–31, 2021.

[35] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, 2015.

[36] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 3510–3520. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.374.

[37] Anh Nguyen, Jason Yosinski, and Jeff Clune. Understanding neural networks via feature visualization: A survey. In *Explainable AI: interpreting, explaining and visualizing deep learning*, pages 55–76. Springer, 2019.

[38] Anh Mai Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 427–436. IEEE Computer Society, 2015. doi: 10.1109/CVPR.2015.7298640.

[39] Anh Mai Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 3387–3395, 2016.

[40] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2(11): e7, 2017.

[41] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. An overview of early vision in inceptionv1. *Distill*, 2020. doi: 10.23915/distill.00024.002. https://distill.pub/2020/circuits/early-vision.

[42] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. https://distill.pub/2020/circuits/zoom-in.

[43] Chris Olah, Nick Cammarata, Chelsea Voss, Ludwig Schubert, and Gabriel Goh. Naturally occurring equivariance in neural networks. *Distill*, 5(12):e00024–004, 2020.

[44] OpenAI. OpenAI Microscope. `https://microscope.openai.com/models`, 2020. (Accessed on 09/12/2020).

[45] Judea Pearl. *Causality*. Cambridge university press, 2009.

[46] Michael Petrov, Chelsea Voss, Ludwig Schubert, Nick Cammarata, Gabriel Goh, and Chris Olah. Weight banding. *Distill*, 6(4):e00024–009, 2021.

[47] Forough Poursabzi-Sangdeh, Daniel G Goldstein, Jake M Hofman, Jennifer Wortman Vaughan, and Hanna Wallach. Manipulating and measuring model interpretability. *arXiv preprint arXiv:1802.07810*, 2018.

[48] Grusha Prasad, Yixin Nie, Mohit Bansal, Robin Jia, Douwe Kiela, and Adina Williams. To what extent do human explanations of model behavior align with actual model behavior? *arXiv preprint arXiv:2012.13354*, 2020.

[49] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y.

[50] Ludwig Schubert, Chelsea Voss, Nick Cammarata, Gabriel Goh, and Chris Olah. High-low frequency detectors. *Distill*, 6(1):e00024–005, 2021.

[51] Shubham Sharma, Jette Henderson, and Joydeep Ghosh. Certifai: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models. *arXiv preprint arXiv:1905.07857*, 2019.

[52] Stefan Sietzen, Mathias Lechner, Judy Borowski, Ramin Hasani, and Manuela Waldner. Interactive analysis of cnn robustness. *Computer Graphics Forum (Proceedings of Pacific Graphics 2021)*, 40(7), 2021.

[53] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA,*

USA, June 7-12, 2015, pages 1–9. IEEE Computer Society, 2015. doi: 10.1109/CVPR.2015.7298594.

[54] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[55] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and Aleksander Madry. From imagenet to image classification: Contextualizing progress on benchmarks. In *International Conference on Machine Learning*, pages 9625–9635. PMLR, 2020.

[56] Berk Ustun, Alexander Spangher, and Yang Liu. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 10–19, 2019.

[57] Arnaud Van Looveren and Janis Klaise. Interpretable counterfactual explanations guided by prototypes. *arXiv preprint arXiv:1907.02584*, 2019.

[58] Chelsea Voss, Nick Cammarata, Gabriel Goh, Michael Petrov, Ludwig Schubert, Ben Egan, Swee Kiat Lim, and Chris Olah. Visualizing weights. *Distill*, 6(2):e00024–007, 2021.

[59] Chelsea Voss, Gabriel Goh, Nick Cammarata, Michael Petrov, Ludwig Schubert, and Chris Olah. Branch specialization. *Distill*, 6(4):e00024–008, 2021.

[60] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr, 2018.

[61] Eric Wong, Shibani Santurkar, and Aleksander Mądry. Leveraging sparse linear layers for debuggable deep networks, 2021.

[62] Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel S Weld. Polyjuice: Automated, general-purpose counterfactual generation. *arXiv preprint arXiv:2101.00288*, 2021.

[63] Xi Ye, Rohan Nair, and Greg Durrett. Evaluating explanations for reading comprehension with realistic counterfactuals. *arXiv preprint arXiv:2104.04515*, 2021.

[64] Ming Yin, Jennifer Wortman Vaughan, and Hanna M. Wallach. Understanding the effect of accuracy on trust in machine learning models. In Stephen A. Brewster, Geraldine Fitzpatrick, Anna L. Cox, and Vassilis Kostakos, editors, *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI 2019, Glasgow, Scotland, UK, May 04-09, 2019*, page 279. ACM, 2019. doi: 10.1145/3290605.3300509.

[65] Mohammad Nokhbeh Zaeem and Majid Komeili. Cause and effect: Concept-based explanation of neural networks. *arXiv preprint arXiv:2105.07033*, 2021.

[66] Yilun Zhou, Serena Booth, Marco Tulio Ribeiro, and Julie Shah. Do feature attribution methods correctly attribute features? *arXiv preprint arXiv:2104.14403*, 2021.

A Appendix

A.1 Details on Methods of Counterfactual-Inspired Experiment

We closely follow our previous work [5] and hence often refer to specific sections of it in this Appendix.

A.1.1 Data Collection

Exclusion Criteria In order to acquire data of high quality from MTurk, we integrate five exclusion criteria. If one or more of these criteria is not met, we post the same HIT again:

- Maximal number of attempts to reach 100% performance in practice trials: 5
- Performance threshold for catch trials: two out of three trials have to be correctly answered
- Answer variability: at least one trial must be chosen from the less frequently selected side (to discard participants who only responded with "left" or "right")
- Time to read the instructions: at least $20\,\mathrm{s}$ ($15\,\mathrm{s}$ in the none condition)
- Time for the whole experiment: at least $90\,\mathrm{s}$ and at most $900\,\mathrm{s}$ (at least $40\,\mathrm{s}$, and at most $900\,\mathrm{s}$ in the none condition)

Minimize Biases To minimize a bias to either query image, the location of the truly maximally activating query image is randomized and participants have to center their mouse cursor by pressing a centered button "Continue" after each trial.

Expert Measurements The two first authors complete all 10 image sets in multiple conditions: At first, they label the query images for the Primary Object Baseline. Then they answer the none, synthetic or natural (counterbalanced between the two authors), mixed, and blur condition. Clicking through the trials several times means that they see identical images repeatedly.

A.1.2 Stimulus Generation

Model In line with previous work (e.g. Borowski et al. [5], Olah et al. [40]), we use an Inception V1 network [53] trained on ImageNet [12, 49]. For more details, see Sec. A.1.2 "Stimuli Selection - Model" in Borowski et al. [5].

Natural Images as Query and Reference Images The natural reference and query images are selected from a random subset of $599,552$ training images of the ImageNet ILSVRC 2012 dataset [49]. For each unit, we select those images that elicit a maximal activation. More specifically, we choose the very most activating images as the query images and the next most activating images as reference images and ensure no overlap between query and references images as well as between image sets. As we follow our work published in Borowski et al. [5], please see A.1.2 for more details on the sampling procedure. In total, we generate 20 different image sets per unit. In the presented data, we only use half of these sets.

Query Images For the query images, we use the 20 maximally activating images for a given unit. To produce the manipulated query images, a square patch of 90×90 pixels is placed on the unperturbed query image. The side length of a patch corresponds to 40% of a preprocessed image's side length. The position of the occlusion patch is chosen such that the manipulated image's activation for a given unit is minimal (maximal) among all possible manipulated images' activations. This maximizes the signal in the query images and means that patches of the two query images can overlap.

In a control experiment, we test whether the partial occlusions of the natural ImageNet images cause the manipulated images to lie outside the natural image distribution. If this was the case, the query images would fail to be representative of the network's activity for natural images. Here, we test how similar the response to the unperturbed and partially occluded images is. Specifically, we count how often there is an overlap of the top-5 predictions. If network activations were drastically different for the occluded than for the unperturbed images, we should find low agreement. However, we do find an agreement for $97.8,\%$ of all tested images. Therefore, the square occlusions only have a marginal effect on the network's overall activity/predictions.

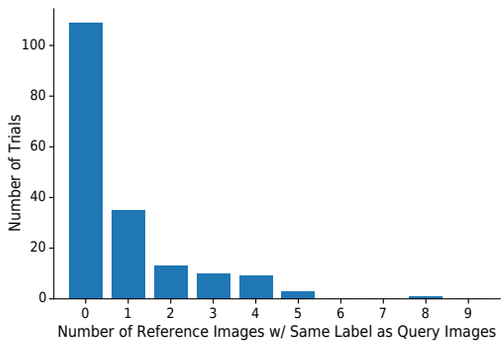


Figure 8: Distribution of the number of natural reference images that have the same label as the query image over the main trials used in the counterfactual-inspired experiment.

Reference Images: Natural Images In a control experiment, we test how often the label of the reference images coincide with the query image's label. If there was a high correspondence of these ImageNet labels, this could suggest that our experiment would rather reveal insights on how well humans would be able to *classify* images according to *labels* rather than to answer a counterfactual-inspired task based on the unit activations. Fig. 8 shows that the overlap of labels between query and reference images is low.

Reference Images: Blurred Images The blurred reference images are created by blurring all but one patch with a Gaussian kernel of size $(21, 21)$. This parameter choice allows participants to still get a general impression of an image, but not recognize details. Further, it is in line with work by Fong et al. [17]. The image choices are identical to the natural condition. Further — and just like for the query images — the position of the unblurred patch is chosen such that the manipulated image's activation for a given unit is maximal among all possible manipulated images' activations. Finally, the size of the unblurred patch is identical to the occlusion patch size: 40% of a preprocessed image's side length.

Reference Images: Synthetic Images from Feature Visualization Depending on the condition, we adjust the number of feature visualizations we generate: For the purely synthetic condition, we generate 9 visualizations, for the mixed condition, we generate 4 visualizations. As we follow our work published in Borowski et al. [5], please see A.1.2 for further details.

A.1.3 Baselines

Primary Object Baseline The Primary Object Baseline simulates that the more strongly activating manipulated image would be the one where the occlusion hides as little as possible from the most prominent object of the query image. To this end, the first two authors and the last author label all images. When doing so, they use a slightly modified logic: They select the image whose most prominent object is *most* occluded. If they cannot clearly identify a primary object in the image, they flag these trials, which are then treated differently in the analysis. For the analysis, the image choice is inverted again to counteract the inverted task that the authors responded to.

The performance reported in Fig. 4 is calculated by averaging over the three individual performances. Each individual performance itself is in turn estimated as the expectation value over random sampling for query images with no clear primary object. This analysis is in line with how the performance of MTurk participants is analyzed. An alternative option would be to take the majority vote of the three answers. When randomly sampling the choice for query images with no clear primary object, taking the majority votes and evaluating the expected accuracy, the performance would evaluate to 0.70 ± 0.02. Notably, 58 of all 180 trials are affected by the sampling as two or more authors responded with a confidence of 1 in 36 trials, and one author responded with a confidence of 1 while the other two gave opposing answers in 22 trials. This represents a fairly large fraction and reflects that many images on ImageNet have more than one prominent object [55, 3]. Consequently, there may not be a ground-truth for each trial in the Primary Object Baseline.

Saliency Baseline The Saliency Baseline simulates that participants select the image with a patch occluding the less prominent image region. To this end, we pass the unoccluded query image through the saliency prediction model DeepGaze IIE [29] which yields a probability density over the entire image. Next, we integrate said density over each of the two square patches. We then select the image with a lower value indicating that less important information is hidden by the occlusion patch.

A.1.4 Trials

Main trials For both the 3×3 and the POOL branch of each of the 9 layers with an Inception module, one randomly chosen unit is tested (see Table 1). These are the same units as in Experiment I of Borowski et al. [5].

Table 1: Units used as main trials in the 3×3 as well as the POOL branch in the counterfactual-inspired experiment. The numbers in brackets after each layer's name correspond to the numbering used in all our plots.

	Unit	
Layer	3×3	POOL
mixed3a (1)	189	227
mixed3b (2)	178	430
mixed4a (3)	257	486
mixed4b (4)	339	491
mixed4c (5)	247	496
mixed4d (6)	342	483
mixed4e (7)	524	816
mixed5a (8)	278	743
mixed5b (9)	684	1007

Instruction, Practice and Catch Trials The instruction, practice and catch trials are hand-picked by the two first authors. As a pool of units, the appendix overview of Olah et al. [40] as well as the "interpretable" POOL units used in Experiment I and all units used in Experiment II of Borowski et al. [5] are used. After generating all 20 reference and query image sets for these units, the authors select those units and image sets that they consider easiest (see Table 2).

Instruction Trial To explain the task as intuitively as possible, we construct an easy, artificial instruction trial (see Fig. 9 and 10): At first, we select a unit with easily understandable feature visualizations: The synthetic images of unit 720 of the POOL branch in layer 8 show relatively clear bird-like structures. From a popular image search engine, we then select an image[8] which not only clearly shows a bird but also other objects, namely a dog and water. To construct the minimally and maximally activating query images, we place the occlusion patches manually on the bird and dog.

Practice Trials In each attempt to pass the practice block, the trials are randomly sampled from a pool of 10 trials (see Table 2). Please note that unlike in any other trial type, participants receive feedback in the practice block: After each trial, they learn whether their chosen image truly is the query image of higher activation.

Catch Trials While all conditions with reference images use hand-picked easy trials (see Table 2), the none condition cannot rely on straight-forward clues from references. Therefore, we exchange the minimal query image with a minimal query image of a different, otherwise unused unit. This ensures that the catch trials in the none condition are also obvious.

A.1.5 Infrastructure

The online experiment is hosted on an Ubuntu 18.04 server running on an Intel(R) Xeon(R) Gold 5220 CPU. The experiment is implemented in JavaScript using jspsych 6.3.1 [11] and flask via

[8] `https://pixnio.com/fauna-animals/dogs/dog-water-bird-swan-lake-waterfowl-animal-swimming` released into public domain under CC0 license by Bicanski.

Table 2: Hand-picked unit choices for instruction, catch and practice trials in the counterfactual-inspired experiment.

Trial Type	Layer	Branch	Unit	Difficulty Level
instruction	mixed5a	pool	720	very easy
catch	mixed4e	pool	783	very easy
	mixed4c	pool	484	very easy
	mixed5a	3×3	557	very easy
practice	mixed4e	1×1	6	very easy
	mixed4a	pool	505	very easy
	mixed4e	pool	809	very easy
	mixed4c	pool	449	easy
	mixed4b	pool	465	easy
	mixed4c	1×1	59	easy
	mixed4e	1×1	83	easy
	mixed3a	1×1	43	easy
	mixed3b	pool	472	easy
	mixed4b	1×1	5	easy

Python 3.6. The generation of the stimuli shown in the experiment was completed in approximately 35 hours on a single GeForce GTX 1080 GPU. The calculation of all baselines required 8 additional GPU hours.

A.1.6 Amazon Mechanical Turk

MTurk participants To increase the chance that all MTurk participants understand the English instructions at the beginning of the experiment, we restrict access to workers from the following English-speaking countries: USA, Canada, Great Britain, Australia, New Zealand and Ireland.

Financial Compensation Based on an estimated duration and pilot experiments as well as a targeted hourly rate of US$ 15, we calculate the pay to be US$ 0.70 for the none condition and US$ 1.95 for all other conditions. MTurk participants whose data we include need a mean time of 209.64 ± 79.53 s and 396.87 ± 145.78 s for the whole experiment for the none condition and for all other conditions, respectively, which results in an hourly compensation of ≈ 12.02 US$/hour and 17.69 US$/hour, respectively. All MTurk participants who fully complete a HIT are paid, regardless of whether their responses meet the exclusion criteria. A total of US$ 1989.06 is spent on all pilot and real replication and counterfactual-inspired experiments.

Rights to Data We do not gather personal identifiable data from any MTurk participant. According to the MTurk Participation Agreement 3a [9], workers agree to vest all ownership and intellectual property rights to the requester (i.e., the authors of this study). Besides informing MTurk participants in the HIT preview about the academic and image classification nature of the experiment, we restate that "By completing this HIT, you consent to your anonymized data being shared with us for a scientific study." Further, we provide an email address, which some MTurk participants used to share feedback.

[9] `https://www.mturk.com/participation-agreement`, accessed on May 22nd, 2021

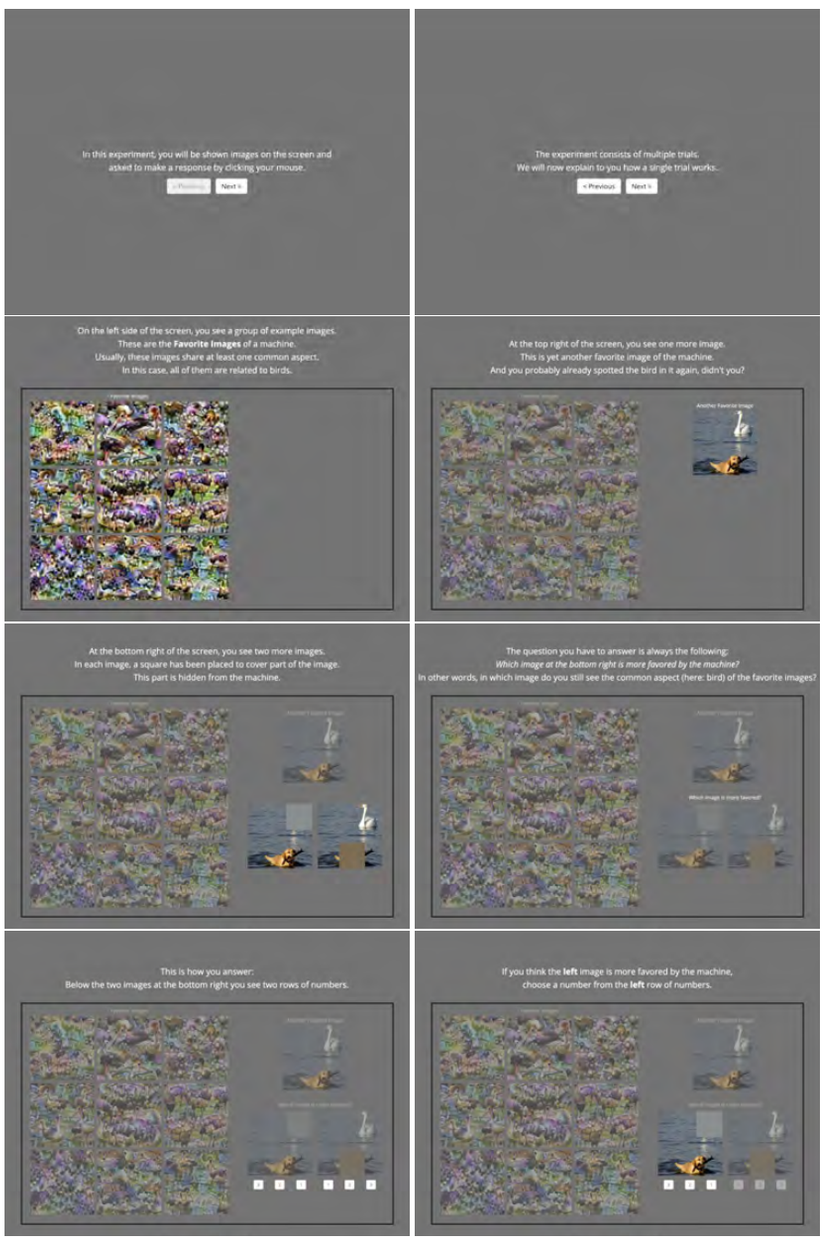


Figure 9: First eight instructions at the beginning of the counterfactual-inspired experiment.

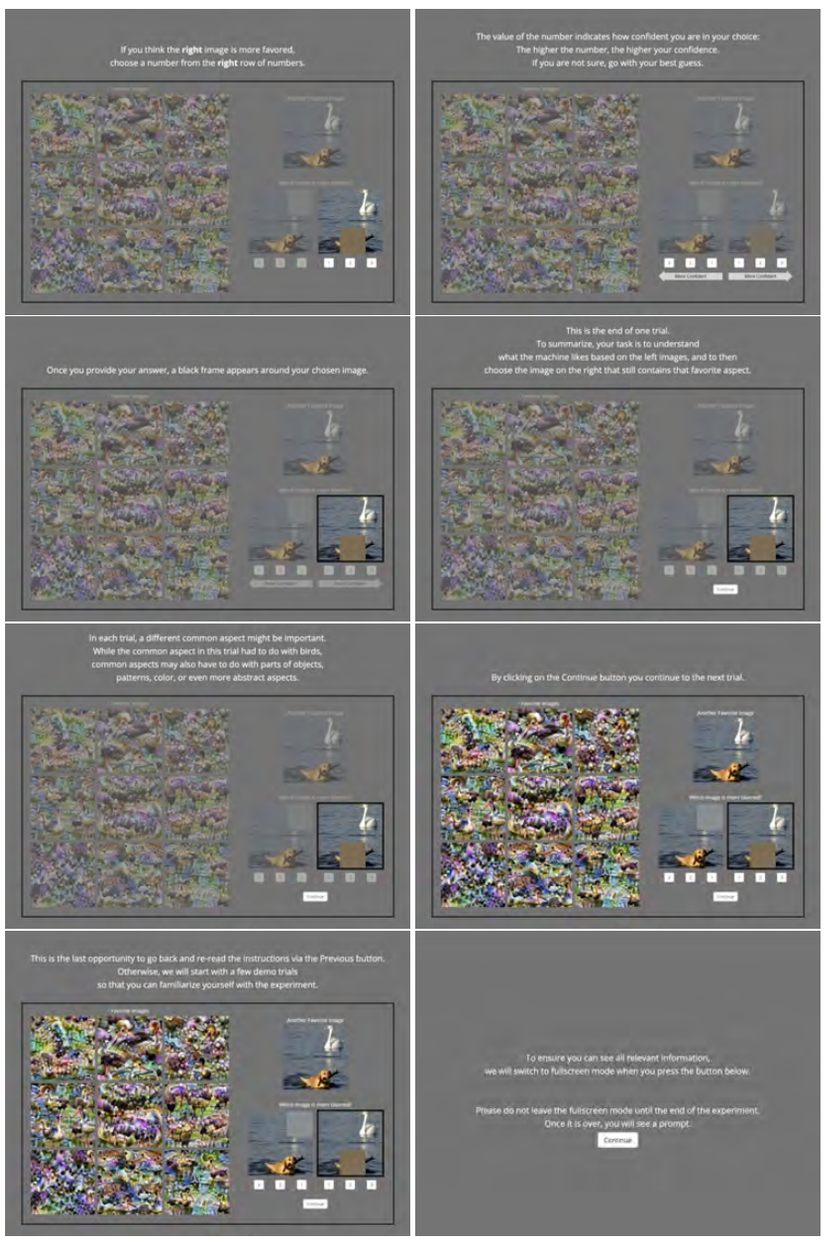


Figure 10: Second eight instructions at the beginning of the counterfactual-inspired experiment.

A.2 Details on Results of Counterfactual-Inspired Experiment

A.2.1 Different Query images

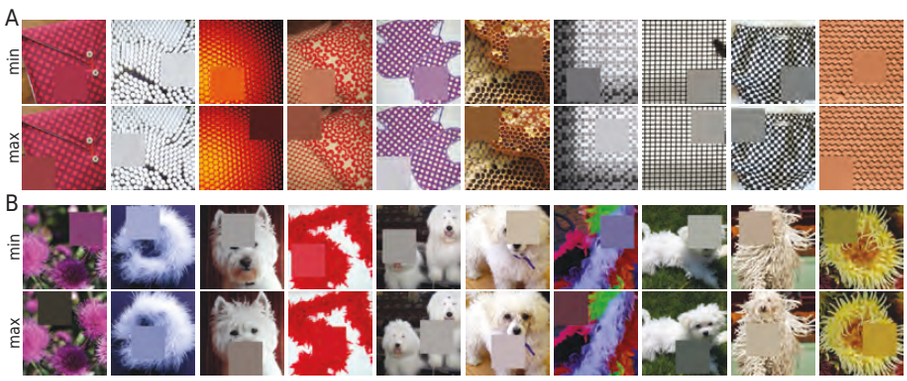
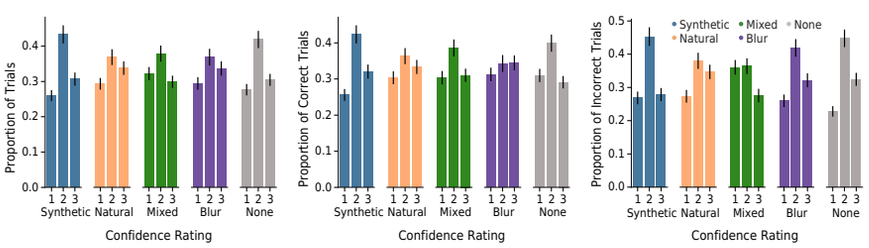


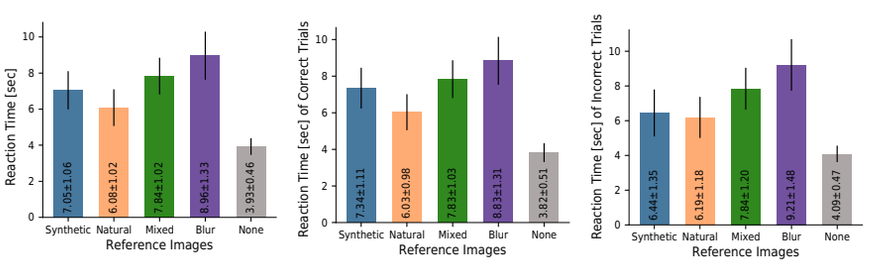
Figure 11: For each unit, we test 10 different image sets in the counterfactual-inspired experiment. The diversity of query images for layer 3 of the 3×3 branch (**A**), and layer 7 of the POOL branch (**B**) gives an intuitive explanation for varying performances.

A.2.2 Confidence Ratings and Reaction Times



(a) (b) (c)

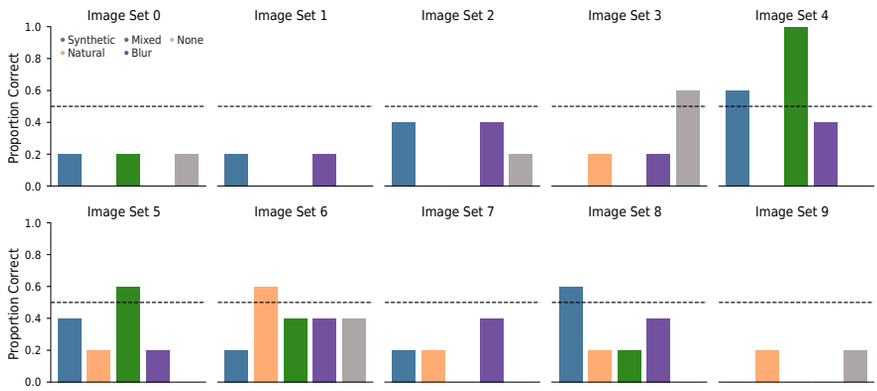
Figure 12: Confidence ratings of MTurk participants in the different reference conditions for (a) all, (b) only correct or (c) only incorrect trials of the counterfactual-inspired experiment.



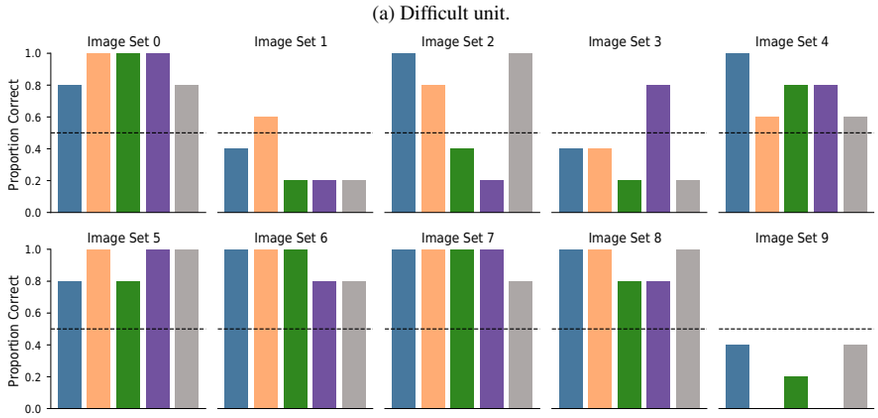
(a) (b) (c)

Figure 13: Reaction times of MTurk participants in the different reference conditions for (a) all, (b) only correct or (c) only incorrect trials of the counterfactual-inspired experiment.

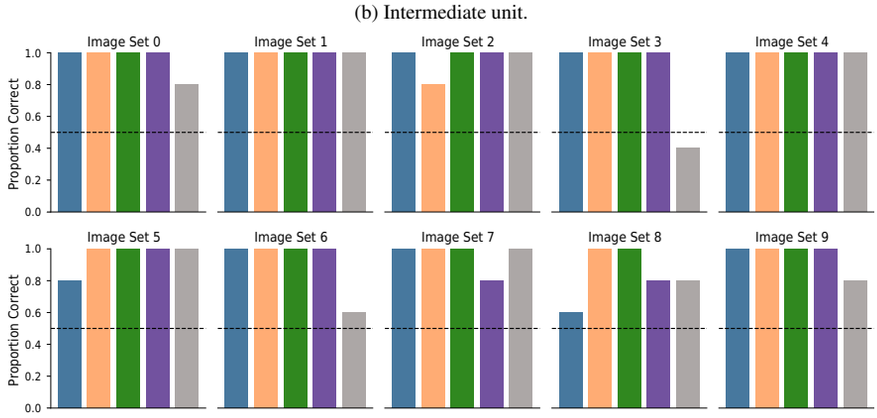
A.2.3 Performance per Image Set



(a) Difficult unit.



(b) Intermediate unit.



(c) Easy unit.

Figure 14: Performance in the counterfactual-inspired experiment split up by image sets and conditions for a difficult (layer 3, POOL), intermediate (layer 7, POOL) and easy unit (layer 8, POOL). Each bar shows the average over 5 MTurk participants.

A.2.4 Strategy Comparisons

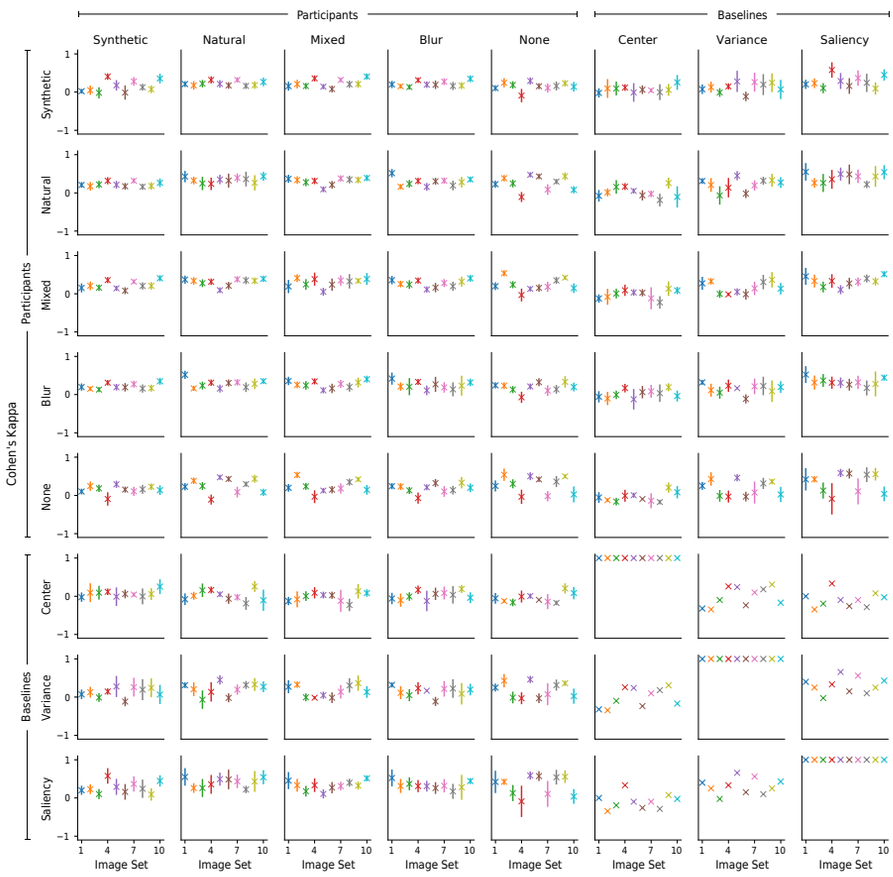
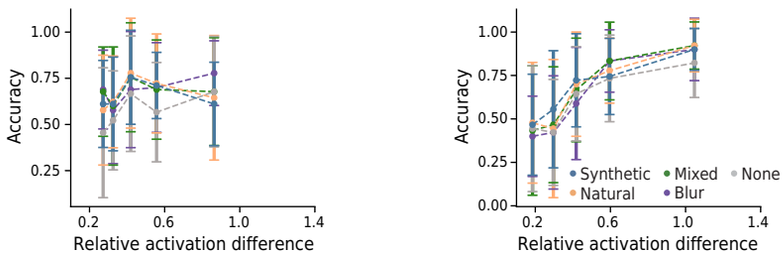


Figure 15: Cohen's kappa per image set in the counterfactual-inspired experiment (averages over participant-participant-, participant-baseline- or baseline-baseline-pairs). Error bars denote two standard errors of the mean.

A.2.5 Relative Activation Differences

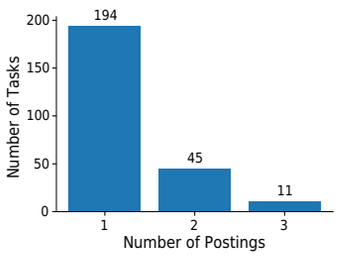


(a) 3×3 branch.

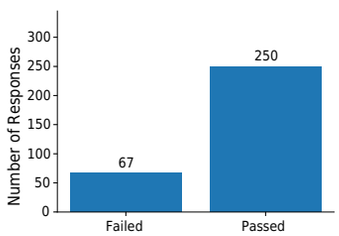
(b) POOL branch.

Figure 16: Accuracy in the counterfactual-inspired experiment as a function of the relative activation difference between the two query images for the (a) 3×3 branch and the (b) POOL branch. Here, the data points shown in Fig. 7 are summarized in 5 bins of equal counts; the plot shows the mean and standard deviation for each of the bins.

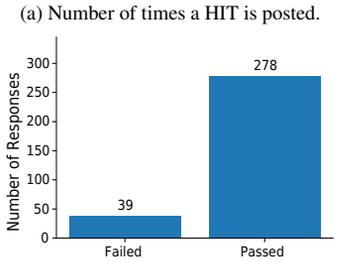
A.2.6 Exclusion Criteria



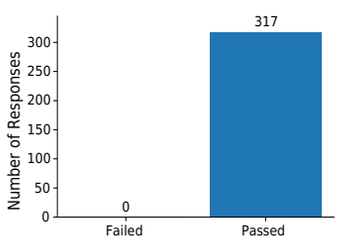
(a) Number of times a HIT is posted.



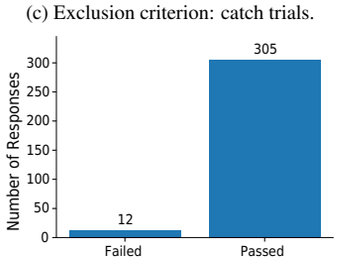
(b) All exclusion criteria.



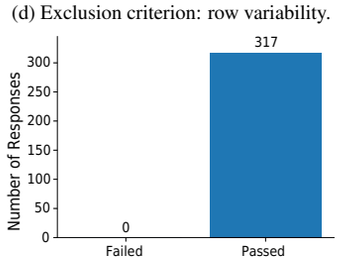
(c) Exclusion criterion: catch trials.



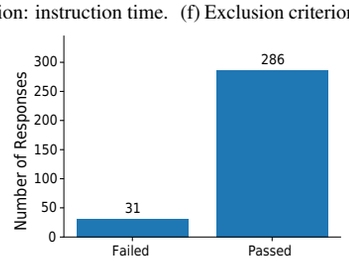
(d) Exclusion criterion: row variability.



(e) Exclusion criterion: instruction time.

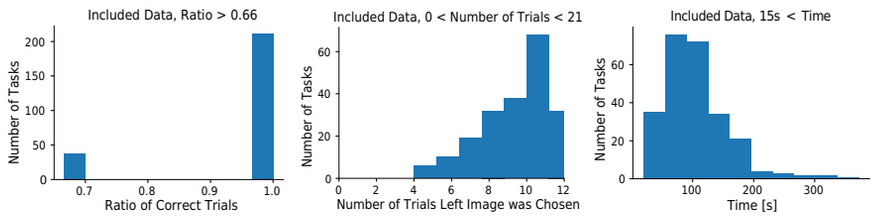


(f) Exclusion criterion: total response time.

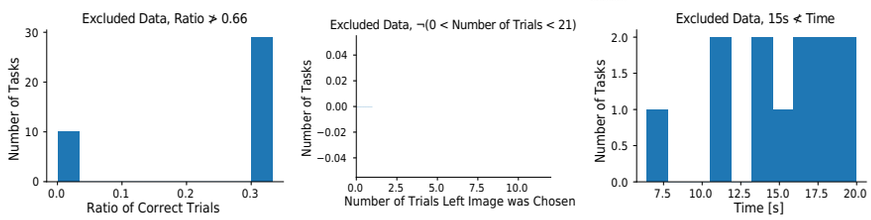


(g) Exclusion criterion: practice block.

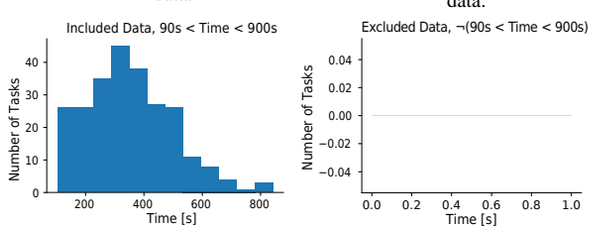
Figure 17: (a) Number of times a HIT is posted. To limit the financial risk, we limit the maximal number of times that a HIT can be posted at 3. (b-g) Distributions of MTurk participants that passed/failed the exclusion criteria in the counterfactual-inspired experiment on MTurk. Note that the sum of the counts of responses for the individual exclusion criteria in c-f is higher than the summary in b because a participant may have failed more than one exclusion criterion.



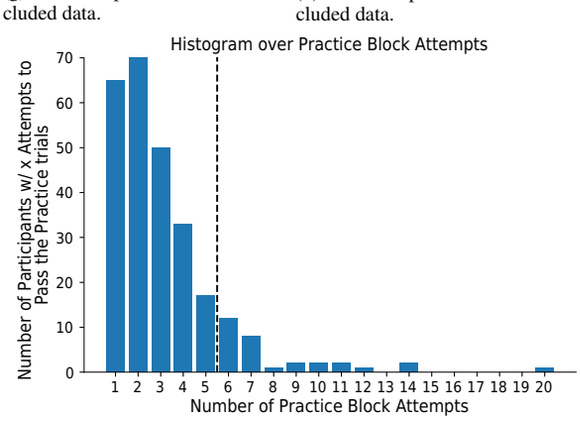
(a) Catch trials from included data. (b) Row variability from included data. (c) Instruction time from included data.



(d) Catch trials from excluded data. (e) Row variability from excluded data. (f) Instruction time from excluded data.



(g) Total response time from included data. (h) Total response time from excluded data.



(i) Practice Block Attempts: We include data from people who needed 5 or fewer attempts.

Figure 18: Distributions of the individual values controlled by the exclusion criteria in the counterfactual-inspired experiment on MTurk. For the first four criteria, a - c and g (d - f and h) show the data for the included (excluded) data. The final criterion in i shows a joint distribution.

A.3 Replication of the Main Result of Borowski et al. [5]

To check whether collecting data on a crowdsourcing platform yields sensible data in our case, we first test whether we can replicate the main finding of our previous human psychophysical experiment on feature visualizations [5]. In the latter, we found in a well-controlled lab environment that natural reference images are more informative than synthetic ones when choosing which of two different images are more highly activating for a given unit. Below, we report how we alter the experimental set-up to turn the lab experiment into an online experiment on MTurk and what results we find.

A.3.1 Experimental Set-up

While keeping as many aspects as possible consistent with our original study [5], we make a few changes: (1) We run an online crowdsourced experiment on MTurk, instead of in a lab. (2) Instead of testing the 45 units used in the original Experiment I, we only test one single branch of each Inception module, namely the 3×3 kernel size. This is a reasonable decision given that the main finding of the superiority of natural over synthetic images was present in all branches and that there was no significant difference per condition between different branches. (3) We exchange the within-participant design for a between-participant design, i.e. one MTurk participant does one condition only, namely either the natural or synthetic reference condition. This version is more suitable for short online experiments. (4) Instead of testing 10 participants in the lab, we test 130 MTurk participants per condition, i.e. 260 in total. This number of participants is estimated with an a priori power analysis using the SIMR package [23] to allow us to detect an effect half as large as the one observed in Borowski et al. [5] 80% of the time. Assumptions about variance, average performance, and effect size are chosen to be conservative relative to the original study because we expect MTurk participants' responses to be noisier.

One HIT on MTurk consists of 1 extensively explained instruction trial, 2 practice trials, and then 9 main trials that are randomly interleaved with a total of 3 catch trials. Each trial type is sampled from a disjoint pool of units: All participants see the same unit for the instruction trial; the catch trials are sampled from the same pool as in the original experiment, and the practice trials are the units that were used as interpretability judgment trials in [5], namely mixed3a, kernel size 1×1, unit 43; mixed4b, POOL, unit 504; mixed5b, 1×1, unit 17. A total of 13 participants see the same main trials that one lab participant saw. The order of the main and catch trials per participants is randomly arranged.

Exclusion Criteria If a participant's response does not meet one or more of the following criteria, which were determined before data collection, we discard it and post the same HIT again:

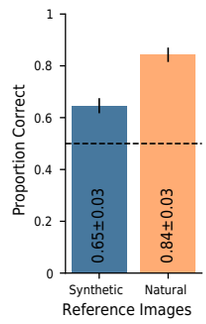
- Performance threshold for catch trials: two out of three trials have to be correctly answered
- Answer variability: at least one trial must be chosen from the less frequently selected side (to discard participants who only responded with "up" or "down")
- Time to read the instructions: at least 15 s
- Time for the whole experiment: at least 90 s and at most 600 s

MTurk compensation Based on an estimated and pilot experiment duration as well as an hourly rate of US$ 15, we calculate the pay to be US$ 1.25. We pay all MTurk participants who fully complete the experiment regardless of whether they succeed or fail in the exclusion criteria. The experiment without pilot experiments costs US$ 447. MTurk participants whose data we include need a mean time of 220.70 ± 71.58 s for the whole experiment, which results in an hourly compensation of ≈ 20.39 US$/hour.

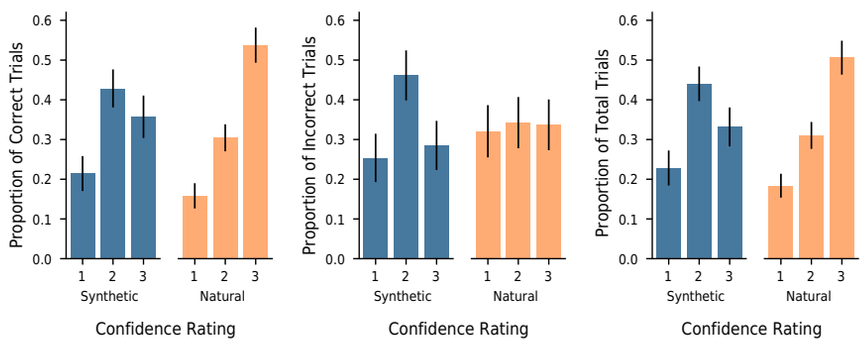
A.3.2 Results

MTurk participants achieve a higher performance when given natural than synthetic reference images: $84 \pm 3\,\%$ vs. $65 \pm 3\,\%$ (see Fig. 19a). Qualitatively, this result is the same as in the original Experiment I, see Figure 16 in Borowski et al. [5]. More precisely, the data shows a 1.35 (2.1) times larger odds (accuracy) difference for the replication. Compared to the lab data, MTurk participants seem more confident on the synthetic condition (see Fig. 19b-d), are faster in the synthetic condition (see Fig. 19e-g), and are about as fast in the natural condition (see Fig. 19e-g).

Fig. 20 shows that most participants passed the exclusion criteria. For more details on the number of postings per HIT and for more details on the MTurk participants' performance on the exclusion criteria, see 21.



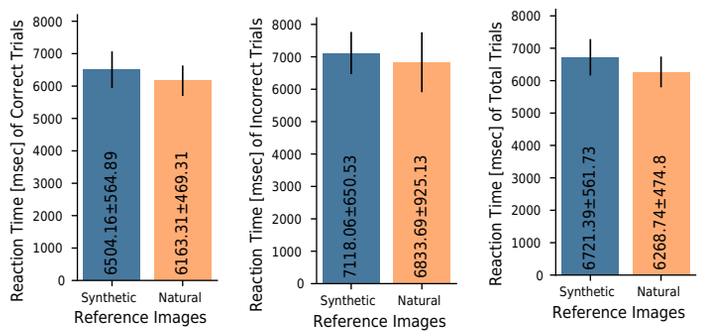
(a) Performance.



(b) Confidence ratings on correctly answered trials.

(c) Confidence ratings on incorrectly answered trials.

(d) Confidence ratings on all trials.

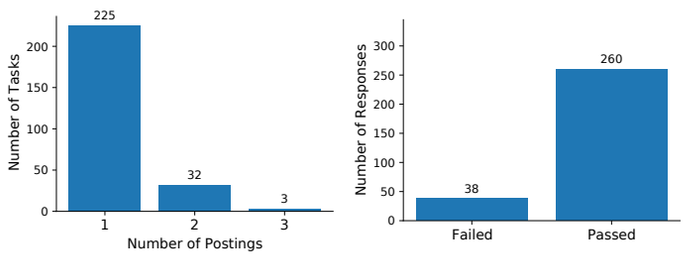


(e) Reaction time on correctly answered trials.

(f) Reaction time on incorrectly answered trials.

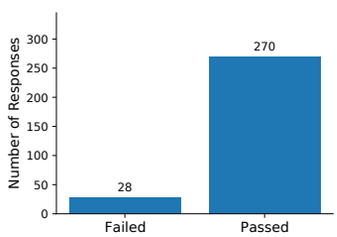
(g) Reaction time on all trials.

Figure 19: Results of the replication experiment of Borowski et al. [5] on MTurk for kernel size 3×3: task performance (a), distribution of confidence ratings (b-d) and reaction times (e-g).

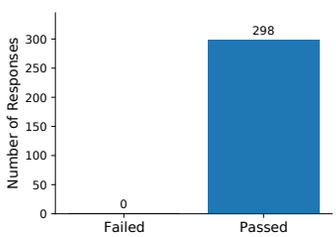


(a) Number of times a HIT is posted.

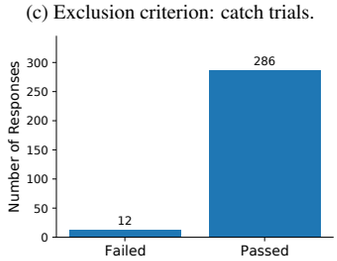
(b) All exclusion criteria.



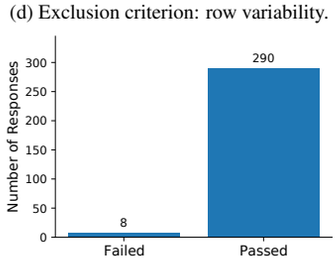
(c) Exclusion criterion: catch trials.



(d) Exclusion criterion: row variability.

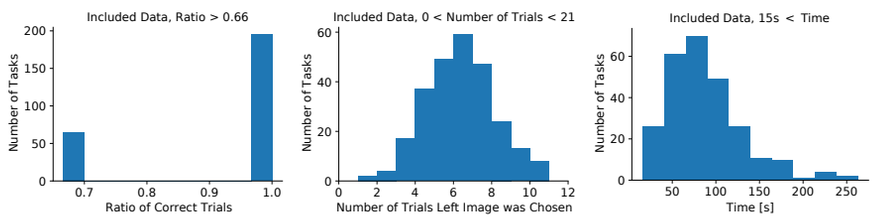


(e) Exclusion criterion: instruction time.

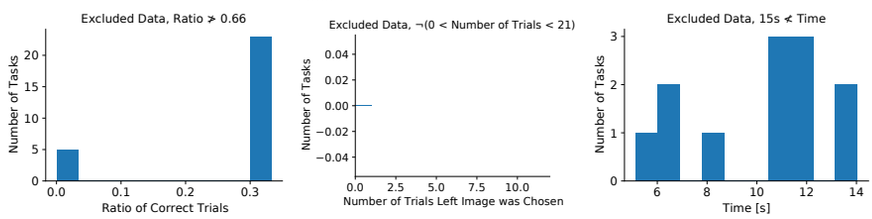


(f) Exclusion criterion: total response time.

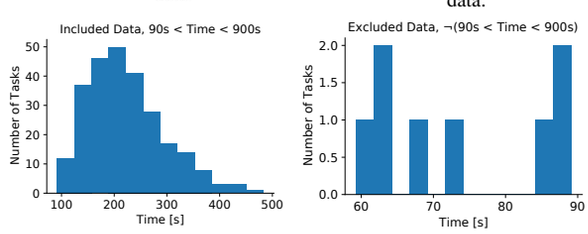
Figure 20: (a) Number of times a HIT is posted. (b-f) Distributions of MTurk participants that passed/failed the exclusion criteria in the replication experiment on MTurk. Note that the sum of the counts of responses for the individual exclusion criteria in c-f is higher than the summary in b because a participant may have failed more than one exclusion criterion.



(a) Catch trials from included data. (b) Row variability from included data. (c) Instruction time from included data.



(d) Catch trials from excluded data. (e) Row variability from excluded data. (f) Instruction time from excluded data.



(g) Total response time from included data. (h) Total response time from excluded data.

Figure 21: Distributions of the individual values controlled by the exclusion criteria in the replication experiment on MTurk. Figures a - c and g (d - f and h) show the data for the included (excluded) data.

A.3 Scale Alone Does not Improve Mechanistic Interpretability in Vision Model

The following 32 pages were published as:

Roland S. Zimmermann*, Thomas Klein*, and Wieland Brendel. "Scale Alone Does not Improve Mechanistic Interpretability in Vision Models." *NeurIPS (2023)*

A summary is given in Section 2.3 on page 37. * Equal contribution.

Abstract

In light of the recent widespread adoption of AI systems, understanding the internal information processing of neural networks has become increasingly critical. Most recently, machine vision has seen remarkable progress by scaling neural networks to unprecedented levels in dataset and model size. We here ask whether this extraordinary increase in scale also positively impacts the field of mechanistic interpretability. In other words, has our understanding of the inner workings of scaled neural networks improved as well? We here use a psychophysical paradigm to quantify mechanistic interpretability for a diverse suite of models and find no scaling effect for interpretability - neither for model nor dataset size. Specifically, none of the nine investigated state-of-the-art models are easier to interpret than the GoogLeNet model from almost a decade ago. Latest-generation vision models appear even less interpretable than older architectures, hinting at a regression rather than improvement, with modern models sacrificing interpretability for accuracy. These results highlight the need for models explicitly designed to be mechanistically interpretable and the need for more helpful interpretability methods to increase our understanding of networks at an atomic level. We release a dataset containing more than 130'000 human responses from our psychophysical evaluation of 767 units across nine models. This dataset is meant to facilitate research on automated instead of human-based interpretability evaluations that can ultimately be leveraged to directly optimize the mechanistic interpretability of models.

Scale Alone Does not Improve Mechanistic Interpretability in Vision Models

Roland S. Zimmermann[1*] **Thomas Klein**[1,2*] **Wieland Brendel**[1]

Abstract

In light of the recent widespread adoption of AI systems, understanding the internal information processing of neural networks has become increasingly critical. Most recently, machine vision has seen remarkable progress by scaling neural networks to unprecedented levels in dataset and model size. We here ask whether this extraordinary increase in scale also positively impacts the field of mechanistic interpretability. In other words, has our understanding of the inner workings of scaled neural networks improved as well? We use a psychophysical paradigm to quantify one form of mechanistic interpretability for a diverse suite of nine models and find no scaling effect for interpretability — neither for model nor dataset size. Specifically, none of the investigated state-of-the-art models are easier to interpret than the GoogLeNet model from almost a decade ago. Latest-generation vision models appear even less interpretable than older architectures, hinting at a regression rather than improvement, with modern models sacrificing interpretability for accuracy. These results highlight the need for models explicitly designed to be mechanistically interpretable and the need for more helpful interpretability methods to increase our understanding of networks at an atomic level. We release a dataset containing more than $130'000$ human responses from our psychophysical evaluation of 767 units across nine models. This dataset facilitates research on automated instead of human-based interpretability evaluations, which can ultimately be leveraged to directly optimize the mechanistic interpretability of models.

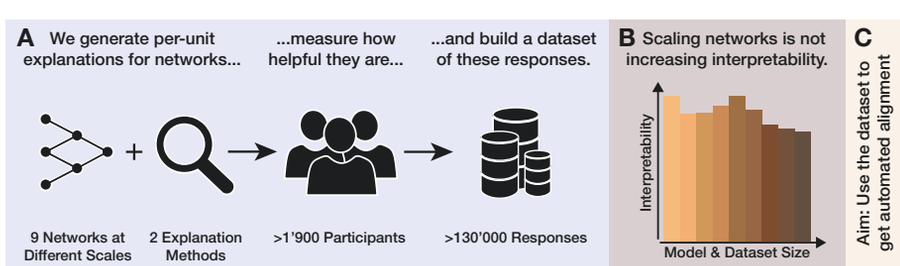


Figure 1: **Has scaling models in terms of their dataset and model size improved interpretability? A.** We perform a large-scale psychophysics experiment to investigate the interpretability of nine networks through the two most-used mechanistic interpretability methods. **B.** We see that scaling has not led to increased interpretability. Therefore, we argue that one has to explicitly optimize models to be interpretable. **C.** We expect our dataset to enable building automated measures for quantifying the interpretability of models and, thus, bootstrap the development of more interpretable models.

[*]Equal contribution. [1] Max Planck Institute for Intelligent Systems, Tübingen AI Center, Tübingen, Germany [2] University of Tübingen, Tübingen AI Center, Tübingen, Germany. Correspondence to: research@rzimmermann.com. Code & Dataset: brendel-group.github.io/imi.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

1 Introduction

Since the early days of deep learning, artificial neural networks have been referred to as black boxes: opaque systems that learn complex functions which cannot be understood, not even by the people who build and train them. Mechanistic interpretability [37] is an emerging branch of explainable AI (XAI) focused on understanding the internal information processing of deep neural networks, possibly by focusing on individual units as their atomic building blocks. This line of research is akin in spirit to the early days of neuroscience, where the receptive fields of cells in the mammalian visual cortex were investigated using single-cell electrophysiology [22]. Designing interpretable neural networks and aligning their information processing with that of humans would not only satisfy academic curiosity but also constitute a major step toward trustworthy AI that can be employed in high-stakes scenarios.

A natural starting point for mechanistic interpretability research is to investigate the individual units of a neural network. For convolutional neural networks (CNNs), the individual output channels of a layer, called activation maps, are often treated as separate units [38]. A common hypothesis is that channel activations correspond to the presence of features of the input [38]. There is hope that by understanding which feature(s) a unit is sensitive to, one could build a fine-grained understanding of a model by identifying complex circuits within the network [5]. To learn about a unit's sensitivity, researchers typically focus on inputs that cause strong activations at the target unit, either by obtaining highly activating images from the training set (*natural exemplars*), or by generating synthetic images that highly activate the unit. The well-known method of feature visualization [12, 38] achieves this through gradient ascent in input space (see Sec. 3.2). However, in practice, identifying a unit's sensitivity is far from trivial [4]. Historically, work on feature visualization has focused on the Inception architecture [47], in particular GoogLeNet. But in principle, both of these methods should work on arbitrary network architectures and models.

The starting hypothesis of this work is that the dramatic increase in both the scale of the datasets and the size of models [7, 45] might benefit per-unit mechanistic interpretability. Evidence for this hypothesis comes from recent work showing that models trained on larger datasets become more similar in their decisions to human judgments as measured by error consistency [14]. It is conceivable that models make more human-like decisions because they rely on non-spurious/human-aligned features. Therefore, one can argue that networks with more human-like decisions are more interpretable. Another argument for the hypothesis that scale is beneficial for unit-wise interpretability is that as models get larger, they can dedicate more units to represent learned features without having to encode features in superposition [10]. This could render the units more interpretable since the image features that activate them become less ambiguous.

We conduct a large-scale psychophysical study (see Fig. 1) to investigate the effects of scale and other design choices and find *no practically relevant* differences between any of the investigated models. While scaling models and datasets has fuelled the progress made on many research frontiers [7, 19, 25], it does not improve the mechanistic interpretability of individual units. Neither scale nor the other design choices make individual units more interpretable on their own.

As our study shows, new model design choices or training objectives are needed to *explicitly* improve the mechanistic interpretability of vision models. We expect the data collected in our study to serve as a starting point and test bed to develop cheap automated interpretability measures that do not require collecting human responses. These automated measures could pave the way for new ways to directly optimize model interpretability. Therefore, we release the study's results as a new dataset, called *ImageNet Mechanistic Interpretability* (IMI), to foster new developments in this line of research.

2 Related Work

The idea of investigating the information processing on the level of individual units in neural networks has a long history [e.g., 2, 53, 3, 32], possibly inspired by work in the neuroscience community that investigates receptive fields of individual neurons [e.g., 1, 39], dating back as far as the seminal work of Hubel and Wiesel [22] which categorized cells in the cat's visual cortex into simple and complex cells. The same holds for the technique of feature visualization, first proposed by Erhan et al. [12], developed further by, e.g., Mahendran and Vedaldi [31], Nguyen et al. [35], Mordvintsev et al. [33], Yosinski et al. [51], and popularized by Olah et al. [38]. Ghiasi et al. [16] present work

on extending feature visualizations to ViTs. Nguyen et al. [36] experimented with imposing priors on feature visualizations to make them more similar to natural images. Kalibhat et al. [24] aim to improve the interpretability afforded by natural exemplars by finding natural language descriptions of units through CLIP models [40]. Only years after the work on improving feature visualizations matured was their usefulness for understanding units experimentally quantified by Borowski et al. [4] and Zimmermann et al. [54], who found that feature visualizations are helpful but not more so than highly activating natural exemplars. Recently, Geirhos et al. [15] demonstrated that feature visualizations are not guaranteed to be reliable and might be misleading.

Much work on interpretability has focused on so-called post-hoc explanations, that is, explaining specific model decisions to end users [e.g. 41, 46, 26]. In contrast, mechanistic interpretability [37], the branch of XAI that we focus on here, is concerned with understanding the internal information processing of a model. This approach is not limited to the interpretability of single features we investigate here but also encompasses the analysis of entire circuits [5] and investigations of phase changes that occur over the course of training [34], to name just a few examples. See the review by Gilpin et al. [17] for a distinction and a broader overview of the field of XAI.

As Leavitt and Morcos [28] point out, it is vitally important to not only generate explanations that look convincing but also to conduct falsifiable hypothesis testing in interpretability research, which is what we attempt here. Furthermore, as Kim et al. [27] emphasize, interpretability should be evaluated in a human-centric way, a stance that motivates employing a psychophysical experiment with humans in the loop to measure interpretability. The field of interpretability has always struggled with a lack of consensus about definitions and suitable measurement scales [8, 29, 6]. Several previous works [e.g. 44, 20, 50, 27] focus on measuring the utility of post-hoc explanations. In contrast, we here are not primarily concerned with methods that explain model decisions to end-users, but instead focus on introspective methods that shed light on the internal information processing of neural networks.

Our psychophysical experiment builds on work by Borowski et al. [4] and Zimmermann et al. [54], whose psychophysical task we expand and adapt for arbitrary models as outlined in Sec. 3.2.

3 Methods

3.1 Measuring the Mechanistic Interpretability of Many Models

Selecting Models. We investigate nine computer vision models compatible with ImageNet classification [42]. These models span four different design axes, allowing us to analyze the influence of an increasing model scale on their interpretability. First, we look at the influence of model size in terms of parameter count, starting with GoogLeNet [47] at 6.8 million parameters and culminating in ConvNeXt-B [30] at 89 million parameters. Next, we look at various model design choices, such as increasing the width or depth of models (GoogLeNet vs. ResNet-50 [18] vs. WideResNet-50 [52] vs. DenseNet-201 [21]) and using different computational blocks (ViT-B [9] vs. ConvNeXt). Third, we scale training datasets up and compare the influence of training on 1 million ImageNet samples to pre-training on 400 million LAION [45] samples (ResNet-50 vs. Clip ResNet-50 [23, 40] and ViT-B vs. Clip ViT-B [23]). Last, we test the relation between adversarial robustness and interpretability (ResNet-50 vs. Robust ResNet-50 [43, 48]) as previous work [11, 49] found adversarial robustness to be beneficial for feature visualizations.

Selecting Units. For each of the investigated models, we randomly select 84 units (see Appx. A.5) by first drawing a network layer from a uniform distribution over the layers of interest and then selecting a unit, again at random, from the chosen layer. This scheme is used instead of randomly drawing units from a uniform distribution over all units since CNNs typically have more units in later layers. The layers of interest are convolution and normalization layers, as well as the outputs of skip connection blocks. We avoid the very first convolution layers since they can be interpreted more directly by inspecting their filters [38, 4]. For GoogLeNet, we select only from the last layers of each inception block in line with earlier work [4, 54]. For the ViT models, we adhere to the insights by Ghiasi et al. [16] and only inspect the position-wise feedforward layers.

Performing & Designing the Psychophysics Experiment. As interpretability is a human-centric model attribute, we perform a large-scale psychophysical experiment to measure the interpretability of models and individual units. For this, we use the experimental paradigm proposed by Borowski

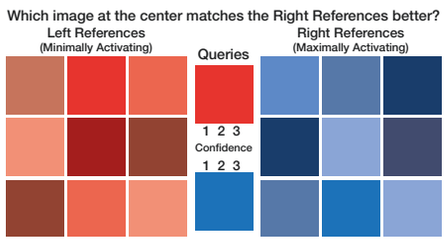


Figure 2: **Illustration of task design.** Users see a set of nine maximally/minimally activating reference images (synthetic feature visualizations or natural exemplars) on the right/left side of the screen. In the center, one strongly positively and a strongly negatively activating natural image are shown. Users need to pick the more positively activating query image (here, the bottom one) by pressing on a number indicating their confidence in their choice. See Fig. 8 for an example.

et al. [4] and Zimmermann et al. [54]: Here, the ability of humans to predict the sensitivity of units is used to measure interpretability. Specifically, crowd workers on Amazon Mechanical Turk complete a series of 2-Alternative-Forced-Choice (2-AFC) tasks (see Fig. 2 for an illustration). In each task, they are presented with a pair of strongly and weakly activating (query) images for a specific unit and are asked to identify the strongly activating one. During this task, they are supported by 18 explanatory (reference) images that strongly activate the unit, either natural dataset exemplars or synthetic feature visualizations. We begin by making the task as easy as possible by choosing the query images as the most/least activating samples from the ImageNet dataset. By choosing query images that cause less extreme activations, the task's difficulty can be increased, which allows us to probe a more general understanding of the unit's behavior by participants. For details refer to Appx. A.1.

While we explain the task to the participants, we do not instruct them to use specific strategies to make their decisions to avoid biasing results. For example, we do not explicitly prompt them to pay attention to the colors or shapes in the images. Instead, participants complete at least five hand-picked practice trials to learn the task and receive feedback in all trials. Once they have successfully solved the practice trials, they are admitted to the main experiment, in which they see 40 real trials interspersed with five fairly obvious catch-trials. See Appx. A.2 for details on how trials are created. In all trials, subjects give a binary response and rate their confidence in their decisions on a three-point Likert scale. For each investigated model, we recruit at least 63 unique participants who complete trials for 84 randomly selected units of each model (see Appx. A.5). This means every unit is seen by 30 different participants. Within each task, no unit is shown more than once. We ascertain high data quality through two measures: First, by restricting the worker pool to experienced and reliable workers. Second, by performing quality checks and excluding participants who show signs of not paying attention, such as failing to get all practice trials correct by the second attempt, failing to pass catch trials, taking too long, or being unreasonably quick. We also forbid workers to participate multiple times in our experiments to avoid biases introduced through learning effects. We keep recruiting new participants until 63 workers pass our quality checks per model. See Appx. A.3 for details.

We finally refer to the ratio of correct answers as *interpretability score* and use it as a measure of a unit's interpretability. As there are two options participants have to choose from, random guessing amounts to a baseline performance of 0.5. We record $> 130'000$ responses from $> 1'900$ unique participants recruited over Amazon Mechanical Turk for 767 units spread across 9 models. For more details, refer to Appx. A.1.

3.2 Scaling Feature Visualization to Many Models

Feature visualization describes the process of synthesizing maximally activating images through gradient ascent on a unit's activation. While simple in principle, this process was refined to produce the best-looking visualizations (see Sec. 2). However, these algorithmic design choices and the required hyperparameters have predominantly been optimized for a single model — the original GoogLeNet. This poses a challenge when creating synthetic feature visualizations for different models, as required for a large-scale comparison of models such as ours: How should these hyperparameters be chosen for each model individually without introducing any biases to the comparison? While we cannot revisit all algorithmic choices, we develop an optimization procedure for setting the most crucial parameters, i.e., the number of optimization steps and the strength of the regularizer responsible for creating visually diverse images. In a nutshell, we stop optimization based on the achieved relative activation value and perform a binary search over the latter hyperparameter, to obtain feature visualizations that

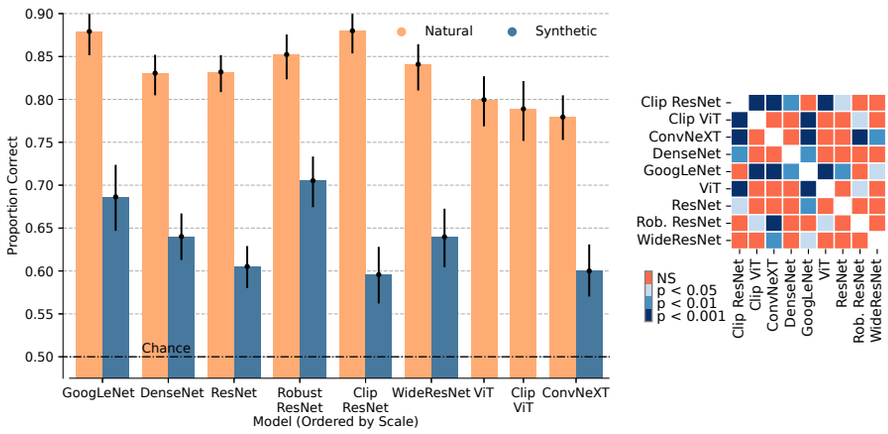


Figure 3: **Left. Model size and training schemes have little influence on per-unit mechanistic interpretability.** We compare the mechanistic interpretability of the units of nine vision models for two interpretability methods: maximally activating dataset samples (Natural) and feature visualizations (Synthetic). In a large-scale psychophysical experiment, we compare models that differ in architecture, training objectives, and training data. While these models reflect the advancements in model design in recent years (sorted by model size first and then dataset size), we surprisingly see little to no effect of these design choices on mechanistic, per-unit interpretability. While these results might appear promising as all models yield scores of about 80 % (natural), note that we demonstrate that interpretability is far more limited than it first appears and breaks down dramatically as the task is made harder in Sec. 4.4. Also, note that error bars represent confidence intervals around the estimated means, not variance of the underlying data (see also Sec. 4.5). **Right. Few models have significantly different interpretability scores.** The differences across models in interpretability afforded by natural exemplars are mostly non-significant (NS) in a Conover test with Holm correction for multiple comparisons; see Fig. 11 for significance values for synthetic feature visualizations.

are comparable in terms of how well they activate a unit. For details, see Appx. A.4. Unfortunately, there is no generally accepted method for generating feature visualizations for ViT models yet: While Ghiasi et al. [16] present a method to generate visualizations for ViTs, we refrain from using it because one of the steps of their procedure seems hard to justify (see Appx. A.4).

4 Results

We now present and analyze the data we obtained through our psychophysical experiment. We look at how scaling models affects mechanistic interpretability (Sec. 4.1), compare feature visualizations and exemplars (Sec. 4.2), investigate systematic layer-dependence of interpretability (Sec. 4.3), and investigate the dependence of our results on task difficulty (Sec. 4.4). Lastly, we introduce a dataset bundling the experimental data that we hope can lead to new avenues for mechanistic interpretability research (Sec. 4.5). Unless noted otherwise, error bars correspond to the 95th percentile confidence intervals of the mean of the unit average estimated through bootstrap sampling.

4.1 Scaling Models Does not Coincide with Improving Interpretability

We begin by visualizing the interpretability of the nine networks investigated in Fig. 3 for both the natural and the synthetic conditions. We sample models with different levels of scale (in terms of model or dataset size) and different training paradigms, but find little to no difference in their interpretability. Strikingly, the latest generation of vision models (i.e., ConvNeXT and ViT) performs *worse* than even the oldest model in this comparison (GoogLeNet).

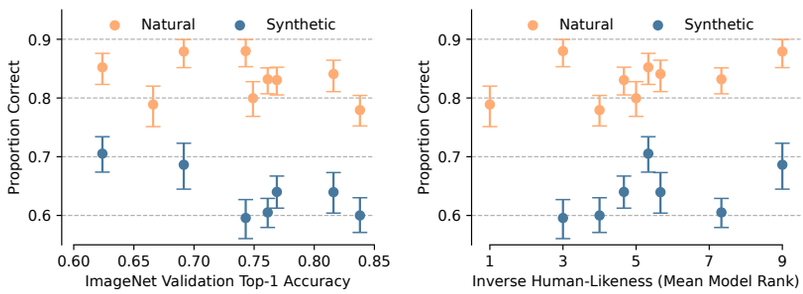


Figure 4: **Neither higher classification performance nor more human-like decisions come with higher interpretability. Left.** While the investigated models have strongly varying classification performance, as measured by the ImageNet validation accuracy, their interpretability shows less variation for both natural exemplars (orange) and synthetic feature visualizations (blue). More accurate classifiers are not necessarily more interpretable. For synthetic feature visualizations, there might even be a regression of interpretability with increasing accuracy. **Right.** A similar result is obtained when quantifying the similarity models have to human behavior. This similarity is measured by the mean rank statistic of the model-vs-human benchmark [14], with a lower rank meaning that the model is more human-like.

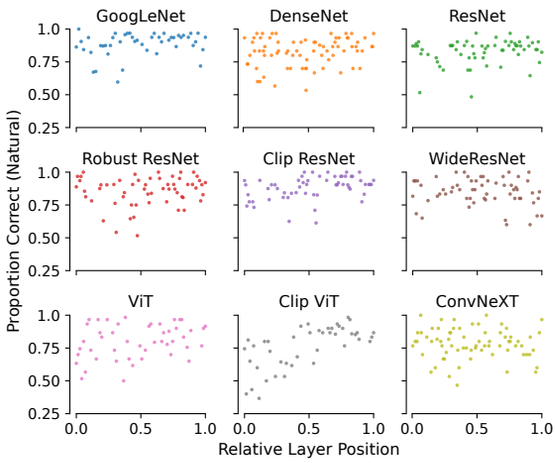
We similarly see no improvements if we plot a model's interpretability as a function of how similar it behaves to humans. For this, we use two metrics: For one, the model's classification performance on ImageNet, for another, a measure of consistency between a model's and human decisions [14]. In Fig. 4, we investigate the relationship between these two similarity measures and a unit's intepretability for both feature visualizations and natural exemplars. While models vary widely in terms of their classification performance ($\sim 60\,\%$ to $\sim 85\,\%$), their interpretability varies in a much narrower range for each method (see Fig. 4a (Left)). For feature visualizations, we see a decline in interpretability as a function of classification performance. For natural exemplars, we do not find any dependency between interpretability and classification performance. We find analogous results for the other similarity metric (see Fig. 4b (Right)). These results highlight that mechanistic interpretability, of the kind investigated here, does not directly benefit from scaling effects, neither in model nor dataset size.

4.2 Feature Visualizations are Less Helpful than Exemplars for all Models

The data in Fig. 3 clearly shows that the findings by [4] generalize to models other than GoogLeNet: Feature visualizations do not explain unit activations better than natural exemplars, regardless of the underlying model. This includes adversarially robust models, which have previously been argued to increase the quality of feature visualizations [11, 49]. The idea was that for non-robust models, naive gradient ascent in pixel space leads to adversarial patterns. To overcome this problem, various image transformations, e.g., random jitter and rotations, are applied to the image over the course of feature visualization. As adversarially more robust models have less adversarial directions, one can hope to obtain visualizations that are visually more coherent and less noisy. There is indeed a substantial and significant increase in performance in the synthetic condition for the robust ResNet-50 over the normal ResNet-50. In fact, this model significantly outperforms all models except GoogLeNet (see Fig. 11). Nevertheless, it remains true that natural exemplars are still far more helpful. To see whether well-interpretable units for one interpretability method are also well-interpretable for the other, we visualize them jointly in Fig. 12. Here, we find a moderate correlation between the two for a few models but no general trend.

4.3 Which Layers are More Interpretable?

In light of the small differences between models regarding the average per-unit interpretability, we now zoom in and ask whether there are rules to identify well-interpretable units within a model.



Model	r
GoogleNet	.28
DenseNet	.17
ResNet	.16
Rob. ResNet	.09
Clip ResNet	.39**
WideResNet	−.16
ConvNeXT	−.09
ViT	.26
Clip ViT	.64***

Figure 5: **The position of a layer is sometimes predictive of its interpretability.** We investigate the interpretability afforded by natural exemplars as measured in our psychophysical experiment by visualizing it for different units of various layers for all investigated networks as a function of their relative position within the network. Here, the first layer corresponds to a relative position of 0, whereas the last layer has a position of 1. The table shows Spearman's rank correlation between the proportion correct (averaged over multiple units from the same layer) and the layer position. Asterisks denote significant correlations using the thresholds shown in Fig. 3b (Right).

A unit's interpretability is not well predicted by its layer's position relative to the network depth (i.e., early vs. late layers). In Fig. 5, we visualize the recorded interpretability scores for all investigated layers as a function of their relative position.[2] We average the interpretability over all investigated units from a layer to obtain a single score per layer. To check for correlations between layer position and interpretability, we compute Spearman's rank correlation for the data of each model. For most models, we do not see a substantial correlation. However, two notable outliers exist: the Clip ResNet and Clip ViT. A strong and highly significant correlation can be found for both of them. We find much smaller correlations for the same architectures trained on smaller datasets (i.e., ResNet and ViT, trained on ImageNet-2012). We thus conclude that (pre-)training on large-scale datasets might benefit the interpretability of later layers while sacrificing that of early layers.

4.4 Do our Findings Depend on the Difficulty of the Task?

As outlined in Sec. 3.1, the difficulty of the task used to quantify interpretability depends on how the query images (i.e., the images that participants need to identify as the more/less strongly activating image) are sampled. So far, we have made the task as easy as possible: The query images were chosen as the most/least strongly activating samples from the entire ImageNet dataset. In this easy scenario, the models were all substantially more interpretable than a random black box (for which we would expect a proportion correct of 0.5). We now ask: Are these models still interpretable in a (slightly) stronger sense, or do their decisions become incomprehensible to humans when increasing the task's difficulty ever so slightly? For this, we repeat our experiment for two models (ResNet-50 and Clip ResNet-50) with query images that are now sampled from the 99th (medium difficulty), 95th (hard difficulty) or 85th (very hard difficulty) percentile of the unit's activations. As the interpretability scores for synthetic feature visualizations are already fairly low in the previously tested easy condition (see Fig. 3a (Left)), we do not test them in the hard condition. Note that the reference images serving as explanations are always chosen from the very end of the distribution of activations, i.e., they are the same for all three difficulties.

[2]Note that the layer position is not precisely defined for layers computed in parallel, e.g., in the Inception blocks of the GoogLeNet architecture.

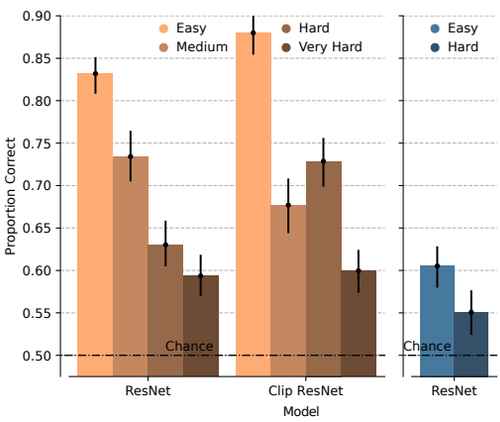


Figure 6: **Human performance decreases with increasing task difficulty.** We increase the task difficulty by not using the most strongly/weakly activating images as the query images (easy) but instead sampling them from the 99th (medium), 95th (hard) or 85th (very hard) percentile. We see a decrease in human performance with increasing difficulty. Strikingly, even a small change in the sampling (easy vs. medium) leads to stark performance decreases when using natural exemplars (left), showing that human understanding of a unit's overall behavior is relatively limited. For the synthetic feature visualizations, the performance is reduced close to chance level by this small change (right).

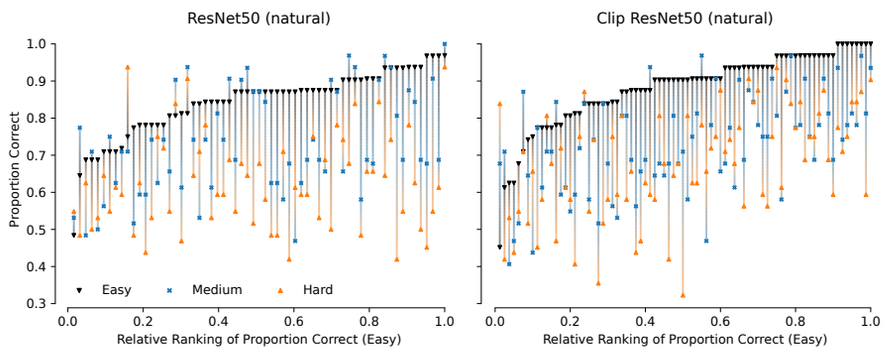


Figure 7: **Well-interpretable units do not necessarily stay interpretable in harder tasks.** We visualize the human performance for each unit investigated of the (Clip) ResNet-50 for the easy (black), medium (blue), and hard (orange) tasks in the natural condition. The units are ordered by the recorded proportion correct values in the easy task. As expected, the performance for almost all units decreases with increasing hardness. However, how much the performance drops is not strongly correlated with performance in the easy task, i.e., well-interpretable units in the easy condition do not necessarily stay well-interpretable in the harder task. For an alternative visualization that displays the gap between the difficulty levels separately, see Fig. 10.

The results in Fig. 6 show a drastic drop in performance when making the task only slightly more difficult (medium). For the synthetic feature visualizations, performance is reduced close to chance level. When looking at how the performance changes per unit (see Fig. 7), we see that for almost all units, the measured interpretability scores do indeed follow the defined difficulty levels, meaning that humans perform best in the easy and worst in the hard task.

But is this a fair modification of the task or does it make the task unreasonably difficult? If the distribution of activations for a unit across the entire dataset was multimodal with small but pronounced peaks at the end for strongly activating images and if we assume each of these modes corresponds to different behavior, making the task harder as described above would be unfair: When the query images are sampled from the 95th percentile while the reference images are still sampled from the distribution's tail, these two sets of images could come from different modes, which might correspond to different types of behavior, making the task posed to participants less meaningful. However, we find a unimodal distribution of activations that smoothly tapers out (see Fig. 16). In other words, the query images used in the harder conditions are in the same mode of unit activation as the ones from the easy condition, and we would, therefore, expect them to also be in a similar behavioural regime.

4.5 IMI - A Dataset to Learn Automated Interpretability Measures

The results above paint a rather disappointing picture of the state of mechanistic interpretability of computer vision models: Just by scaling up models and datasets, we do not get increased interpretability for free, suggesting that if we want this property, we need to *explicitly* optimize for it. One hurdle for research in this direction is that experiments are costly due to the requirement of human psychophysical evaluations. While those can be afforded for some units of a few models (as done in this work), it is infeasible to evaluate an entire model or even multiple models fully. However, this might be required for developing new models that are more interpretable. For example, applying the experimental paradigm used in this work to each of the roughly seven thousand units in GoogLeNet would amount to obtaining more than 200 thousand responses costing around 25 thousand USD. One conceivable way around this limitation is to remove the need for human evaluations by developing *automated* interpretability evaluations aligned with *human* judgments. Put differently, if one had access to a model that can estimate the interpretability of a unit (as perceived by humans), we could potentially leverage this model to directly optimize for more interpretable models.

To enable research on such automated evaluations, we release our experimental results as a new dataset called *ImageNet Mechanistic Interpretability* (IMI). Note that this is the *first* dataset containing interpretability measurements obtained through psychophysical experiments for multiple explanation methods and models. The dataset contains $> 130'000$ anonymized human responses, each consisting of the final choice, a confidence score, and a reaction time. Out of these $> 130'000$ responses, $76'000$ passed all our quality assertions while the rest failed (some of) them.[3] We consider the former to be the main dataset and provide the latter as data for development/debugging purposes. Furthermore, the dataset contains the used query images as well as the generated explanations for 767 units across nine models.

The dataset itself should be seen as a collection of labels and meta information without fixed features that should be predictive of a unit's interpretability. While there seem to be no large differences between models, there are considerable differences between individual units, even within the same model (e.g., see Fig. 5). Finding and constructing features that are predictive of these differences will be one of the open challenges posed by this line of research. We illustrate how this dataset could be used by trying to predict a unit's interpretability from the pattern of its activations in Appx. B.4 in two examples: First, we test the hypothesis that easier units are characterized by a clearly localized peak of activation within the activation map, while for harder units, the activation is more distributed, making it harder for humans to detect the unit's sensitivity. However, we do not find a reliable relationship between measures for the centrality of activations, e.g. the local contrast of activation maps, and the unit's interpretability. Second, we analyze whether more sparsely activated units, i.e., units sensitive to a very particular image feature, are easier to interpret as the unit's driving feature might be easier to detect and understand by humans. Similar to the other hypothesis, we also do not find a meaningful relation between the sparseness of activations and a unit's interpretability.

We deliberately do not suggest a fixed cross-validation split: Depending on the intended use case of models fit on the data, different aspects must be considered resulting in other splits. For example, when building a metric that has to generalize to different models, another split might be used than when building a measure meant to work for a single model only. For that reason, we recommend researchers to follow best practices when training models on our dataset.

5 Discussion & Conclusion

Discussion Due to the costly nature of psychophysical experiments involving humans, we cannot test every vision model but had to make a selection. To perform the most meaningful comparisons and obtain as informative results as possible, we chose the four design axes outlined above and models representing different points along each axis. For some axes, we did not test all conceivable models, such as the largest vision model presented so far [7] as the weights have not been released yet. However, based on the trends in the current results, it is unlikely that the picture would drastically change when considering more models.

[3] Of the $57'310$ rejected responses, $10'570$ were only rejected because they came from crowd workers who participated more than once; see also Appx. A.3.

An explicit assumption of the approach to mechanistic interpretability investigated here is that feature representations are axis-aligned, i.e., features are encoded as the activations of individual units instead of being encoded using a population code. This can be motivated by the fact that human participants do not fail in our experiments completely — they achieve better than chance-level performance. Therefore, this approach of investigating a network does not seem to be entirely misguided, but that alone does not exclude other coding schemes.[4] Furthermore, Fig. 12 reveals that the two interpretability methods we investigated here are only partially correlated, so other explanation methods might come to different conclusions.

Assessing the interpretability of neural networks remains an ongoing field of research, with no clear gold standard yet. This work utilizes an established experimental paradigm to quantify human understanding of individual units within a neural network. While it is possible that the construction of a new paradigm may alter the results, we contend that the employed experimental paradigm closely mirrors how mechanistic interpretability is applied in practice. Additionally, one could argue that the models analyzed in this work are already interpretable — we just have not discovered the most effective explanation method yet. Although this is theoretically possible, it is important to note that we employed the two best and most widely-used explanation methods currently available, and we were unable to detect any increase in interpretability when scaling models up. We encourage further research on interpretability methods.

Conclusion In this paper, we set out to answer the question: Does scale improve the mechanistic interpretability of vision models at the level of individual units? By running extensive psychophysical experiments and comparing various models, we conclude that none of the investigated axes seem to positively affect model interpretability: Neither the size of the model nor that of the dataset nor model architecture or training scheme improve interpretability. This result highlights the importance of building more interpretable models: Unless we explicitly design models with interpretability in mind, we do not get it for free by just increasing downstream task performance. We believe that the benchmark dataset we released can play an important enabling role in this line of research.

Author Contributions

RSZ and WB conceived the idea for the project as a continuation of their earlier work, TK joined at an early stage. RSZ lead the project. WB initiated and supervised the project. RSZ and TK jointly implemented and conducted the experiment, building heavily on the existing setup by RSZ, with advice and feedback from WB. TK contributed code to extend the preparation of natural and synthetic stimuli to support multiple models with help from RSZ. The a priori power analysis was done by TK. RSZ conducted the final analysis and was responsible for the figures with contributions from TK. The manuscript was written jointly by RSZ and TK with advice from WB.

Acknowledgements

We thank Evgenia Rusak, Felix Wichmann, Matthias Kümmerer, Matthias Tangemann, Robert Geirhos and Robert-Jan Bruintjes for their valuable feedback (in alphabetical order) and Max Wolff for his explorative research. This work was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A. WB acknowledges financial support via an Emmy Noether Grant funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1 and via the Open Philantropy Foundation funded by the Good Ventures Foundation. WB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. This research utilized compute resources at the Tübingen Machine Learning Cloud, DFG FKZ INST 37/1057-1 FUGG. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting RSZ and TK.

[4] See work by Elhage et al. [10] for further arguments.

References

[1] Horace Barlow. Single units and sensation: A neuron doctrine for perceptual psychology? *Perception*, 1:371–94, 02 1972. doi: 10.1068/p010371. Cited on page 2.

[2] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *CVPR*, pages 3319–3327. IEEE Computer Society, 2017. Cited on page 2.

[3] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, September 2020. doi: 10.1073/pnas.1907375117. Cited on page 2.

[4] Judy Borowski, Roland Simon Zimmermann, Judith Schepers, Robert Geirhos, Thomas S. A. Wallis, Matthias Bethge, and Wieland Brendel. Exemplary natural images explain CNN activations better than state-of-the-art feature visualization. In *ICLR*. OpenReview.net, 2021. Cited on pages 2, 3, 4, 6, 15, and 16.

[5] Nick Cammarata, Shan Carter, Gabriel Goh, Chris Olah, Michael Petrov, Ludwig Schubert, Chelsea Voss, Ben Egan, and Swee Kiat Lim. Thread: Circuits. *Distill*, 2020. doi: 10.23915/distill.00024. Cited on pages 2 and 3.

[6] Diogo Vieira Carvalho, Eduardo Marques Pereira, and Jaime S. Cardoso. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 2019. Cited on page 3.

[7] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme Ruiz, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd van Steenkiste, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Collier, Alexey A. Gritsenko, Vighnesh Birodkar, Cristina Nader Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetic, Dustin Tran, Thomas Kipf, Mario Lucic, Xiaohua Zhai, Daniel Keysers, Jeremiah J. Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 7480–7512. PMLR, 2023. Cited on pages 2 and 9.

[8] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint*, abs/1702.08608, 2017. Cited on page 3.

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint*, abs/2010.11929, 2020. Cited on page 3.

[10] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. Cited on pages 2 and 10.

[11] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. *arXiv preprint*, abs/1906.00945, 2019. Cited on pages 3 and 6.

[12] Dumitru Erhan, Y. Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *Technical Report, Univeristé de Montréal*, 01 2009. Cited on page 2.

[13] Franz Faul, Edgar Erdfelder, Albert-Georg Lang, and Axel Buchner. G*power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior Research Methods*, 39(2):175–191, May 2007. doi: 10.3758/bf03193146. Cited on page 19.

[14] Robert Geirhos, Kantharaju Narayanappa, Benjamin Mitzkus, Tizian Thieringer, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Partial success in closing the gap between human and machine vision. In *NeurIPS*, pages 23885–23899, 2021. Cited on pages 2 and 6.

[15] Robert Geirhos, Roland S Zimmermann, Blair Bilodeau, Wieland Brendel, and Been Kim. Don't trust your eyes: on the (un) reliability of feature visualizations. *arXiv preprint arXiv:2306.04719*, 2023. Cited on page 3.

[16] Amin Ghiasi, Hamid Kazemi, Steven Reich, Eitan Borgnia, Manli Shu, Micah Goldblum, Andrew Gordon Wilson, and Tom Goldstein. What do vision transformers learn? a visual exploration, 2023. Cited on pages 2, 3, 5, and 17.

[17] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael A. Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89, 2018. Cited on page 3.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. Cited on page 3.

[19] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *arXiv preprint*, abs/2203.15556, 2022. Cited on page 2.

[20] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *NeurIPS*, pages 9734–9745, 2019. Cited on page 3.

[21] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. Cited on page 3.

[22] D H Hubel and T N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.*, 160(1):106–154, January 1962. Cited on page 2.

[23] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. Cited on page 3.

[24] Neha Mukund Kalibhat, Shweta Bhardwaj, C. Bayan Bruss, Hamed Firooz, Maziar Sanjabi, and Soheil Feizi. Identifying interpretable subspaces in image representations. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 15623–15638. PMLR, 2023. Cited on page 3.

[25] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint*, abs/2001.08361, 2020. Cited on page 2.

[26] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018. Cited on page 3.

[27] Sunnie S. Y. Kim, Nicole Meister, Vikram V. Ramaswamy, Ruth Fong, and Olga Russakovsky. HIVE: evaluating the human interpretability of visual explanations. In *ECCV (12)*, volume 13672 of *Lecture Notes in Computer Science*, pages 280–298. Springer, 2022. Cited on page 3.

[28] Matthew L. Leavitt and Ari S. Morcos. Towards falsifiable interpretability research. *arXiv preprint*, abs/2010.12016, 2020. Cited on page 3.

[29] Zachary Lipton. The mythos of model interpretability. *Communications of the ACM*, 61, 10 2016. doi: 10.1145/3233231. Cited on page 3.

[30] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *CVPR*, pages 11966–11976. IEEE, 2022. Cited on page 3.

[31] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *CVPR*, pages 5188–5196. IEEE Computer Society, 2015. Cited on page 2.

[32] Ari S. Morcos, David G. T. Barrett, Neil C. Rabinowitz, and Matthew M. Botvinick. On the importance of single directions for generalization. In *ICLR (Poster)*. OpenReview.net, 2018. Cited on page 2.

[33] Alexander Mordvintsev, Chris Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, 2015. URL https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html. Cited on page 2.

[34] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *ICLR*. OpenReview.net, 2023. Cited on page 3.

[35] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 12 2014. Cited on page 2.

[36] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017. Cited on page 3.

[37] Chris Olah. Mechanistic interpretability, variables, and the importance of interpretable bases, 2022. Cited on pages 2 and 3.

[38] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. Cited on pages 2, 3, and 16.

[39] R Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107, 2005. Cited on page 2.

[40] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021. Cited on page 3.

[41] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 1135–1144, 2016. Cited on page 3.

[42] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y. Cited on page 3.

[43] Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? In *NeurIPS*, 2020. Cited on page 3.

[44] Philipp Schmidt and Felix Biessmann. Quantifying interpretability and trust in machine learning systems. *ArXiv*, abs/1901.08558, 2019. Cited on page 3.

[45] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5B: an open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. Cited on pages 2 and 3.

[46] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626. IEEE Computer Society, 2017. Cited on page 3.

[47] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9. IEEE Computer Society, 2015. Cited on pages 2 and 3.

[48] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv preprint*, abs/1805.12152, 2018. Cited on page 3.

[49] Eric Wong, Shibani Santurkar, and Aleksander Madry. Leveraging sparse linear layers for debuggable deep networks. In *International Conference on Machine Learning*, pages 11205–11216. PMLR, 2021. Cited on pages 3 and 6.

[50] Mengjiao Yang and Been Kim. Benchmarking Attribution Methods with Relative Feature Importance. *CoRR*, abs/1907.09701, 2019. Cited on page 3.

[51] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. In *Deep Learning Workshop, International Conference on Machine Learning (ICML)*, 2015. Cited on page 2.

[52] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint*, abs/1605.07146, 2016. Cited on page 3.

[53] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Revisiting the importance of individual units in cnns via ablation. *arXiv preprint*, abs/1806.02891, 2018. Cited on page 2.

[54] Roland S. Zimmermann, Judy Borowski, Robert Geirhos, Matthias Bethge, Thomas S. A. Wallis, and Wieland Brendel. How well do feature visualizations support causal understanding of CNN activations? In *NeurIPS*, pages 11730–11744, 2021. Cited on pages 3, 4, 16, and 19.

A Methodological Details

A.1 Measuring the Mechanistic Interpretability of Many Models

To measure the interpretability afforded by a model, we extend the paradigm established by [4]. Participants in our study complete a sequence of 2-Alternative-Forced-Choice (2-AFC) trials, where each trial measures the interpretability of one unit of a network. In each trial, participants are presented with two so-called *query images*, sourced from the training set of ImageNet. One query image is highly positively activating for the investigated unit, i.e., feeding this image through the network would cause a large positive activation at the target unit. In contrast, the other query image is highly negatively activating. Participants are tasked with determining which of the two query images is the positive one. To do so, they are presented with two sets of nine *reference images* which characterize the unit. One set contains highly positively activating images, while the other contains highly negatively activating images. In the *natural* condition, these reference images are other natural images, whereas in the *synthetic* condition, the reference images are synthetic images generated by Feature Visualization. See Fig. 8 for an example of one trial in the natural condition. We phrase the task by asking which set of reference images fits the positive query image better so that participants can be completely agnostic with respect to the true semantics of the task. We also do not give overly specific instructions to avoid biasing the participants' behavior. Instead, participants learn the task by completing at least five hand-picked practice trials at the beginning of the experiment. Participants give a binary response and rate their confidence in their decision on a three-point Likert scale.

Figure 8: **Example of one trial.** What a crowd worker sees after having completed one trial: Two query images in the middle, two blocks of nine reference images to the sides, instructions, and feedback in the form of the green frame around the correct query image. Of course, this feedback is shown only after a correct response. In case of an incorrect response, the frame would be red.

A.2 Sampling Images for the Psychophysical Tasks

The difficulty of an individual trial depends to a certain degree on the specific images that are shown in the trial. To avoid biasing the results for an individual unit, we do not only select the single highest/lowest activating image as a query image but instead create t different trials for each unit. For each of these, we collect responses from crowd workers thrice. In the following, we describe the stimuli selection process for positively activating images, with negatively activating images being selected analogously. This procedure is similar to that of Borowski et al. [4], who also illustrate the approach in more detail. First, we select the top $9 \cdot t$ activating images as candidates for reference images, where t is the number of unique trials to be generated. Then, we select the next t images to be used as query images. To ensure that the range of activations yielded by the reference images does not differ across the t tasks, we use the following procedure: We divide the range of candidate images into 9 groups of t images each and create a set of reference images by sampling one image from each of the 9 groups without replacement. We initially create $t = 20$ trials but use only 10 of those, keeping the rest for an anticipated later experiment.

A.3 Amazon Mechanical Turk

Our psychophysical study is conducted on Amazon Mechanical Turk to meet the requirement of scale. To maintain high data quality, we exclude participants who do not fulfill certain criteria. First of all, we restrict participation in our experiment to countries in which workers can be expected to be adequately proficient in English and in which completion of our click-work at the expected hourly wage is not unreasonably more profitable than other work, which we deemed unethical. Specifically, we restrict participation to the USA, Canada, Great Britain, Australia, New Zealand, and Ireland. As a second barrier, we only offer our Human Intelligence Task (HIT) to experienced workers who have submitted at least $2'000$ HITs for which the response was approved. To ascertain high reliability, we further restrict the pool to workers whose approval rate is at least 99%. Of course, we also prevent workers from participating in our experiments more than once[5]. Even if workers meet the aforementioned requirements, they might still be distracted during the experiment or give random answers to quickly finish the experiment (e.g., if they are unmotivated or frustrated due to the task difficulty). Therefore, we filter our data further. To use only data from workers who understand the task, we only accept HITs that require no more than three attempts at solving the demo trials and reject workers who spend less than 15 seconds reading the instructions. To catch workers who click mindlessly, we exclude responses in which fewer than four of our five catch-trials were answered correctly and responses that take the worker less than 135 seconds overall. On the other hand, we also reject responses that take them longer than $2'500$ seconds since it can be assumed that these workers interrupted their work. We also reject responses in which participants select the same query image (as in, the upper / lower one) in more than 90% of trials.

We recruit participants for each investigated model and experimental condition until 63 unique participants pass our quality checks. The responses of the workers who have not passed these checks are not used in our analysis but are included in our IMI dataset. Each participant completes at least 5 practice trials to get used to the task, 40 real trials, and 5 catch trials with obvious, hand-picked stimuli. In total and excluding pilot experiments, we collect data for $133'310$ trials, of which $76'000$ pass all quality checks.

We select 84 units of each model so that every unit is seen by 30 different participants since, within each task, no unit is shown more than once. All procedures conform to Standard 8 of the American Psychological 405 Association's "Ethical Principles of Psychologists and Code of Conduct" (2016). Participants are compensated at a targeted hourly rate of 15 USD, which amounts to 2.79 USD per task.

A.4 Scaling Feature Visualization to Many Models

A fundamental problem with using natural images to characterize the receptive field of individual units (apart from idiosyncrasies of the used dataset) is that visual features do not usually appear in isolation, resulting in ambiguity. For example, highly activating ImageNet-exemplars for a unit sensitive to feathers would probably depict birds, making it hard to isolate feathers as the crucial visual feature instead of beaks, claws, or a background of greenery or blue sky.

The promise of Feature Visualization is to circumvent these limitations by synthetically generating images that only contain visual features contributing to high unit activation. The procedure starts with an initial random noise image and performs gradient ascent on the activation achieved by this image at the unit of interest. Following established work [e.g. 4, 54], a unit is defined as one feature map of a convolutional layer, where the activation across the feature map is aggregated by calculating the mean, just like for natural stimuli. To prevent mode collapse of the generated batch of feature visualizations, i.e. to truthfully capture the receptive field of so-called polysemantic units that show sensitivity to multiple different concepts, a regularization term is added to the loss to diversify the images.

We build on an existing implementation [38] and extend it to support various models flexibly. Previous implementations had two critical hyperparameters: the number of gradient ascent steps to be performed and the weight used for the diversity term. As earlier work mainly focused on the GoogLeNet model, hyperparameters were tuned for it. We find, however, that these fixed values do

[5] Due to technical issues, some workers participated more than once. However, we exclude their data in our analysis and recollect the missing data by recruiting new participants.

not generalize well to other models, but their optimal[6] values heavily depend, among other factors, on the model and location of the unit within the network — in extreme cases, the ideal value can even be different for two units of the same layer in the same network. Therefore, using any fixed value would introduce an unfair bias for or against some models. Furthermore, since a larger weight for the diversity term hinders the optimization, the number of necessary gradient ascent steps depends partially on the diversity weight, meaning these parameters cannot be set independently.

To overcome the latter problem of choosing an appropriate number of optimization steps, we implement an adaptive procedure that interrupts the optimization when the gradients become small. The procedure performs at least $2'500$ steps of gradient ascent and records a trajectory of the observed gradient magnitude. We smooth these trajectories with a large sliding window and halt optimization once the average gradient magnitude in the last window is larger than in the second-to-last window.

To solve the first problem, we determine the diversity weight for each unit individually as follows. We first record the maximum and minimum activation achieved by natural dataset samples for the unit. Then, we generate feature visualizations without diversity and assert that they achieved a stronger activation. We then try to find the largest possible diversity value that still produces images that achieve at least as strong activations as all dataset samples. To do so, we first perform an exponential search starting at a diversity of 1, increasing by a factor of 10 in each step. Once the value becomes too large, we perform 6 steps of binary search between the largest diversity value still known to work and the final value tested in the exponential search. If no value tested during the binary search worked, we return the lower bound of the search range, i.e. the images generated in the end are always guaranteed to be at least as activating as the strongest natural images. Generating one batch of Feature Visualizations, i.e., one step of the procedure, takes between two and 90 minutes on an Nvidia 2080Ti GPU, depending mostly on the width of the layer of the unit, since the diversity term scales quadratically. A qualitative comparison of feature visualizations generated for the different models considered in this work can be found in Fig. 9.

For ViTs, feature visualization could theoretically be performed using the same method by maximizing the activation at the position-wise feedforward layers. However, just applying the existing methodology does not lead to visually coherent images. Ghiasi et al. [16] present a method for adapting the procedure to ViTs that seems to produce intelligible images, but one step of their algorithm just adds large-scale noise to the visualizations, effectively performing a random search in image space to find activating images. Removing this augmentation or reducing the scale of the noise leads to unintelligible images again. In light of these issues, we chose not to evaluate ViTs in the synthetic condition.

[6] Judged by the first authors.

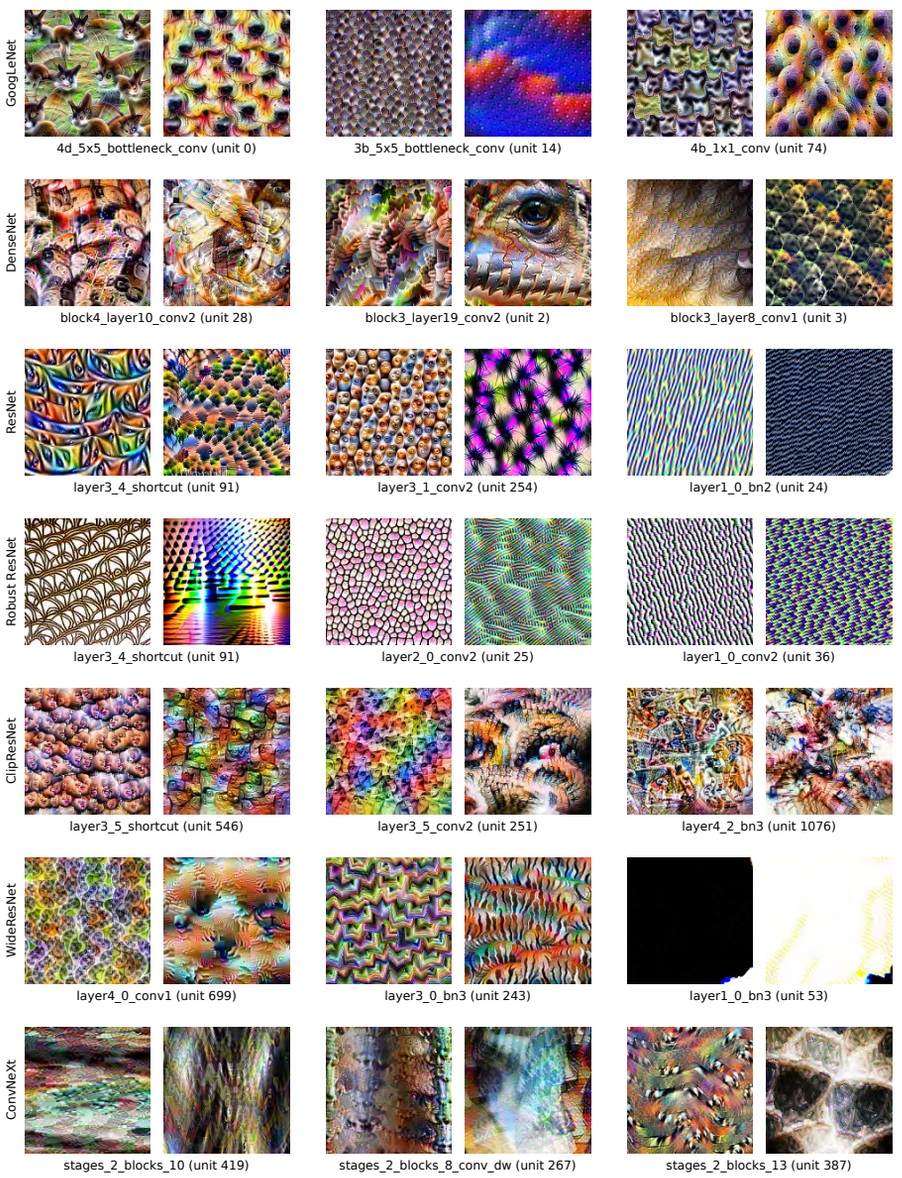


Figure 9: **Qualitative Comparison of Feature Visualizations.** For each model, we randomly choose three units and display the maximally (left) and minimally (right) activating feature visualizations generated without the diversity regularizer.

A.5 A Priori Power Analysis

A central question for the experimental design of this study is how many units need to be sampled per model to obtain a result representative of the entire model. Answering this question is non-trivial as there might be large inter-unit interpretability differences within one model. Indeed, this is what we observe as displayed in Fig. 5). While the most naive approach would be to test all units, this is unfeasible due to the associated financial costs. Therefore, we need to find a trade-off between these considerations and keep the number of sampled units as low as possible while still getting representative results. Put differently: What is the lowest number of units one can select while still being reasonably sure that the found effect is statistically significant?

To answer this question, we first ran a pilot study where we controlled for inter-participant differences by showing stimuli from two models (GoogLeNet and Robust ResNet-50) to the same subjects. Participants in this pilot were the study's first authors and other lab members. This means that the obtained data is of high quality, and we can be confident that all participants understood the task. The mean difference in the proportion of correctly completed trials came out to be 0.1, with standard deviations of 0.15 for both interpretability methods, resulting in a relatively large effect size, with Cohen's d of 0.67. Irrespective of concerns of statistical significance, we deem an effect of this size to be practically relevant; in other words, if the difference in interpretability between two models would be at least 10 percentage points, we would consider this practically relevant. To determine the required number of sampled units at these effect sizes, we then performed an a-priori power analysis using the software G*Power [13] — a standard tool widely used in psychology and the social sciences. To avoid unrealistic assumptions about the shape of the distribution of measurements (the normality-assumption of the t-test will almost certainly not be met because the data points are proportions expected to lie between 0.5 and 1.0), we opted for the non-parametric Mann-Whitney-U test. We assumed an α-level of 0.01 (subject to Bonferroni-correction to safely conduct up to five significance tests on the same data) and a β-level of 0.95. This analysis yields that at least 86 units are required.

However, the situation is further complicated by the fact that we are comparing values of which we cannot actually take a continuous measurement since we aggregate binary trials to estimate the proportion of correctly completed trials for each unit, i.e. there is measurement noise. This can be modeled as a Binomial distribution, characterized by the parameter p, the probability of answering correctly in any given trial for units of this model. This gives rise to the question of how many measurements we should take per unit to be able to assess an individual unit's interpretability with any confidence. Accepting a standard deviation of 0.1 in the estimate of each unit's p results in 30 independent trials per unit.

Another consideration is how many trials one participant can be asked to complete. Earlier work presented up to 24 trials to each participant under similar conditions [54]. Still, again we might be interested in accurately estimating the participant's performance, and each participant incurs some fixed cost for the time spent instructing them and completing the practice trials. On the other hand, MTurk HITs are typically very short. Constructing long tasks, e.g. of 100 trials or more, would increase the risk of participants losing focus or becoming frustrated and just answering randomly. We deemed 55 trials per participant (40 real trials, 10 instruction trials, and 5 catch trials) a suitable balance of these concerns.

Finally, the required number of participants is the total number of trials divided by the number of trials per participant. The total number of trials is, of course, the number of units times the number of necessary measurements per unit, resulting in $86 \cdot 30/40$ trials. As this is not an integer, we opt for using 84 units instead, which brings the number of needed participants to 63.

B Further Experimental Results

B.1 Extended Visualizations of Results in Sec. 4

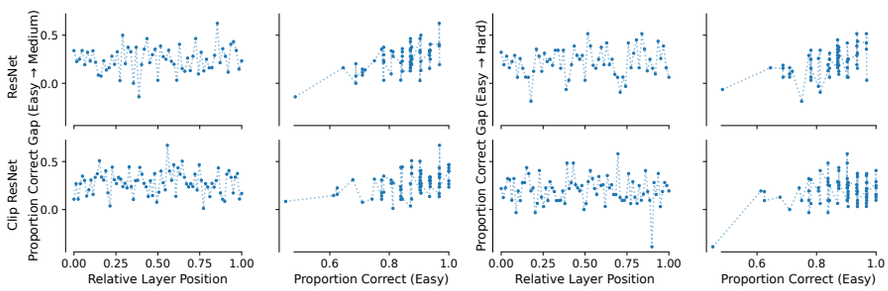


Figure 10: **Well-interpretable units do not necessarily stay interpretable in harder tasks.** For each unit investigated of the ResNet-50 (first row) and the Clip ResNet-50 (second row) model, we visualize the gap in human performance between the easy and medium (first two columns) and the easy and hard (last two columns) tasks. We show these gaps as functions of the relative layer position (first and third column) and of the human performance in the easy condition (second and fourth column).

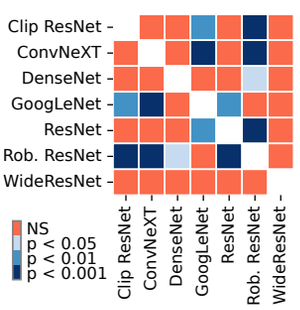
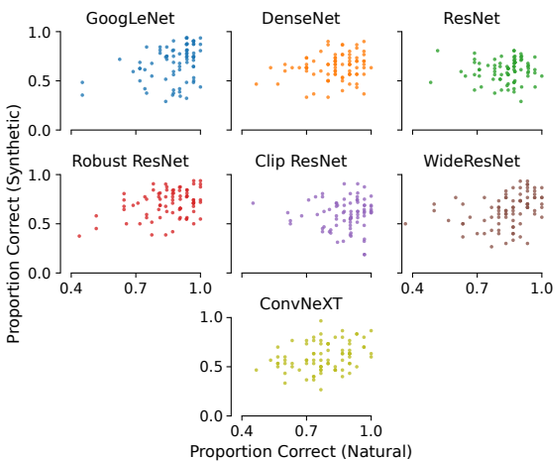
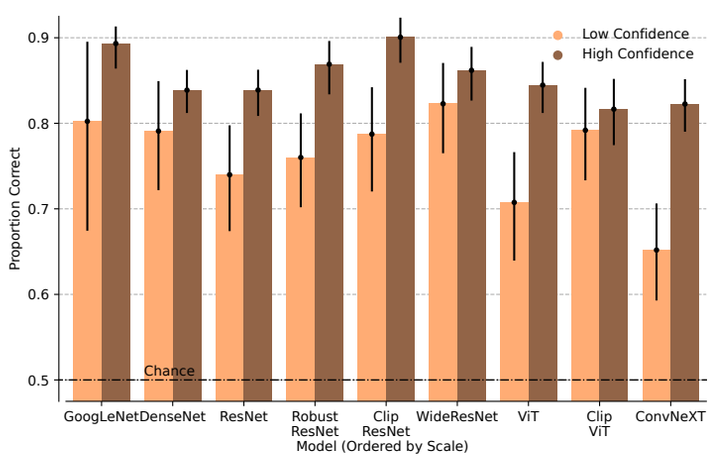


Figure 11: **Few models have significantly different interpretability scores.** The differences in interpretability afforded by synthetic feature visualizations are mostly non-significant (NS) in a Conover test with Holm correction for multiple comparisons; see Fig. 3 for significance values for natural exemplars.

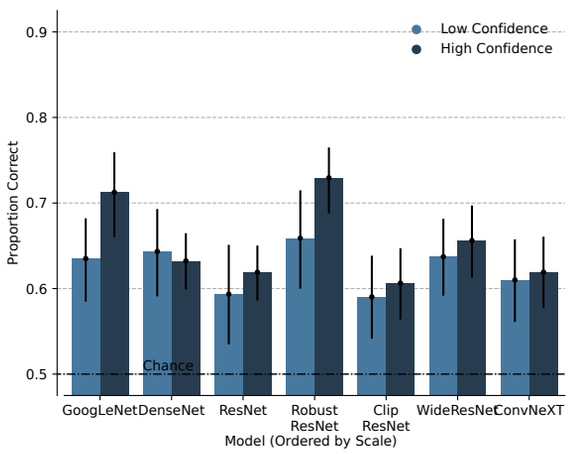


Model	r
GoogLeNet	$.37^{***}$
DenseNet	$.18$
ResNet	$.11$
Rob. ResNet	$.27^{*}$
Clip ResNet	$.11$
WideResNet	$.46^{***}$
ConvNeXT	$.31^{**}$

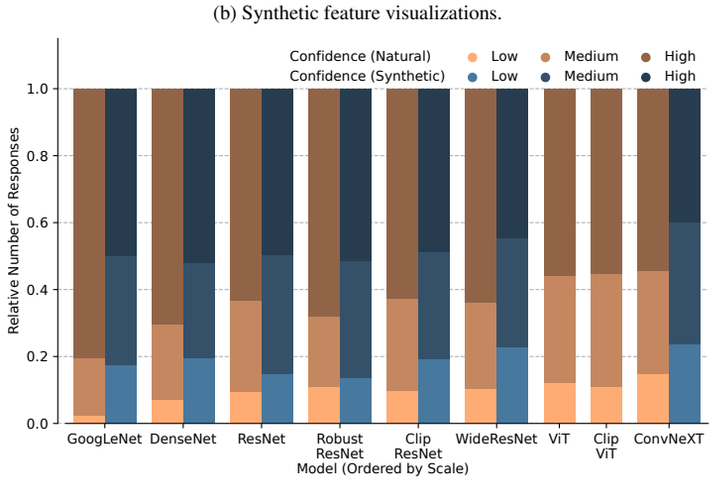
Figure 12: **Measured interpretability using different methods is partially correlated.** We investigate how the interpretability measured in our psychophysical experiment for the explanation method of natural dataset samples is predictive for that measured using synthetic feature visualizations. The table shows Spearman's rank correlation between the proportions correct when using natural and synthetic explanations. Asterisks denote significant correlations. While we see a strong correlation for some models, this does not hold for all.



(a) Natural exemplars.



(b) Synthetic feature visualizations.



(c) Distribution of confidence ratings.

Figure 13: **More confident responses are mostly more correct.** We investigate the relationship between the confidence indicated by the participants and the correctness of the given response. For this, we compare the proportion correct for responses with low (i.e., $= 1$) and high (i.e., $= 3$) confidence ratings for all models and both natural exemplars (a) and synthetic feature visualizations (b). For the natural exemplars (a), we find that for almost all models, a higher proportion of responses are correct when the associated confidence ratings are higher. For the synthetic condition (b), this only holds for two models, if at all. Additionally, the distribution of confidence ratings (c) shows that natural examples lead to higher confidence scores for all models.

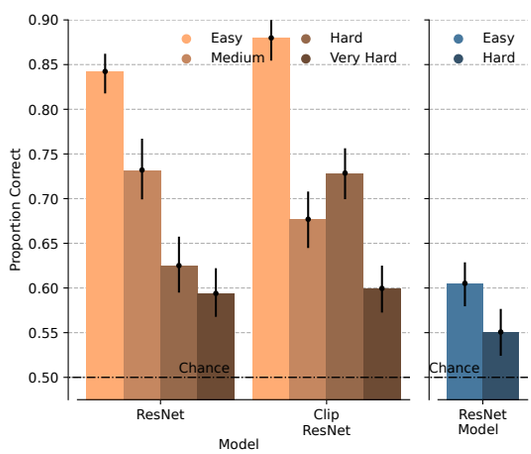
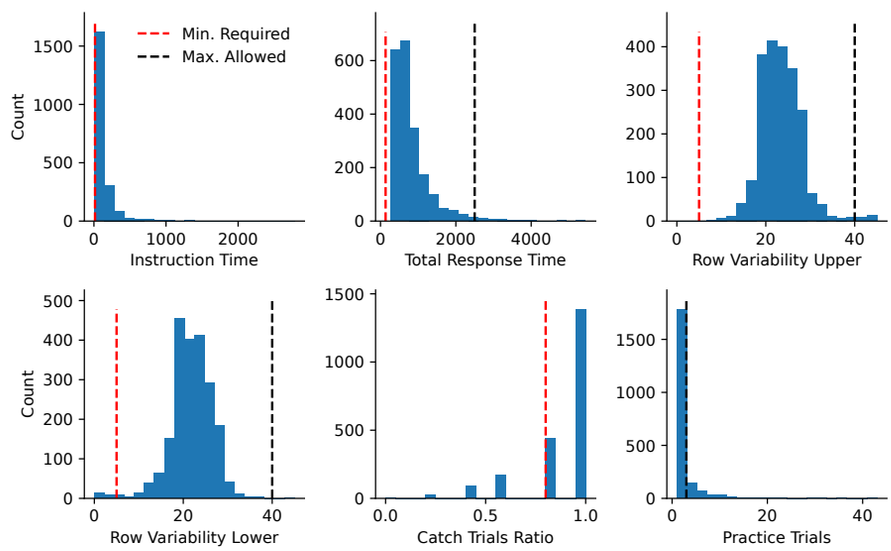


Figure 14: **Impact of unit sampling on performance.** In Fig. 6, we investigate the effect of task difficulty on performance. Due to an oversight, not all 80 units sampled for this experiment were kept identical between the difficulty levels, but only 63. Here, we visualize the result for only those units that were shared between the difficulty levels. The inconsistency has no relevant qualitative effect on the conclusion: Performance rapidly declines as the task becomes harder.

B.2 Analysis of Quality Checks

(a) Distribution of decisions.



(b) Distribution of values used for decision.

Figure 15: **Most participants pass quality checks.** For each of the five quality checks outlined in Appx. A.3, we show a distribution over the number of participants that have passed/failed this check (top) and the distribution over the values used by the checks. The black and red lines in the latter indicate the minimally required and the maximally allowed values, respectively.

B.3 Distribution of Activations

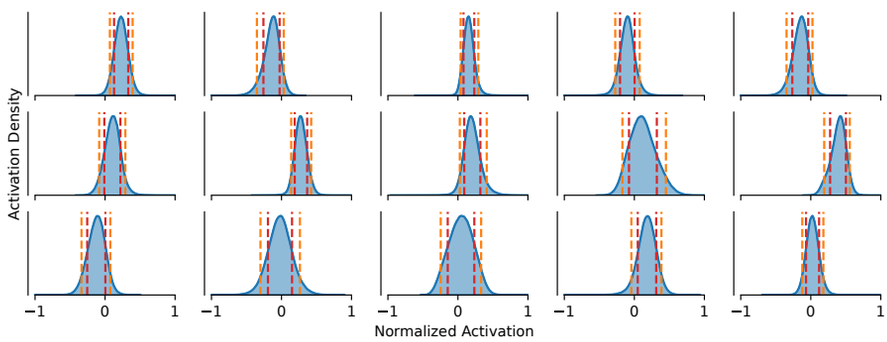


Figure 16: **Activation distribution is unimodal.** We display the distribution of activation for 15 randomly chosen units from GoogLeNet. The activations have been divided by the largest absolute activation per unit to restrict the distribution to values between -1 and 1. The orange and red lines indicate the location of the 85th and 95th percentile as well as that of the 15th and 5th percentile, respectively. It is apparent that the distribution is unimodal and does not feature multiple pronounced peaks/modes at its tail.

B.4 Are Activation Patterns in Feature Maps Predictive of a Unit's Interpretability?

Since we observe large differences in unit-wise interpretability across all networks, a logical research direction is to find out what drives these differences. As an example, we investigate two hypotheses here.

Contrast. First, we investigate whether there is a relationship between a unit's interpretability and the local contrast in the activation maps of convolutional layers caused by validation set images. This is motivated by the idea that if a feature is concentrated at one location in the image, it might be easier to be detected by human observers than if the activation is distributed across the image.

We visualize the relationship between a unit's interpretability and the computed contrast in its activation maps in Fig. 17. There does not appear to be a strong relationship between the two, as supported by low Spearman's rank correlations ($-0.24 \leq \rho \leq 0.14$).

Sparseness. Second, we analyze whether the sparseness of activations in a feature map is predictive of a unit's interpretability. This is motivated by the argument that units that sparsely fire over a large dataset are sensitive to a particular image feature that might be easier for humans to detect and understand.

To test this, we investigate two measures of sparseness: First, we compute the fraction of non-positive values (i.e., zeros after ReLU activation) in a unit's feature map averaged over the ImageNet validation set. The resulting data and the units' interpretability scores are shown in Fig. 18. As for the contrast baseline, we see only a weak, non-significant relation between the two. Second, we compute the fraction of images in the ImageNet validation set for which an entire feature map achieves only non-positive values (i.e., zeros after ReLU activation). Analogously to before, the resulting data is shown in Fig. 19, and we find no strong relationship.

C Broader Impacts

We expect the broader impacts of our work to be positive since advancements made with respect to the interpretability of AI systems should increase their transparency and fairness. However, as is always the case for interpretability work, explanations can also give users a false sense of trust in the explained model. This can lead to the deployment of models that, under real-world conditions, give incorrect or undesired results. Too much trust in AI systems can also lead to their deployment in areas

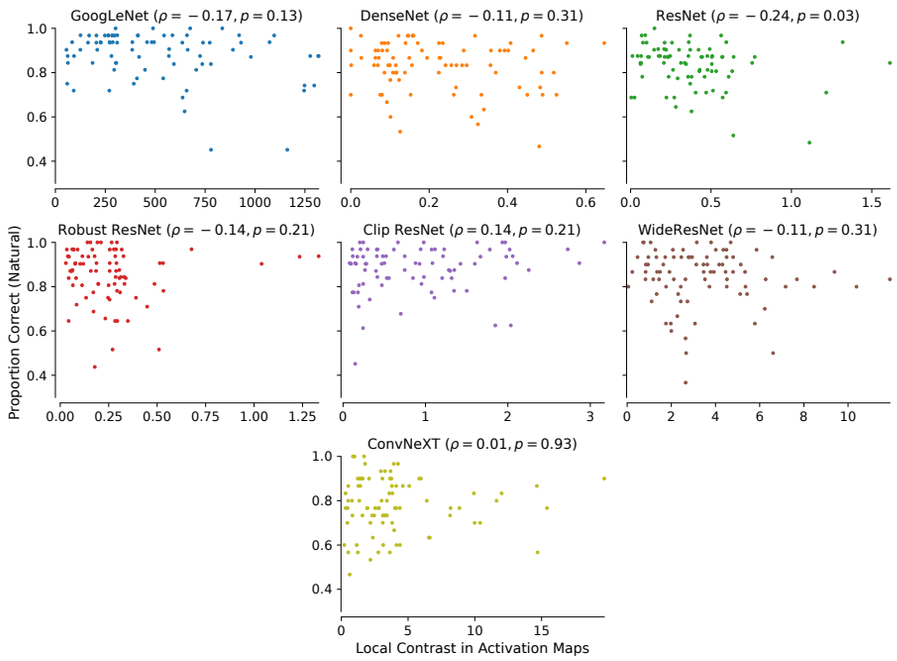


Figure 17: **Local contrast of activation maps does not predict a unit's interpretability.** We compute the average local contrast in the activation maps caused by validation set images for the sampled units of the investigated convolutional networks. The units' interpretability, measured by the proportion correct, does not appear to be a function of the local contrast.

that are better left in human hands for ethical reasons, such as policing or the justice system. Apart from these general and high-level concerns, we see no direct way in which someone could use the findings and data presented here to cause harm, especially since we do not build an interpretability method but investigate whether models are interpretable.

D Computational and Financial Cost

The most computationally intensive aspect of this work is creating stimuli for the experiments, which can be further subdivided into collecting natural exemplars and producing feature visualizations. The former point is negligible since all that is required is one forward pass over the ImageNet training set for each model. We record the activations on Nvidia 2080Ti GPUs and perform multiple forward passes due to memory constraints, but even if we assume a pessimistic 4 hours of GPU time and full utilization of the GPU at 250 W, this results in 9 kWh power consumption for all models in total. Creating feature visualizations for 100 randomly selected units — we later randomly sample 84 units for each model and kept some stimuli for anticipated later experiments — requires the parallel use of 25 2080Ti GPUs for about 12 hours for all models except ConvNeXt, which takes about 24 hours on average. Since this is done for only seven models because we do not generate feature visualizations for the ViTs, the required electricity amounts to 600 kWh. Assuming our country's consumer electricity price of 0.4812 € / kWh and the country's typical CO2 emissions per kWh of 428 g CO2e / kWh, both of which are pessimistic estimates given that the experiments ran in a local academic datacenter, these requirements translate to about 300 USD and 256 kg of CO2 equivalent emissions.

The financial cost of this work is dominated by crowdworker compensations. As outlined in Appx. A.3, workers are compensated at an hourly wage of 15 USD, or 2.79 USD / HIT. Since all workers are

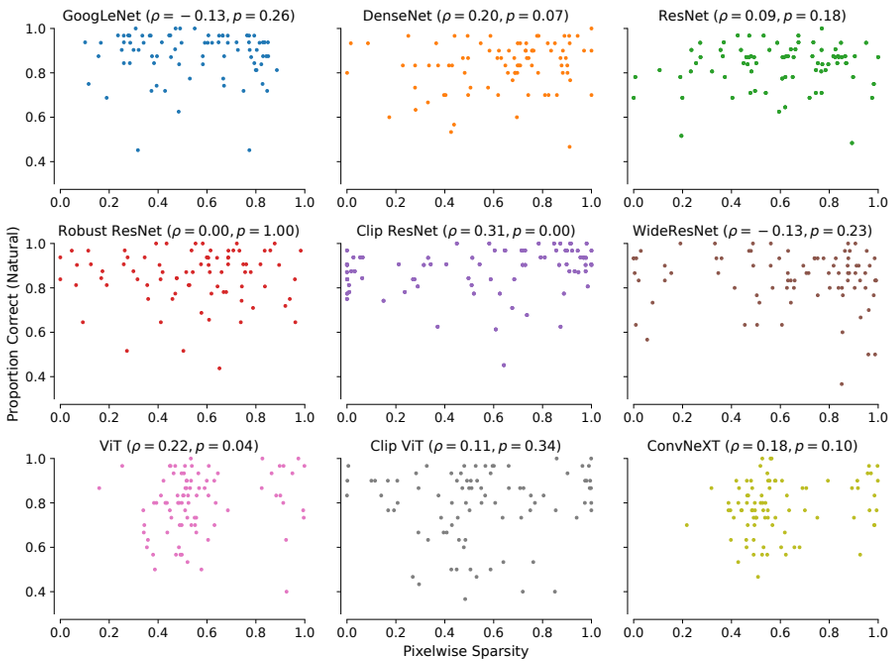


Figure 18: **Sparseness of activations does not predict a unit's interpretability.** We compute the fraction of non-positive values (i.e., zero after ReLU activation) in the feature maps of the units of interest averaged over the ImageNet validation set for all investigated models. We then show a unit's interpretability as a function of this pixel-wise sparseness measure. However, the two do not appear to have a meaningful relationship, as indicated by Spearman's rank correlation shown above each plot.

compensated, even if the results of their HIT do not pass our quality checks, the total cost incurred by the experiment (including the fees paid to MTurk) amounts to around $12'000$ USD.

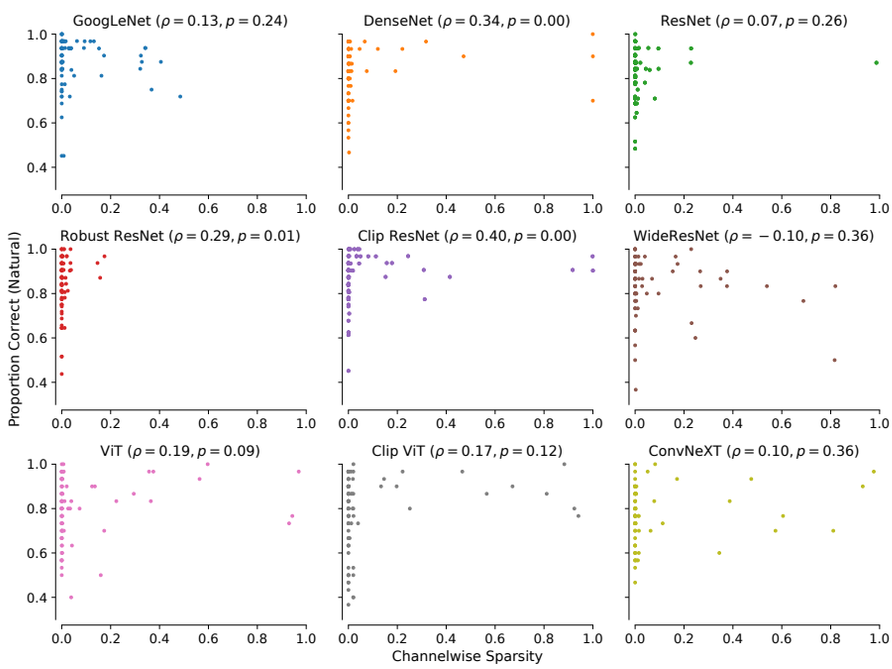


Figure 19: **Sparseness of entire channels does not predict a unit's interpretability.** Similar to Fig. 18, we compute the fraction of images for which an entire feature map achieves only non-positive values (i.e., zero after ReLU activation). Analogously to before, we plot a unit's interpretability as a function of the channel-wise sparseness and find no strong relation between this sparseness measure and a unit's interpretability.

E Further Screenshots of Psychophysics Trials

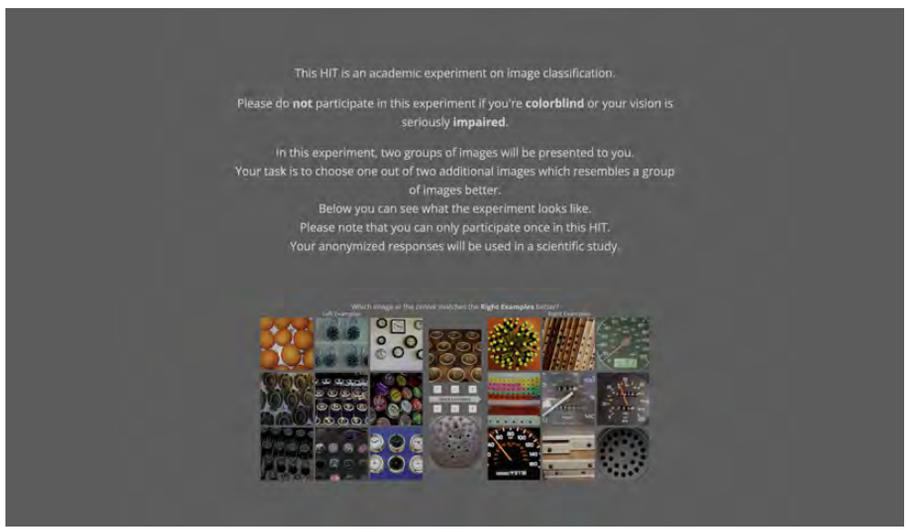


Figure 20: Screenshot of the initial overview of the HIT presented to workers considering the task. We inform participants that they consent to their anonymized data being used for a scientific study.

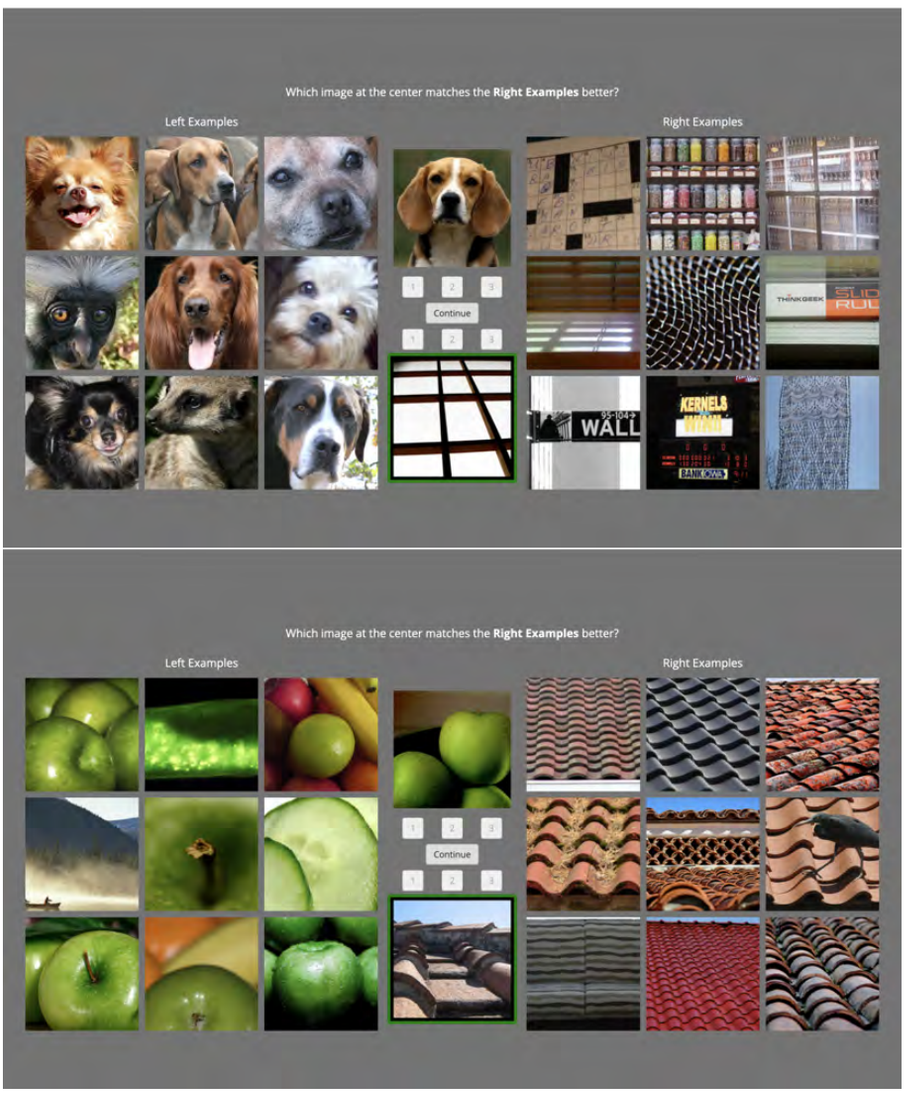


Figure 21: Screenshots of two of the twelve possible instruction trials to explain the task to participants in the natural condition after the participant has given the correct response. See Fig. 22 for examples in the other condition.

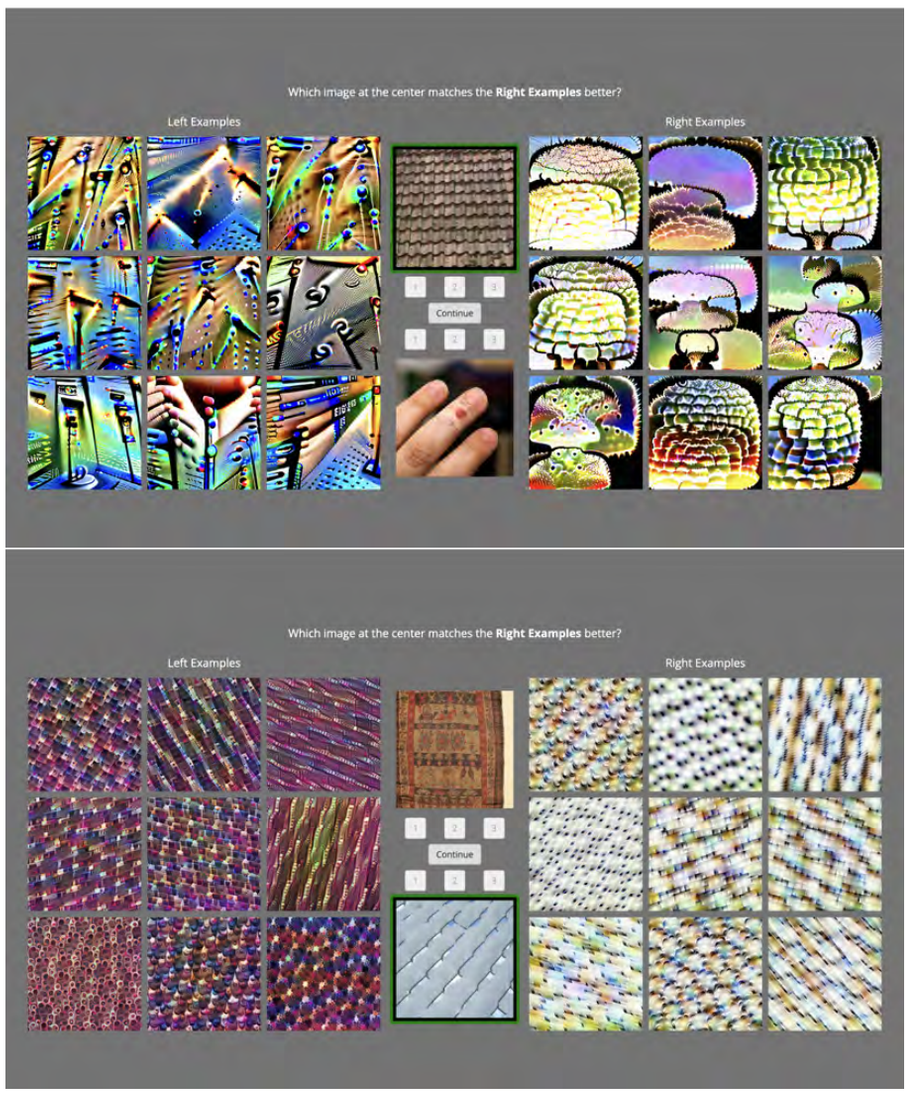


Figure 22: Screenshots of two of the twelve possible instruction trials to explain the task to participants in the synthetic condition after the participant has given the correct response. See Fig. 21 for examples in the other condition.

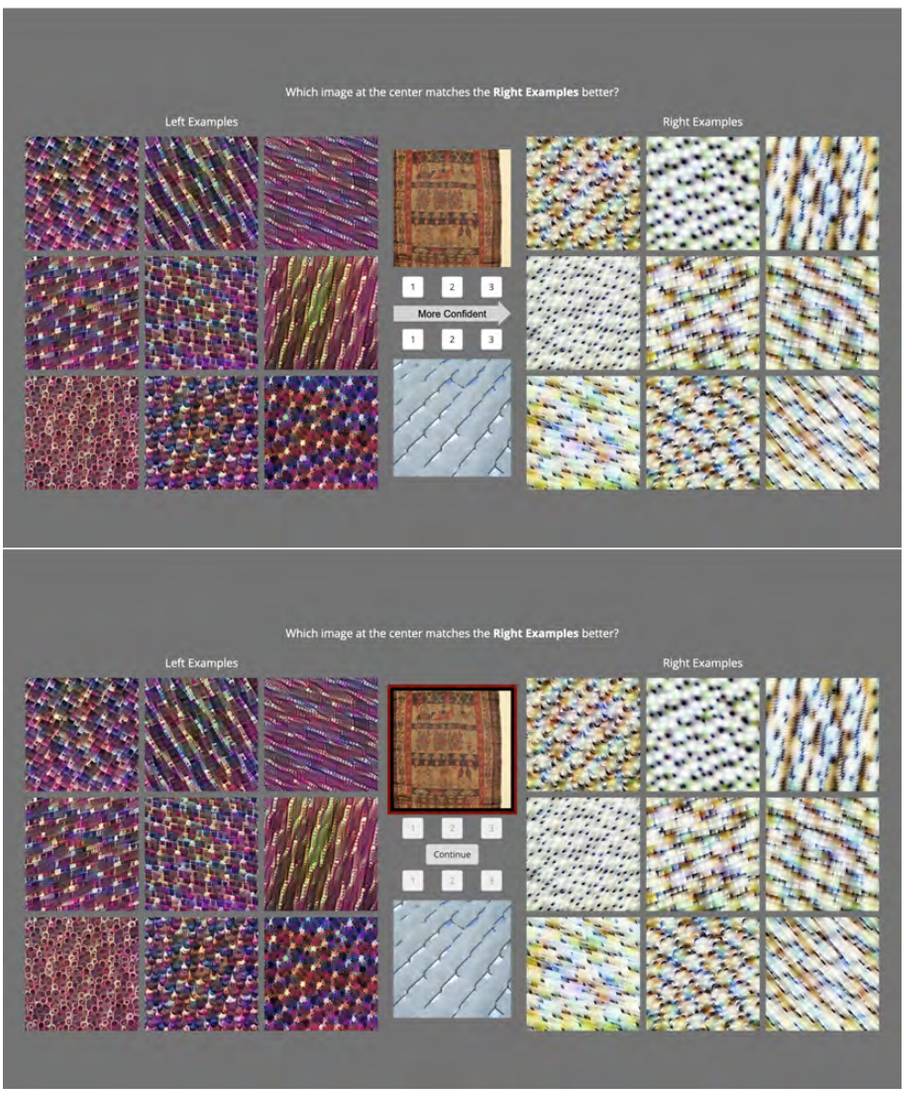


Figure 23: Screenshots of two of the twelve possible instruction trials to explain the task to participants in the synthetic condition before the participant has given a response (top) and after the participant has given the wrong response (bottom).

A.4 Don’t Trust Your Eyes: On The Unreliability of Feature Visualizations

The following 37 pages were published as:

Robert Geirhos*, **Roland S. Zimmermann***, Blair Bilodeau*, Wieland Brendel, and Been Kim. "Don’t Trust Your Eyes: On The Unreliability of Feature Visualizations" *ICML (2024)*

A summary is given in Section 2.2 on page 34. * Equal contribution.

Abstract

How do neural networks extract patterns from pixels? Feature visualizations attempt to answer this important question by visualizing highly activating patterns through optimization. Today, visualization methods form the foundation of our knowledge about the internal workings of neural networks, as a type of mechanistic interpretability. Here we ask: How reliable are feature visualizations? We start our investigation by developing network circuits that trick feature visualizations into showing arbitrary patterns that are completely disconnected from normal network behavior on natural input. We then provide evidence for a similar phenomenon occurring in standard, unmanipulated networks: feature visualizations are processed very differently from standard input, casting doubt on their ability to "explain" how neural networks process natural images. This can be used as a sanity check for feature visualizations. We underpin our empirical findings by theory proving that the set of functions that can be reliably understood by feature visualization is extremely small and does not include general black-box neural networks. Therefore, a promising way forward could be the development of networks that enforce certain structures in order to ensure more reliable feature visualizations.

Don't trust your eyes: on the (un)reliability of feature visualizations

Robert Geirhos [*1] **Roland S. Zimmermann** [*2 3] **Blair Bilodeau** [*4] **Wieland Brendel**[§ 2 3] **Been Kim**[§ 1]

Abstract

How do neural networks extract patterns from pixels? Feature visualizations attempt to answer this important question by visualizing highly activating patterns through optimization. Today, visualization methods form the foundation of our knowledge about the internal workings of neural networks, as a type of mechanistic interpretability. Here we ask: How reliable are feature visualizations? We start our investigation by developing network circuits that trick feature visualizations into showing arbitrary patterns that are completely disconnected from normal network behavior on natural input. We then provide evidence for a similar phenomenon occurring in standard, unmanipulated networks: feature visualizations are processed very differently from standard input, casting doubt on their ability to "explain" how neural networks process natural images. This can be used as a sanity check for feature visualizations. We underpin our empirical findings by theory proving that the set of functions that can be reliably understood by feature visualization is extremely small and does not include general black-box neural networks. Therefore, a promising way forward could be the development of networks that enforce certain structures in order to ensure more reliable feature visualizations.

1. Introduction

A recent open letter called for a "pause on giant AI experiments" in order to gain time to make "state-of-the-art systems more accurate, safe, interpretable, transparent, robust, aligned, trustworthy, and loyal" (Future of Life Institute, 2023). While the call sparked controversial debate, there is general consensus in the field that given the real-world impact of AI, developing systems that fulfill those qualities is no longer just a "nice to have" criterion. In particular, we need *"reliable" interpretability methods* to better understand models that are often described as black-boxes. The development of interpretability methods has followed a pattern similar to Hegelian dialectic: a method is introduced (*thesis*), often followed by a paper pointing out severe limitations or failure modes (*antithesis*), until eventually this conflict is resolved through the development of an improved method (*synthesis*), which frequently forms the starting point of a new cycle. An example of this cycle are saliency maps: Developed to highlight which image region influences a model's decision (e.g., Springenberg et al., 2014; Sundararajan et al., 2017), many existing saliency methods were shown to fail simple sanity checks (Adebayo et al., 2018; Nie et al., 2018), which then spurred the ongoing development of methods that aim to be more reliable (e.g., Gupta & Arora, 2019; Rao et al., 2022).

In contrast to saliency maps and attribution methods like GradCAM (Selvaraju et al., 2017), LIME (Ribeiro et al., 2016) and SHAP (Lundberg & Lee, 2017) where the field has developed a relatively good understanding of their reliability, another central mechanistic interpretability method currently lacks good sanity checks: feature visualizations (Erhan et al., 2009; Mordvintsev et al., 2015; Olah et al., 2017). While attribution/saliency methods attempt to explain how a network responds to an individual sample, feature visualizations attempt to explain the general sensitivity of a unit (e.g., a single channel of a convolutional layer) in a neural network. This is achieved by visualizing highly activating patterns through activation maximization. First introduced by Erhan et al. (2009), feature visualizations have continually been refined through better priors and regularization terms that improve their intuitive appeal (e.g., Yosinski et al., 2015; Mahendran & Vedaldi, 2016; Nguyen et al., 2016; Olah et al., 2017; Fel et al., 2023). Today, feature visualization methods underpin many of our intuitions about the inner workings of neural networks. They have been proposed as debugging tools (Nguyen et al., 2019), found applications in neuroscience (Walker et al., 2019; Bashivan et al., 2019; Ponce et al., 2019), and according to Olah et al. (2017), "to make neural networks interpretable, feature visualization stands out as one of the most promis-

[*]Joint first authors; order between RSZ and BB determined by coinflip [§]Joint senior authors [1]Google DeepMind [2]Max Planck Institute for Intelligent Systems [3]Tübingen AI Center [4]Department of Statistical Sciences, University of Toronto. Correspondence to: Robert Geirhos <lastname@google.com>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

Don't trust your eyes: on the (un)reliability of feature visualizations

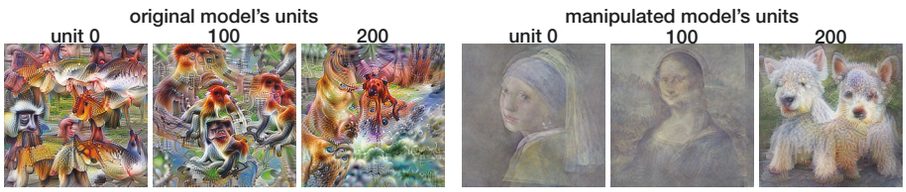


Figure 1: **Arbitrary feature visualizations.** Don't trust your eyes: Feature visualizations can be arbitrarily manipulated by embedding a fooling circuit in a network, which changes visualizations while maintaining the original network's ImageNet accuracy. **Left:** Original feature visualizations. **Right:** In a network with a fooling circuit as described in Section 2.1, feature visualizations can be tricked into visualizing arbitrary patterns (e.g., Mona Lisa).

ing and developed research directions." So what do we know about feature visualization's reliability? Despite its widespread use within the mechanistic interpretability community, relatively little: While they appear to provide some information, humans often struggle to make sense of those visualizations (Gale et al., 2020; Borowski et al., 2021; Zimmermann et al., 2021; 2024). Furthermore, we know that "by itself, feature visualization will never give a completely satisfactory understanding" (Olah et al., 2017), but we don't know to which degree we can trust or rely on them. After initial excitement, many areas of interpretability research have become more cautious and sceptical in general—but scepticism alone is not going to answer important questions such as: Can a method be fooled? How may we sanity-check its reliability? And under which circumstances can the method be guaranteed to be reliable? In this article we provide answers to those three questions:

1. **Adversarial perspective: Can feature visualizations be fooled?** We develop fooling circuits that trick feature visualizations into displaying arbitrary patterns or visualizations of unrelated units. Thus, feature visualizations can be deceived if one has access to the model (Section 2). While this scenario may rarely be plausible, the observation serves as a starting point and motivation for our main empirical and theoretical sections.
2. **Empirical perspective: How can we sanity-check feature visualizations?** We provide a simple sanity check and show that feature visualizations, which are widely used for mechanistic interpretability, are processed largely along different paths compared to natural images, casting doubt on their ability to explain how neural networks process natural images (Section 3).
3. **Theoretical perspective: Under which circumstances is feature visualization guaranteed to be reliable?** Our theory proves that this is only possible if we know a lot about the network already, and impossible if the network is a black-box (Section 4).

We do not mean to imply that feature visualizations per se are not a useful tool for analyzing hidden representations (they are, and it is important to know how individual parts of a neural network function). Instead, we hope that our investigations can help inspire the development of more reliable feature visualizations: a *synthesis* or new avenue.

2. Adversarial perspective: Can feature visualizations be fooled?

One important requirement for interpretability is that the explanations are reliable. We use the following definition of unreliability: A visualization method is unreliable if one can change the visualizations of a unit without changing the unit's behavior on (relevant) test data. More formally, this can be expressed as: Let $\mathcal{U}$ denote the space of all units. A visualization method m is unreliable if $\exists u, v \in \mathcal{U} : m(u) = m(v) \wedge \neg u \overset{bhv}{\sim} v$, where $\overset{bhv}{\sim}$ denotes an equivalence class of equal behavior.

To understand the reliability of feature visualizations, we start by actively deceiving visualizations. For this, we design two different fooling methods: a *fooling circuit* (Section 2.1) and *silent units* (Section 2.2). The motivation for this is twofold. Most importantly, if we can show that one can actively fool feature visualizations, this provides a proof of concept by showing that it is possible to build networks where feature visualizations are completely independent of network behavior on natural images. This concept (different network behavior for natural images vs. feature visualizations) is later investigated for non-adversarial settings both empirically and theoretically. Furthermore, since feature visualizations have been proposed as model auditing tools (Brundage et al., 2020) that should be integrated "into the testbeds for AI applications" (Nguyen et al., 2019, p. 20), it is important to understand whether an adversary (i.e., someone with malicious intent) might be able to construct a model such that feature visualizations are manipulated. This corresponds to a **threat scenario** where the model itself can be arbitrarily changed while the interpretability technique (feature visualization) is kept fixed without control over hyperparameters or the random starting point. For example, a startup may be interested in hiding certain aspects of its model's behavior from a third-party (e.g. regulator) audit that uses feature visualizations. In this context, our

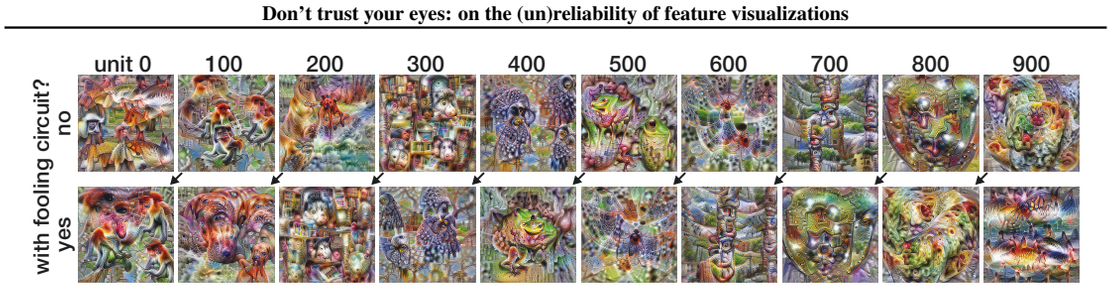


Figure 2: **Using a fooling circuit to arbitrarily permute visualizations. Top row:** Visualizations of the last-layer units in the original Inception-V1 model. **Bottom row:** After integrating a fooling circuit as described in Section 2.1, units show an arbitrarily permuted visualization (here: offset by 100 indices).

demonstration of unreliability relates to a line of work on deceiving other interpretability methods (described in detail in Appendix A.1). We don't know whether such a scenario could become realistic in the future, but as we stated above the main motivation for this section is to provide a proof of concept by showing that one can build networks that deceive feature visualizations.

2.1. Manipulating feature visualizations through a fooling circuit

Our first method to deceive feature visualizations is a *fooling circuit*. It can be embedded in a standard neural network architecture and changes how feature visualizations look without changing the behavior of the network on natural input. By circuit we mean a set of interconnected units carrying out a specific function (Pulvermüller et al., 2014; Olah et al., 2020). In the literature, the term unit either means a single convolutional channel in a convolutional layer or a single neuron u in a fully-connected layer that computes $u(x) = \mathrm{ReLU}(Wx + b)$. For the sake of introducing the fooling circuit, we use the latter definition. We start by taking a standard pre-trained neural network, Inception-V1 (Szegedy et al., 2015), and randomly pick a unit in the last layer (i.e., just before the softmax is applied). When visualizing this unit, denoted F, using the standard visualization method by Olah et al. (2017), we might see, for instance, feathers if this unit corresponds to class "feather" (given that the unit is picked from the last layer, the unit is class-selective since the network was trained to do object classification). The goal of the fooling circuit is to insert a new deceptive network unit A that shows two different modes of behavior: If the network processes natural images, the unit should respond to feathers just like unit F, whereas if feature visualization is performed on the unit, the unit's visualization should depict something completely different, e.g., a donut. We achieve this by wiring six units with ReLU activation functions together as shown in Figure 3.

The fooling circuit has three layers with three, two and one unit(s) each. For the first layer consisting of three units, we start by taking an existing unit D that responds to donuts and the existing unit that responds to feathers F. In the same layer, we attach a binary classifier unit E that operates directly on the network's input. This classifier is not part of the original network. It is trained to tell whether the network's input is a natural image (in which case the classifier output is 1) or a synthetic feature visualization (output 0). Note that this binary classifier has a relatively simple task since feature visualizations start from random noise and end up at a synthetic visualization; thus they are clearly distinguishable from natural images at every step of the process. The middle layer consists of two units (B, C) and the last layer of a single unit A showing deceptive behavior by responding to natural feather images while feature-visualizing a donut. The weights connecting the layers are set as follows: The activation of A is identical to B which is identical to D (the donut unit) for feature visualizations; for natural input, the

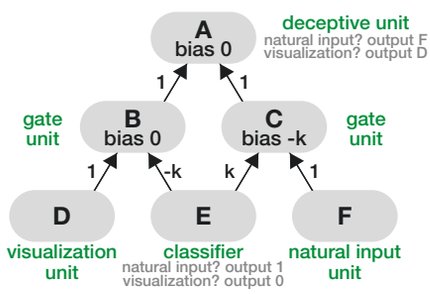


Figure 3: **Fooling circuit.** This circuit consists of six units. Unit A responds like unit F for natural images, but the feature visualizations of A are identical to the ones of D. This is achieved by a classifier unit (E) distinguishing between natural and visualization input, and two intermediate units with ReLU nonlinearities (B and C) selectively suppressing information depending on the classifier's output. k is an arbitrary large constant that ensures the gradient flows only through either the left or the right part of the circuit, not both, by pushing B or C's pre-ReLU activation below zero.

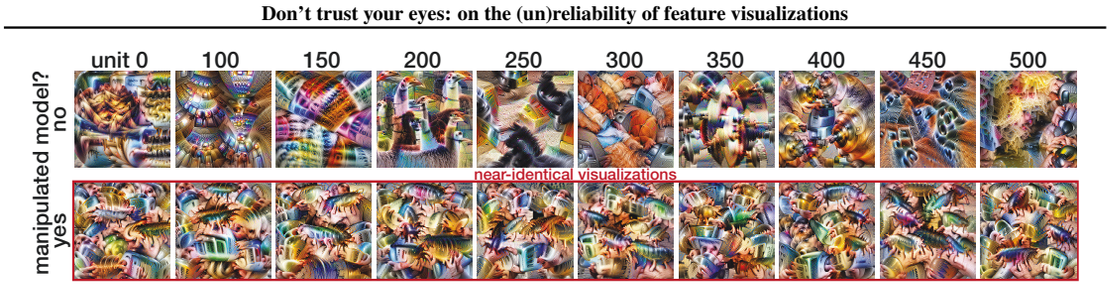


Figure 4: **Leveraging silent units to produce identical visualizations throughout a layer.** The top row shows feature visualizations for units of a layer (block 4-1, conv 2) in a standard, unmanipulated ResNet-50. For the bottom row, we manipulate the model such that the feature visualizations of all units become near-identical (indicated by the red box). Nevertheless, the units still perform the same computations as in the original model on natural input, as evident by an unchanged validation loss. This is achieved by leveraging orthogonal filters in silent units as described in Section 2.2.

activation of A is identical to C which is identical to F (the feather unit). To distinguish between those two cases, the classifier detects feature visualizations and depending on its output, either unit B or unit C has a zero activation since the classifier activation selectively suppresses the pre-ReLU activations of those units by pushing them below zero. Hence, their activations are exactly zero after the ReLU. As a result, inserting this fooling circuit creates a unit A with donut feature visualizations yet when processing natural images, the unit ignores donuts and only responds to feathers.

Using this canonical fooling circuit, we can not only create a single deceptive unit but also simply replicate the circuit for each unit in the last layer of a model and thus, for instance, arbitrarily permute the visualizations of an entire layer as shown in Figure 2. We can even go so far as to design completely arbitrary visualizations through a small modification to the circuit: instead of connecting an existing unit (like the donut unit D), we can add a new unit D' where the receptive field (a single convolutional filter) is hard-coded to an arbitrary pattern such as the Mona Lisa, as shown in Figure 1. Irrespective of how we manipulate the feature visualizations, the network still responds normally to natural input. This can be verified by checking the network's validation accuracy on ImageNet-1K, which only minimally changes when deceiving all visualizations in the last layer of Inception-V1 (top-1 accuracy changes from 69.146 % to 68.744 %; top-5 from 88.858 % to 88.330 %). The tiny drop in performance is a result of the binary classifier achieving slightly less-than-perfect accuracy (99.49% on a held-out test set) when distinguishing between natural input and visualization input. Experimental details are available in Appendix C.1. In Appendix E we additionally provide evidence that our fooling methods do not change model behavior on out-of-distribution input.

Our fooling circuit shows that **it is possible to maintain essentially the same network behavior on natural input while drastically altering feature visualizations**. In Appendix B.1, we formalize this fooling circuit, and prove that it will always behave in the way we observe in our experiments. Given that this fooling circuit requires training a binary classifier, we next explore an alternative fooling method without this requirement.

2.2. Manipulating feature visualizations by leveraging silent units

Our second fooling method does not require a separate classifier but instead leverages orthogonal filters embedded in *silent units*; i.e., units that do not activate for the entire training set. We designed this method to show that fooling can easily be achieved in different ways and across different architectures. In order to demonstrate that different model families can be fooled, we here consider a different architecture (ResNet-50 (He et al., 2016) instead of Inception-V1) and a randomly selected intermediate layer instead of the last layer (but note that the approach is not specific to a certain architecture or layer). We replace a standard computational block,

$$y = \mathrm{ReLU}(\mathrm{BatchNorm}(\mathrm{Conv}(x, \Theta), \Psi)), \qquad (1)$$

where Θ and Ψ are learned conv / batch norm parameters, with the manipulated computation block

$$\bar{y} = y + \Delta y, \text{ where } \Delta y = \mathrm{ReLU}(\mathrm{Conv}(x, \bar{\Theta}) + b). \qquad (2)$$

Our goal is to set $\bar{\Theta}$ and b such that $\bar{y} = y$ on natural images while the feature visualizations of $\bar{y}$ are not related to those of y and instead dominated by Δy. Since feature visualizations usually lead to substantially larger activations than natural samples, we can exploit this property without requiring an explicit classifier like the fooling circuit did. Note that we do not change the original unit, i.e., how y is computed; this means that Θ and Ψ stay unchanged. Instead, we introduce a new unit and choose its two free parameters $\bar{\Theta}$ and b such that $\Delta y = 0$ for natural inputs but $\Delta y \neq 0$ for feature visualizations. Specifically, we construct $\bar{\Theta}$ as a

linear combination of the weight Θ used to compute y and a sufficiently strong orthogonal perturbation $\Delta\Theta^{\perp}$; that is, $\bar{\Theta} = \alpha\Theta + \beta\Delta\Theta^{\perp}$, where α, β control the relative strength of the filter directions. By choosing a sufficiently negative bias b, we ensure that Δy remains silent (no activation, ensuring that natural input is processed as before) unless y is strongly activated. Letting $\bar{y}_{\text{max. nat}}$ denote the maximal observed activation on natural input for the constructed filter $\bar{\Theta}$, we set $b = -\alpha/\beta\bar{y}_{\text{max. nat}}$. Since we empirically observe a large gap between activations from feature visualizations and natural input, we are able to steer the visualizations to arbitrarily chosen images. We demonstrate this by applying it to a ResNet-50 model trained on ImageNet. In Figure 4, all 512 units in a ResNet layer yield near-identical feature visualization. This has no impact on the overall behavior of the network: neither the top-1 nor the top-5 validation accuracy change at all. Further experimental results are presented in Appendix H. In summary, we developed two different methods that trick feature visualizations into showing arbitrary, permuted, or identical visualizations across a layer. This establishes that **feature visualizations can be arbitrarily manipulated** if one has access to the model.

3. Empirical perspective: How can we sanity-check feature visualizations?

In Section 2, we ask whether an adversary may be able to fool a feature visualization. This serves as a proof of concept - feature visualizations can be fooled, at least under "adversarial" circumstances, they are not an inherently reliable method. This naturally leads to the more pressing practical question: do feature visualizations have a reliability problem in normal, non-adversarial circumstances too? To this end, we designed the sanity check from Section 3.

In Section 2 we have seen that feature visualizations can be fooled under adversarial circumstances. This serves as a proof of concept: feature visualizations can be fooled and at least under adversarial circumstances they are not an inherently reliable method. However, in most cases we do not expect an adversary to manipulate a network. This naturally leads to the more pressing practical question: Do feature visualizations have a reliability problem in normal, non-adversarial circumstances too? If so, how would we be able to check this? In the context of saliency methods, sanity checks have proven highly valuable for investigating method reliability (Adebayo et al., 2018). We here provide an empirical sanity check for feature visualizations to investigate the reliability of feature visualizations under standard, non-adversarial circumstances.

The core idea is simple: Feature visualizations are designed to explain how neural networks process natural input (and this is exactly what the literature states they do, see Appendix A.2). This means that once they are generated, good visualizations should be processed along a similar path as natural images—otherwise they're doing something different. We here provide an empirical sanity check to verify this, and we find that this is not the case. Hence, **feature visualization does not explain how neural networks process natural images.**

Let's say we take a unit from the last layer of a network, a unit for which we know that it responds to "cat" images. If we run feature visualization on this unit and feed the resulting visualization through the network, a good "cat" visualization should show typical cat features and thus activate very similar units as natural cat images. Generally speaking, we expect images from the same class to be processed along a similar path because they share certain features. For instance, all airplanes have wings, and all cats have paws. Some features are shared across classes (e.g., both cat and airplane images may contain a blue sky), and some are more class-specific (airplane: wings, cats: paws). If a neural network contains units that respond to certain features—for instance, one unit responds to paws and a different unit to wings—then natural images of the same class should, to a certain degree, activate similar units; and those units should also become activated when processing a good feature visualization. Conceptually, this approach is motivated by the *fooling circuit* from Section 2.1, where the circuit leads to feature visualizations being disconnected from network behavior on natural input by using different paths for different inputs. Hence, we can proceed by analyzing the following three properties for each layer of a standard network:

- How similarly are natural images from the same class processed (e.g., one cat image vs. another cat image)? This serves as the upper bound: the maximal similarity we can hope to capture with a good feature visualization.
- How similarly are natural images from different classes processed (e.g., a cat image vs. an airplane image)? This serves as a lower bound: the baseline similarity we can expect from processing completely unrelated input.
- How similarly are natural images from a class vs. feature visualizations for the same class processed? This indicates a feature visualization method's reliability.

For the sake of this sanity check, we focus on feature visualizations (Olah et al., 2017) for the last layer of a standard network, Inception-V1. The last layer is a perfect choice for this kind of analysis since in contrast to hidden layers, the units in the last layer have perfectly well-known ground truth selectivity: each unit is selective for one class. In terms of measuring similarity between activations of input x_i and input x_j for a network f in layer l, we compute $\Gamma(f_l(x_i), f_l(x_j))$ where Γ could be any similarity metric. We here use Spearman's rank order correlation but our findings are not limited to this metric—other choices such as

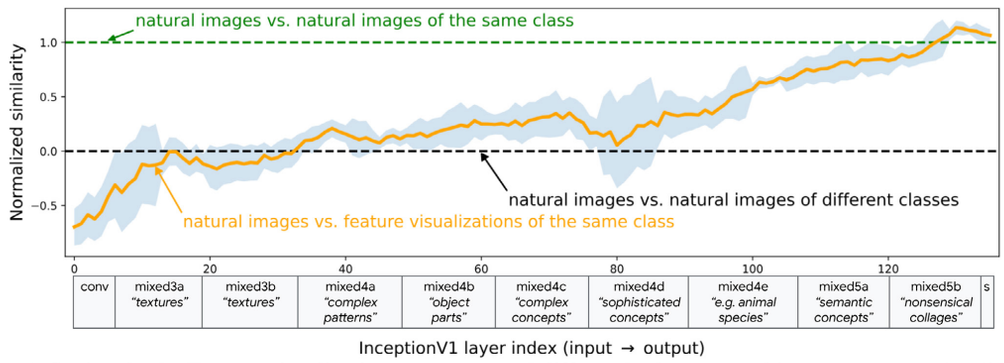


Figure 5: **Sanity check: Feature visualizations are processed differently than natural images.** Feature visualizations are designed to explain how neural networks process natural input—but do feature visualizations for a certain class actually activate similar units as natural input from this class? We measure the similarity of a layer's activations caused by natural images and feature visualizations across layers. Throughout the first two thirds of Inception-V1 layers, activations of natural images have roughly as little similarity to same-class visualizations as they have to completely arbitrary images of different classes. In the last third of the network, similarity increases. Layer annotations (e.g., "textures") are from Olah et al. (2017).

Cosine Similarity or Pearson correlation lead to the same results, as can be seen in Appendix C.3. Using this similarity metric, we can compare whether images of a class are processed similarly to feature visualizations for the same class throughout the network. If so, they should activate roughly the same units in preceding layers (similar activations → high correlation). If they are processed along arbitrary independent paths instead, we would obtain zero correlation. In Figure 5, we plot the results of this analysis, normalized such that a value of 1 corresponds to the Spearman similarity obtained by comparing natural images of the same class (airplanes vs. airplanes, cats vs. cats), and 0 corresponds to the similarity that is obtained from comparing images of one class against images of a different class (airplanes vs. cats etc.,). The results are averaged across classes; raw data and additional information can be found in Appendix C.3.

As can be seen in Figure 5, last-layer feature visualizations are processed differently from natural images throughout most of the network. If they would be processed along the same path, similarity would need to be high across all layers. Later layers have a higher correlation, but that does not mean that the activations are resulting from the same paths. In many earlier and mid-level layers, the activations of, say, cat images are as similar to activations of cat visualizations as they are to activations of flower, airplane or pizza images. While it would be fine for a feature visualization to show different low-level features compared to natural images, any visualization that seeks to explain how natural input is processed should capture similarities in mid- and high-level layers that are described by Olah et al. (2017) as responding to "object parts" and "complex/sophisticated concepts". This means that throughout most of the network, the investigated feature visualization does not pass the sanity check: **processing along different paths means that feature visualizations do not explain how neural networks process natural images**. Looking ahead, we hope that the similarity sanity check we introduce here facilitates rigorous, quantitative evaluation of feature visualizations and guides researchers in designing more reliable feature visualization methods.

Like any sanity check, the method we introduce can only serve as a necessary, but not a sufficient, condition for trustworthiness in the sense that if a sanity check fails, this is evidence for a method being untrustworthy; but if a sanity check passes, this does not guarantee the method's trustworthiness (just like a specific medical check like an X-ray scan is evidence of a problem if for instance a fracture is detected, while "passing" this check does not guarantee that other checks like a blood test would pass too).

4. Theoretical perspective: Under which circumstances is feature visualization guaranteed to be reliable?

Section 2 shows that an adversary can impose arbitrary feature visualizations when they have access to the model, while Section 3 shows that—even without model access—feature visualizations don't explain how natural input is processed. Taken together, this raises a natural question: could a modified version of feature visualization fix these issues? Section 4 theoretically proves that this is not possible: any visualization based on the current dominant approach of maximally activating images won't be able to reliably describe a model's behavior. This is similar to how knowing the maximum of a mathematical function does provide enough information to make accurate predictions about how the rest of the function behaves.

To show this impossibility, we begin with asking: When are feature visualizations guaranteed to be reliable, i.e., guaranteed to produce results that can be relied upon? Feature visualizations are expected to help us "*answer what the network detects*" (Olah et al., 2018), "*understand what a model is really looking for*" (Olah et al., 2017), and "*understand the nature of the functions learned by the network*" (Erhan et al., 2009). When formalizing these statements, two aspects need to be considered. First, the structure of functions to be visualized. The current literature does not place assumptions on the function—it could be any unit in a "*black-box*" (e.g., Heinrich et al., 2019; Nguyen et al., 2019) neural network. Second, we need to characterize which aspects of the function feature visualizations promise to help understand. The existing literature (quotes above and in Appendix A.2) broadly claims that feature visualizations are useful for 'understanding' a function f (such as a unit in a neural network). If that is indeed the case, then a user should be able to use feature visualizations to make meaningful predictions about the behavior of f on some input x. Our theory quantifies this by assessing whether it is possible to predict $f(x)$ for any network input x based on feature visualizations. We investigate three different settings (see Table 1): exact prediction, approximate prediction up to an error ε , and predicting at least whether $f(x)$ is closer to the min- or maximum of f. If none of these can be predicted, then feature visualizations cannot be said to have provided us with any meaningful understanding of the f. We investigate these three scenarios for twelve different function classes, ranging from general black-box functions (no assumptions about the function) to more restrictive settings (e.g., assuming f to be convex).

Conceptually, our theory is based on the insight that feature visualization based on activation maximization seeks to synthesize a highly activating image, which corresponds to finding the $\arg\max$ of f—an insight that might seem trivial. Paradoxically, it is well-known that it is impossible to conclude much, if anything, about an unconstrained function from its $\arg\max$. Yet, feature visualizations are purported to help us understand what black-box functions (e.g., neural network units) detect. To resolve this paradox, we can impose stronger assumptions on the function, or lower our expectations by considering successively weaker notions of understanding, such as instead of asking whether a feature visualization can help predict $f(x)$ simply asking whether it can tell us at least whether the activation for a new test image x will be closer to the maximum or the minimum of the function. In this section, we explore both directions, and show that **even strong assumptions on the function f are *insufficient* to guarantee that feature visualizations are reliable for understanding f, even for very weak notions of understanding.** The core results of our theory are summarized in Table 1; exact definitions for each function class as well as proofs are in Appendix B.

Notation and definitions. We denote the indicator function of a Boolean expression E as $\mathbf{1}_E$, which is 1 if $E(x)$ is true and 0 otherwise. Let d denote the input dimensionality (e.g., number of pixels and channels), $\mathcal{I} = [0,1]^d$ the input space, and $\mathcal{F} = \{\mathcal{I} \to [0,1]\}$ the set of all functions from inputs to scalar, bounded activations.[1] A *maximally activating feature visualization* is the map from $\mathcal{F}$ to $\mathcal{I}^2 \times [0,1]^2$ that returns a function's $\arg\min$, $\arg\max$, and values at these two points, which we denote by $\Phi_{\min\max}(f) = (\arg\min_{x\in\mathcal{I}} f(x), \arg\max_{x\in\mathcal{I}} f(x), \min_{x\in\mathcal{I}} f(x), \max_{x\in\mathcal{I}} f(x))$. When f is clear from context, we write $\Phi_{\min\max} = (x_{\min}, x_{\max}, f_{\min}, f_{\max})$ for brevity. We assess the reliability of a feature visualization by how well it can be used to predict $f(x)$ at new inputs x. To make such a prediction, the user must *decode* feature visualization into useful information. We denote a *feature visualization decoder* as a map $D \in \mathcal{D} = \{\mathcal{I}^2 \times [0,1]^2 \to \mathcal{F}\}$. Our results do not rely on the structure of D in any way. Rather, "No" in Table 1 means that for *every* D the assumptions are insufficient to guarantee accurate prediction of f.

4.1. Main theoretical results

Throughout, we measure the accuracy of predicting f using $\|\cdot\|_\infty$. This is primarily for convenience; the equivalence of L_p norms on bounded, finite-dimensional spaces implies we could prove the same results with $\|\cdot\|_p$ for any p at the expense of dimension-dependent constants. This captures many cases of interest: $\|\cdot\|_p$ for $p \in \{1,2\}$ is the standard measure of accuracy for regression, and for f that outputs bounded probabilities, the logistic loss is equivalent to $\|\cdot\|_2$ up to constants.

First, we note that the boundedness of f implies a trivial ability to predict $f(x)$.

Proposition 1. *There exists $D \in \mathcal{D}$ such that for all $f \in \mathcal{F}$,*

$$\left\| f - D(\Phi_{\min\max}(f)) \right\|_\infty \leq \frac{f_{\max} - f_{\min}}{2}. \tag{3}$$

This means that for any function f, a user can take the feature visualization $\Phi_{\min\max}(f)$ and apply a specific decoder (the constant function taking value halfway between $f_{\min}$ and $f_{\max}$) to predict $f(x)$ for *any* new x. If the user imposed assumptions on f, one might conjecture that a clever choice of decoder could lead to a better prediction of $f(x)$. Our first main result shows that this is impossible even for strong assumptions.

Theorem 1. *For all $\mathcal{G} \in \{\mathcal{F}, \mathcal{F}_{\text{NN}}, \mathcal{F}_{\text{ERM}}, \mathcal{F}_{\text{PAff}}, \mathcal{F}_{\text{Mono}}\}$, $D \in \mathcal{D}$, and $f \in \mathcal{G}$, there exists $f' \in \mathcal{G}$ such that*

[1] For example, class probabilities or normalized activations of a bounded unit in a neural network.

Table 1: **Theory overview.** Feature visualization aims to help understand a function f (e.g., a unit in a network). While understanding is an imprecise term, it can be formalized: Given f and its arg max $x_{\max}$ and arg min $x_{\min}$ (approximated by feature visualization), how well can we predict $f(x)$ for new values of x? We show that this is impossible if f is a black-box. Instead, to make meaningful predictions, we need strong additional knowledge about f.

Stronger assumptions about f ↓		Given feature visualization for a function f and an input x, can we reliably predict… $f(x)$?	$f(x)$ ε-approx.?	if $f(x)$ is closer to $f_{\max}$ or $f_{\min}$?
	black-box $\mathcal{F}$	No	No	No
	neural network (NN) $\mathcal{F}_{\text{NN}}$	No	No	No
	NN trained with ERM $\mathcal{F}_{\text{ERM}}$	No	No	No
	$L-$Lipschitz (known L) $\mathcal{F}_{\text{Lip}}^{L}$	No	No	Only for small L
	piecewise affine $\mathcal{F}_{\text{PAff}}$	No	No	No
	monotonic $\mathcal{F}_{\text{Mono}}$	No	No	No
	convex $\mathcal{F}_{\text{Convx}}$	No	No	No
	affine (input dim. > 1) $\mathcal{F}_{\text{Aff}}^{d>1}$	No	No	No
	affine (input dim. = 1) $\mathcal{F}_{\text{Aff}}^{d=1}$	Yes	Yes	Yes
	constant $\mathcal{F}_{\text{Const}}$	Yes	Yes	Yes

$\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ *and*

$$\left\| f' - D(\Phi_{\min\max}(f')) \right\|_\infty \geq \frac{f'_{\max} - f'_{\min}}{2}. \quad (4)$$

Consider a user who knows that the unit to visualize is piecewise affine ($f \in \mathcal{F}_{\text{PAff}}$). Using this knowledge, they hope to predict f by applying some decoder to the visualization $\Phi_{\min\max}(f)$. However, for every f, there is always another f' that satisfies the user's knowledge ($f' \in \mathcal{F}_{\text{PAff}}$) and has $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$. Therefore, without further information it is impossible to distinguish between the case when the true function is f and when it is f', regardless of how refined the decoder is. Theorem 1 says that f and f' are sufficiently different, and thus, the user will do poorly at predicting at least one of them; that is, the user does not improve on the uninformative predictive ability prescribed by Proposition 1. This implies No for the first two columns in Table 1. A similar result can be shown for $\mathcal{F}_{\text{Convx}}$ and $\mathcal{F}_{\text{Lip}}^{L}$ as shown in Theorem 3 and Theorem 4, accordingly.

Our second result is an analogous negative result for predicting whether $f(x)$ is closer to $f_{\max}$ or $f_{\min}$, implying No for the third column in Table 1. To state it, for any $f \in \mathcal{F}$ we define $m_f = (f_{\max} + f_{\min})/2$; note that $f(x)$ is closer to $f_{\max}$ iff $f(x) > m_f$.

Theorem 2. *For all $\mathcal{G} \in \{\mathcal{F}, \mathcal{F}_{\text{NN}}, \mathcal{F}_{\text{ERM}}, \mathcal{F}_{\text{PAff}}, \mathcal{F}_{\text{Mono}}, \mathcal{F}_{\text{Convx}}\}$, $D \in \mathcal{D}$, and $f \in \mathcal{G}$, there exists $f' \in \mathcal{G}$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ and*

$$\left\| \mathbf{1}_{f'>m_{f'}} - \mathbf{1}_{D(\Phi_{\min\max}(f'))>m_{f'}} \right\|_\infty \geq \mathbf{1}_{f_{\max} \neq f_{\min}}. \quad (5)$$

The LHS of Eq. (5) quantifies "Can the user tell if $f'(x)$ is closer to $f'_{\max}$ or $f'_{\min}$?" Since indicator functions are bounded in $[0, 1]$, the LHS is trivially bounded above by 1. Again consider the user who knows $f \in \mathcal{F}_{\text{PAff}}$. Theorem 2 says that for any f—unless f also happens to be constant (i.e., $f_{\max} = f_{\min}$)—there is always some $f' \in \mathcal{F}_{\text{PAff}}$ that is indistinguishable from f to the user *and* sufficiently different from f so that the user cannot reliably tell if $f'(x)$ is closer to $f'_{\max}$ or $f'_{\min}$ (i.e., the RHS is also 1). The same result can be shown for $\mathcal{F}_{\text{Lip}}^{L}$ with dependence on L (Theorem 5).

For an analogous analysis and further results for more function classes, see Theorems 6 to 8. In summary, we prove that **without additional assumptions about a function, it is impossible to guarantee that standard feature visualizations can be used to understand (i.e., predict) many types of functions, including black boxes, neural networks, and even convex functions**. That said, if strong additional knowledge is available, for instance, if the function is known to be affine with low input dimensionality, then feature visualizations are provably reliable. In line with other work (Srinivas & Fleuret, 2019; Bilodeau et al., 2024; Fokkema et al., 2022; Han et al., 2022), this marks a departure from the conventional concept of black-box interpretability and suggests that more knowledge about the function—for instance, enforced through architectural primitives—are necessary to ensure reliability.

5. Conclusion

Feature visualizations based on activation maximization are a widely used tool within the mechanistic interpretability community. We here asked whether feature visualizations

are reliable, i.e., whether we can trust/rely on their results. Our work has the following practical implications:

1. **Adversarial perspective: Feature visualizations can be arbitrarily manipulated/fooled.** Thus, contrary to calls in the literature, feature visualizations are not a reliable tool for model auditing by a third party that did not train the model itself (e.g., a regulator).
2. **Empirical perspective: A sanity check shows that feature visualizations are processed very differently from natural images; they currently do not explain how natural input is processed.** We therefore recommend: (a) using feature visualization for *exploratory* but not for *confirmatory* use cases; (b) when proposing a new feature visualization method, running the quantitative sanity check we introduced to measure whether the method reflects how natural input is processed throughout the network; (c) consistent with the recommendation by Olah et al. (2018; 2020), always combining visualizations with additional methods including dataset samples. That said, even a combination of feature visualizations with natural samples may not be reliable, since natural samples as an interpretability method can be manipulated, too (Nanfack et al., 2024).
3. **Theoretical perspective: Feature visualization based on activation maximization can only be guaranteed to be reliable if we know a lot about the network already; it's impossible if the network is a black-box.** This challenges the concept of post-hoc interpretability methods: explaining completely black-box systems may sometimes be more than we can hope for. We would love to see future work with theoretical guarantees.

We believe that developing novel, more reliable feature visualizations is a challenging and important direction for future work. Given that our theory proves that visualizing black-box systems via feature visualizations based on activation maximization (i.e., the current dominant paradigm) cannot be guaranteed to be reliable without making strong assumptions about the system, we see two potential avenues: either radically deviating from activation maximization, or making much stronger assumptions on the network (e.g., stronger linearity as explored in Appendix D). In either case, it is important to understand the problem first before we can solve it—in fact, developing solutions may well be a multi-year effort. This article aims to convince readers that there is indeed an important problem, proposes a sanity check, develops a theoretical framework for reliability guarantees, and seeks to motivate future work on solutions.

Reproducibility statement

Code to replicate experiments from this paper is available here: https://github.com/google-research/fooling-feature-visualizations/

The proofs for our theory can be found in Appendix B. Method details beyond the descriptions from the main text, including the choice of hyperparameters, are available from our extensive appendix as well (Appendix C). There are no special compute requirements (e.g., we do not train large models). Information pertaining to code libraries and feature visualization details can be found in Appendix C.2. We added a table of contents at the beginning of the appendix section to facilitate accessibility.

Impact Statement

Our paper investigates the reliability of feature visualizations. Overall, we expect this to contribute to better scrutiny towards existing interpretability methods, which hopefully inspires the development of more reliable interpretability methods in the future, as well the development of models that incorporate certain reliably "interpretability-enabling" assumptions right from the start, rather than being faced with the (sometimes impossible) task of post-hoc interpretability through feature visualizations.

In terms of potential negative impact, the fooling methods developed here could be used to deceive an entity (e.g., a model auditor or regulator) as described in Section 2 and Appendix A.1. That being said, we believe that the risk is lower if this knowledge is public—it would be much more problematic to believe that feature visualizations can be taken at face value, because then whoever designs a fooling circuit would be met with an unsuspecting audience.

Acknowledgments

We would like to thank (in alphabetic order): Matthias Bethge, Judy Borowski, Thomas Klein, Pang Wei Koh, David Fleet, Ari Morcos, Chris Olah, Lisa Schut, Caroline Seidel, Paul Vicol, Felix Wichmann and our anonymous reviewers for helpful discussions and feedback. All opinions expressed in this article are our own and are not necessarily shared by any of the colleagues we thank here. This work was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A, 01IS18039B. BB acknowledges support from the Vector Institute. WB acknowledges financial support via an Emmy Noether Grant funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1 and via the Open Philantropy Foundation funded by the Good Ventures Foundation. WB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting RSZ.

Author contributions

The project was led and coordinated by RG. WB developed the core idea that the arg max does not constrain a function sufficiently, which can be exploited in order to manipulate feature visualizations (key insight behind Section 2 and Section 4). RG had the idea for Section 2.1; RG and RSZ conducted the experiments. RSZ and WB had the idea for Section 2.2; RSZ conducted the experiments. RG had the idea and ran the analysis for Section 3 based on discussions with BK. RG conceived of Lemma 1, and BB proved it with input from RG and RSZ. BB conceived of Lemma 2, and BB proved

it with input from RG and RSZ. BB and RG conceived of the main results in Section 4. BB formalized and proved the results in Section 4 and the corresponding appendix. WB had the idea; RSZ ran the analysis for Appendix D based on discussions with WB and RG. BK, and WB at a later stage, provided overall guidance throughout the course of the project, helping with presentation and experiment details. The first draft was written by RG apart from Section 2.2 (RSZ), Section 4 (BB; intro jointly with RG) and Appendix D (RSZ and RG). BB curated the final presentation of theoretical results and plain-language descriptions with input from RG, RSZ and BK. All authors contributed to the final writing.

References

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. Sanity checks for saliency maps. *NeurIPS*, 31, 2018. [Cited on pages 1 and 5.]

Adi, Y., Baum, C., Cisse, M., Pinkas, B., and Keshet, J. Turning your weakness into a strength: Watermarking deep neural networks by backdooring. In *27th USENIX Security Symposium (USENIX Security 18)*, pp. 1615–1631, 2018. [Cited on page 15.]

Aïvodji, U., Arai, H., Fortineau, O., Gambs, S., Hara, S., and Tapp, A. Fairwashing: the risk of rationalization. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 161–170. PMLR, 2019. [Cited on page 15.]

Anders, C. J., Pasliev, P., Dombrowski, A., Müller, K., and Kessel, P. Fairwashing explanations with off-manifold detergent. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pp. 314–323. PMLR, 2020. [Cited on page 15.]

Arora, R., Basu, A., Mianjy, P., and Mukherjee, A. Understanding deep neural networks with rectified linear units. In *ICLR (Poster)*. OpenReview.net, 2018. [Cited on page 22.]

Banerjee, A., Guo, X., and Wang, H. On the optimality of conditional expectation as a Bregman predictor. *IEEE Transactions on Information Theory*, 51(7):2664–2669, 2005. [Cited on page 19.]

Baniecki, H., Kretowicz, W., and Biecek, P. Fooling partial dependence via data poisoning. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2022, Grenoble, France, September 19–23, 2022, Proceedings, Part III*, pp. 121–136. Springer, 2023. [Cited on page 15.]

Bareeva, D., Höhne, M. M.-C., Warnecke, A., Pirch, L., Müller, K.-R., Rieck, K., and Bykov, K. Manipulating feature visualizations with gradient slingshots. *arXiv preprint*, abs/2401.06122, 2024. URL https://arxiv.org/abs/2401.06122. [Cited on page 15.]

Bashivan, P., Kar, K., and DiCarlo, J. J. Neural population control via deep image synthesis. *Science*, 364(6439): eaav9436, 2019. [Cited on page 1.]

Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6541–6549, 2017. [Cited on page 16.]

Bilodeau, B., Jaques, N., Koh, P. W., and Kim, B. Impossibility theorems for feature attribution. *Proceedings of the National Academy of Sciences*, 121(2):e2304406120, 2024. [Cited on page 8.]

Bitterwolf, J., Müller, M., and Hein, M. In or out? fixing imagenet out-of-distribution detection evaluation. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2471–2506. PMLR, 2023. [Cited on page 33.]

Borowski, J., Zimmermann, R. S., Schepers, J., Geirhos, R., Wallis, T. S. A., Bethge, M., and Brendel, W. Exemplary natural images explain CNN activations better than state-of-the-art feature visualization. In *ICLR*. OpenReview.net, 2021. [Cited on pages 2, 16, 25, and 30.]

Brundage, M., Avin, S., Wang, J., Belfield, H., Krueger, G., Hadfield, G., Khlaaf, H., Yang, J., Toner, H., Fong, R., et al. Toward trustworthy AI development: mechanisms for supporting verifiable claims. *arXiv preprint*, abs/2004.07213, 2020. URL https://arxiv.org/abs/2004.07213. [Cited on pages 2 and 14.]

Chen, K., Garudadri, H., and Rao, B. D. Improved bounds on neural complexity for representing piecewise linear functions. In *NeurIPS*, 2022. [Cited on page 22.]

Chen, X., Liu, C., Li, B., Lu, K., and Song, D. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint*, abs/1712.05526, 2017. URL https://arxiv.org/abs/1712.05526. [Cited on page 15.]

Erhan, D., Bengio, Y., Courville, A., and Vincent, P. Visualizing higher-layer features of a deep network. *University of Montreal*, 1341(3):1, 2009. [Cited on pages 1, 7, and 15.]

Fang, S. and Choromanska, A. Backdoor attacks on the DNN interpretation system. In *AAAI*, pp. 561–570, 2022. [Cited on page 15.]

Fel, T., Boissin, T., Boutin, V., Picard, A., Novello, P., Colin, J., Linsley, D., Rousseau, T., Cadène, R., Gardes, L., et al. Unlocking feature visualization for deeper networks with MAgnitude Constrained Optimization. *arXiv preprint*, abs/2306.06805, 2023. URL https://arxiv.org/abs/2306.06805. [Cited on page 1.]

Fokkema, H., de Heide, R., and van Erven, T. Attribution-based explanations that provide recourse cannot be robust. *arXiv preprint*, abs/2205.15834, 2022. URL https://arxiv.org/abs/2205.15834. [Cited on page 8.]

Future of Life Institute. Pause giant AI experiments: An open letter, 2023. URL https://futureoflife.org/open-letter/pause-giant-ai-experiments/. [Cited on page 1.]

Gale, E. M., Martin, N., Blything, R., Nguyen, A., and Bowers, J. S. Are there any 'object detectors' in the hidden layers of CNNs trained to identify objects or scenes? *Vision Research*, 176:60–71, 2020. [Cited on page 2.]

Goldblum, M., Tsipras, D., Xie, C., Chen, X., Schwarzschild, A., Song, D., Madry, A., Li, B., and Goldstein, T. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45 (2):1563–1580, 2022. [Cited on page 15.]

Greentfrapp. Lucent. https://github.com/greentfrapp/lucent, v0.1.8. [Cited on page 26.]

Gu, T., Dolan-Gavitt, B., and Garg, S. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint*, abs/1708.06733, 2017. URL https://arxiv.org/abs/1708.06733. [Cited on page 15.]

Gupta, A. and Arora, S. A simple saliency method that passes the sanity checks. *arXiv preprint*, abs/1905.12152, 2019. URL https://arxiv.org/abs/1905.12152. [Cited on page 1.]

Han, T., Srinivas, S., and Lakkaraju, H. Which explanation should I choose? A function approximation perspective to characterizing post hoc explanations. In *NeurIPS*, 2022. [Cited on page 8.]

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. [Cited on page 4.]

Hein, M. and Andriushchenko, M. Formal guarantees on the robustness of a classifier against adversarial manipulation. *Advances in neural information processing systems*, 30, 2017. [Cited on page 32.]

Heinrich, K., Zschech, P., Skouti, T., Griebenow, J., and Riechert, S. Demystifying the black box: A classification scheme for interpretation and visualization of deep intelligent systems. In *Twenty-fifth Americas Conference on Information Systems*, 2019. [Cited on page 7.]

Heo, J., Joo, S., and Moon, T. Fooling neural network interpretations via adversarial model manipulation. *NeurIPS*, 32, 2019. [Cited on page 15.]

Hornik, K., Stinchcombe, M., and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989. [Cited on page 17.]

Kriegeskorte, N. Deep neural networks: a new framework for modeling biological vision and brain information processing. *Annual Review of Vision Science*, 1:417–446, 2015. [Cited on page 16.]

Leavitt, M. L. and Morcos, A. Towards falsifiable interpretability research. *arXiv preprint*, abs/2010.12016, 2020. URL https://arxiv.org/abs/2010.12016. [Cited on page 16.]

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30, 2017. [Cited on page 1.]

Mahendran, A. and Vedaldi, A. Understanding deep image representations by inverting them. In *CVPR*, pp. 5188–5196. IEEE Computer Society, 2015. [Cited on page 15.]

Mahendran, A. and Vedaldi, A. Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision*, 120:233–255, 2016. [Cited on page 1.]

Mordvintsev, A., Olah, C., and Tyka, M. DeepDream-a code example for visualizing neural networks. *Google Research*, 2(5), 2015. [Cited on page 1.]

Nanfack, G., Fulleringer, A., Marty, J., Eickenberg, M., and Belilovsky, E. Adversarial attacks on the interpretation of neuron activation maximization. In *AAAI*, volume 38, pp. 4315–4324, 2024. [Cited on pages 9, 15, and 16.]

Nguyen, A., Yosinski, J., and Clune, J. Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *arXiv preprint*, abs/1602.03616, 2016. URL https://arxiv.org/abs/1602.03616. [Cited on page 1.]

Nguyen, A., Yosinski, J., and Clune, J. Understanding neural networks via feature visualization: A survey. *Explainable AI: interpreting, explaining and visualizing deep learning*, pp. 55–76, 2019. [Cited on pages 1, 2, 7, 14, and 16.]

Nie, W., Zhang, Y., and Patel, A. A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. In *ICML*, pp. 3809–3818. PMLR, 2018. [Cited on page 1.]

Noppel, M., Peter, L., and Wressnegger, C. Backdooring explainable machine learning. *arXiv preprint*, abs/2204.09498, 2022. URL https://arxiv.org/abs/2204.09498. [Cited on page 15.]

Olah, C., Mordvintsev, A., and Schubert, L. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization. [Cited on pages 1, 2, 3, 5, 6, 7, 15, 16, and 25.]

Olah, C., Satyanarayan, A., Johnson, I., Carter, S., Schubert, L., Ye, K., and Mordvintsev, A. The building blocks of interpretability. *Distill*, 2018. doi: 10.23915/distill.00010. https://distill.pub/2018/building-blocks. [Cited on pages 7, 9, and 15.]

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., and Carter, S. Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001, 2020. [Cited on pages 3, 9, and 16.]

Ponce, C. R., Xiao, W., Schade, P. F., Hartmann, T. S., Kreiman, G., and Livingstone, M. S. Evolving images for visual neurons using a deep generative network reveals coding principles and neuronal preferences. *Cell*, 177(4): 999–1009, 2019. [Cited on page 1.]

Pulvermüller, F., Garagnani, M., and Wennekers, T. Thinking in circuits: toward neurobiological explanation in cognitive neuroscience. *Biological Cybernetics*, 108:573–593, 2014. [Cited on page 3.]

Rao, S., Böhle, M., and Schiele, B. Towards better understanding attribution methods. In *CVPR*, pp. 10223–10232, 2022. [Cited on page 1.]

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do ImageNet classifiers generalize to ImageNet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019. [Cited on page 33.]

Ribeiro, M. T., Singh, S., and Guestrin, C. "why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144, 2016. [Cited on page 1.]

Sabour, S., Cao, Y., Faghri, F., and Fleet, D. J. Adversarial manipulation of deep representations. In *International Conference on Learning Representations*, 2015. [Cited on page 15.]

Salman, H., Ilyas, A., Engstrom, L., Kapoor, A., and Madry, A. Do adversarially robust imagenet models transfer better? *NeurIPS*, 33:3533–3545, 2020. [Cited on page 32.]

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 618–626, 2017. [Cited on page 1.]

Shamsabadi, A. S., Yaghini, M., Dullerud, N., Wyllie, S. C., Aïvodji, U., Alaagib, A., Gambs, S., and Papernot, N. Washing the unwashable : On the (im)possibility of fairwashing detection. In *NeurIPS*, 2022. [Cited on page 15.]

Sharkey, L. Circumventing interpretability: How to defeat mind-readers. *arXiv preprint*, abs/2212.11415, 2022. URL https://arxiv.org/abs/2212.11415. [Cited on page 14.]

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. Fooling LIME and SHAP: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pp. 180–186, 2020. [Cited on page 15.]

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*, 2014. [Cited on page 1.]

Srinivas, S. and Fleuret, F. Full-gradient representation for neural network visualization. In *NeurIPS*, pp. 4126–4135, 2019. [Cited on page 8.]

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *ICML*, pp. 3319–3328. PMLR, 2017. [Cited on page 1.]

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. Going deeper with convolutions. In *CVPR*, pp. 1–9. IEEE Computer Society, 2015. [Cited on page 3.]

Tramer, F., Carlini, N., Brendel, W., and Madry, A. On adaptive attacks to adversarial example defenses. *NeurIPS*, 33:1633–1645, 2020. [Cited on page 37.]

Viering, T., Wang, Z., Loog, M., and Eisemann, E. How to manipulate CNNs to make them lie: the GradCAM case. *arXiv preprint*, abs/1907.10901, 2019. URL https://arxiv.org/abs/1907.10901. [Cited on page 15.]

Walker, E. Y., Sinz, F. H., Cobos, E., Muhammad, T., Froudarakis, E., Fahey, P. G., Ecker, A. S., Reimer, J., Pitkow, X., and Tolias, A. S. Inception loops discover what excites neurons most using deep predictive models. *Nature Neuroscience*, 22(12):2060–2065, 2019. [Cited on page 1.]

Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., and Lipson, H. Understanding neural networks through deep visualization. *arXiv preprint*, abs/1506.06579, 2015. URL https://arxiv.org/abs/1506.06579. [Cited on page 1.]

Zimmermann, R. S., Klindt, D. A., and Brendel, W. Measuring mechanistic interpretability at scale without humans. In *ICLR 2024 Workshop on Representational Alignment*. [Cited on page 16.]

Zimmermann, R. S., Borowski, J., Geirhos, R., Bethge, M., Wallis, T. S. A., and Brendel, W. How well do feature visualizations support causal understanding of CNN activations? In *NeurIPS*, pp. 11730–11744, 2021. [Cited on pages 2, 16, 25, 30, and 31.]

Zimmermann, R. S., Klein, T., and Brendel, W. Scale alone does not improve mechanistic interpretability in vision models. *NeurIPS*, 36, 2024. [Cited on pages 2, 30, and 31.]

Appendix

Table of Contents

A Literature **14**
A.1 Related work on deceiving interpretability methods . . . 14
A.2 Literature expectations about feature visualization . . . 15
A.3 Relationship to highly activating natural samples . . . 16

B Proofs and theory details **17**
B.1 Proof for fooling circuit (Section 2.1) . . . 17
B.2 Details on interpretation of Table 1 . . . 18
B.3 Additional impossibility results . . . 19
B.4 Positive results . . . 20
B.5 Proofs for Section 4 . . . 20
B.6 Can we move beyond worst-case analyses? . . . 25
B.7 Relationship of theoretical results to psychophysical experiments . . . 25

C Method details **25**
C.1 Classifier training (Section 2.1) . . . 25
C.2 Feature visualization figures (Section 2) . . . 26
C.3 Sanity check (Section 3) . . . 26

D Are more linear units easier to interpret? **30**

E Do fooling methods change model behavior on OOD input? **32**

F Fooling feature visualizations of adversarially robust models **32**

G Do feature visualizations fail the sanity check simply because they are out-of-distribution? **33**

H Silent unit manipulation: sensitivity studies **35**

I What happens when a slightly different method for visualization is used which is considered out of distribution by the classifier? **36**

J Limitations **36**

K Image sources **37**

A. Literature

A.1. Related work on deceiving interpretability methods

Our experiments from Section 2 serve two purposes. First, we provide a *proof of concept* that it is possible to develop networks with arbitrary or misleading visualizations. Second, feature visualizations have been proposed as model auditing tools (Brundage et al., 2020) that should be integrated "into the testbeds for AI applications" (Nguyen et al., 2019, p. 20). Our work demonstrates the first "interpretability circumvention method" (term by Sharkey, 2022) for feature visualization, which corresponds to a well-known *attack scenario* where an entity wants to hide certain network behavior (e.g., to fool

a third-party model audit or regulator). For instance, the literature considers scenarios where a model bias is discovered (e.g., a model exploits protected attributes like gender for classification), but since removing this bias decreases model performance, there is an incentive to hide the bias instead (Heo et al., 2019; Anders et al., 2020; Shamsabadi et al., 2022) without compromising model performance. Adapting models to maintain their behavior on standard input while showing malicious behavior under adversarial circumstances is known under various names: *fairwashing* if the goal is to hide model bias (Anders et al., 2020; Aïvodji et al., 2019), *model backdooring* or *weight poisoning* (Chen et al., 2017; Gu et al., 2017; Adi et al., 2018) (applied to saliency maps by (Fang & Choromanska, 2022; Noppel et al., 2022)), *data poisoning* (Goldblum et al., 2022) if the change in model weights is achieved through interfering with the training data (explored by Baniecki et al. (2023) in the context of explanation methods), *model manipulation* to fool GradCAM (Viering et al., 2019), *adversarial model manipulation* to fool saliency maps (Heo et al., 2019), and *scaffolding* for fooling LIME and SHAP (Slack et al., 2020). Thus, while we are the first to successfully deceive feature visualizations in this manner, the scenario of adapting a model to deceive an interpretability method has a rich history. A complementary approach is proposed by Nanfack et al. (2024), which changes highly activating dataset samples without changing feature visualizations. Sabour et al. (2015) demonstrated that hidden representations of neural networks can be adversarially manipulated, fooling the early visualization method by Mahendran & Vedaldi (2015). Finally, Bareeva et al. (2024) is a highly related work that provides a way to manipulate feature visualizations through fine-tuning; a finding that is in line with our theory and related to our experiments in Section 2. While the paper was published (on arXiv) only recently and later than this paper, we encourage readers to take a look since it provides a really nice complementary perspective.

A.2. Literature expectations about feature visualization

This short section provides a few expectations/hopes that are presented in the literature when it comes to feature visualizations.

Original activation maximization paper by Erhan et al. (2009):

- "a pattern to which the unit is responding maximally could be a good first-order representation of what a unit is doing"
- "It is perhaps unrealistic to expect that as we scale the datasets to larger and larger images, one could still find a simple representation of a higher layer unit."
- "we hope that such visualization techniques can help understand the nature of the functions learned by the network"

More recent literature:

- "Feature visualization allows us to see how GoogLeNet, trained on the ImageNet dataset, builds up its understanding of images over many layers" (Olah et al., 2017)
- "Feature visualization answers questions about what a network—or parts of a network—are looking for by generating examples." (Olah et al., 2017)
- "If we want to find out what kind of input would cause a certain behavior—whether that's an internal neuron firing or the final output behavior—we can use derivatives to iteratively tweak the input towards that goal" (Olah et al., 2017)
- "optimization approach can be a powerful way to understand what a model is really looking for, because it separates the things causing behavior from things that merely correlate with the causes". "Optimization isolates the causes of behavior from mere correlations." (Olah et al., 2017)
- "In the quest to make neural networks interpretable, feature visualization stands out as one of the most promising and developed research directions. By itself, feature visualization will never give a completely satisfactory understanding. We see it as one of the fundamental building blocks that, combined with additional tools, will empower humans to understand these systems." (Olah et al., 2017)
- "To make a semantic dictionary, we pair every neuron activation with a visualization of that neuron and sort them by the magnitude of the activation."; "Semantic dictionaries give us a fine-grained look at an activation: what does each single neuron detect?" (Olah et al., 2018)
- "Feature visualization helps us answer what the network detects" (Olah et al., 2018)

- "The behavior of a CNN can be visualized by sampling image patches that maximize activation of hidden units [...], or by using variants of backpropagation to identify or generate salient image features" (Bau et al., 2017)

- "Activation maximization techniques enable us to shine light into the black-box neural networks." (Nguyen et al., 2019)

Critical voices:

- "While these methods may be useful for building intuition, they can also encourage three potentially misleading assumptions: that the visualization is representative of the neuron's behavior; that the neuron is responsible for a clearly delineated portion of the task or the network's behavior; and that the neuron's behavior is representative of the network's behavior." (Leavitt & Morcos, 2020)

- "synthetic images from a popular feature visualization method are significantly less informative for assessing CNN activations than natural images" (Borowski et al., 2021)

- "[We] find no evidence that a widely-used feature visualization method provides humans with better 'causal understanding' of unit activations than simple alternative visualizations" (Zimmermann et al., 2021)

- "Neural networks often contain 'polysemantic neurons' that respond to multiple unrelated inputs." (Olah et al., 2020)

- "Units similar to those [hand-picked units] may be the exception rather than the rule, and it is unclear whether they are essential to the functionality of the network. For example, meaningful selectivities could reside in linear combinations of units rather than in single units, with weak distributed activities encoding essential information." (Kriegeskorte, 2015)

A.3. Relationship to highly activating natural samples

In the interpretability community, visualizing highly activating natural samples for certain units is often done either alongside or instead of feature visualizations (Olah et al., 2017; Borowski et al., 2021; Zimmermann et al., 2021; Zimmermann et al.). A natural question to ask is whether our results would apply to highly activating natural samples, too. Since this paper covers three perspectives we have three answers to this question:

From the *adversarial perspective*, we could easily use our method from Section 2.2 to build a network where the top k natural images do not correspond to what the unit is usually selective for. This can be achieved by setting the bias parameter b to a smaller value such that it would only suppress activations up to the, say, 95th percentile of natural input. A recent paper specifically looked into manipulating the top-k activating images (Nanfack et al., 2024).

From the *empirical perspective*, highly activating natural images would pass the sanity check since highly activating natural images are, by definition, natural images that highly activate a unit and they would thus be processed like other natural images.

From the *theoretical perspective*, our impossibility results can be extended to many variations on using the argmax for feature visualization, including using the most activating dataset samples as explanations. Essentially, as long as the feature visualization method does not narrow down the function space too much, our results will apply. It is easy to see that two simple (e.g., piecewise linear) functions could have the same 5 (or 10, etc.) local (arg)maxima and yet behave very differently even quite near these local maxima, and hence our theorems could be extended.

Thus in summary, highly activating natural images would pass our sanity check but it would still be possible to construct networks that show misleading highly activating natural images, which is a case that is covered by the theory.

B. Proofs and theory details

B.1. Proof for fooling circuit (Section 2.1)

Lemma 1. *Let $k > 0$, $A : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}$ and $B, C : \mathbb{R}_+ \times \{0, 1\} \to \mathbb{R}_+$ with*

$$\begin{aligned} A(x, y) &= x + y \\ B(x, z) &= \max(0, x - kz) \\ C(x, z) &= \max(0, x + kz - k) \end{aligned}$$

be computations represented by a sub-graph of a neural network. Denote the combination of these computations as $N : \mathbb{R}_+ \times \mathbb{R}_+ \times \{0, 1\}$ with

$$N(x, y, z) = A(B(x, z), C(y, z)).$$

Then it holds that

$$\forall x, y \in \mathbb{R}_+ : k \geq \max(x, y) \implies \begin{cases} N(x, y, 0) = x \\ N(x, y, 1) = y \end{cases}.$$

Proof of Lemma 1. First, consider the case where the binary input of N is 0; that is, $z = 0$. Then, since $k \geq y$,

$$\begin{aligned} B(x, 0) &= \max(0, x) = x \\ C(y, 0) &= \max(0, y - k) = 0, \end{aligned} \tag{6}$$

so $N(x, y, 0) = A(B(x, 0), C(y, 0)) = A(x, 0) = x$.

Analogously consider $z = 1$. Then, since $k \geq x$,

$$\begin{aligned} B(x, 1) &= \max(0, x - k) = 0 \\ C(y, 1) &= \max(0, y) = y, \end{aligned} \tag{7}$$

so $N(x, y, 1) = A(B(x, 1), C(y, 1)) = A(0, y) = y$, completing the proof. □

Lemma 2. *Let $\mathcal{X}$ denote the space of all possible inputs (e.g., all images), $\mathcal{D}$ some distribution on $\mathcal{X}$ (e.g., ImageNet). Let $D, F : \mathcal{X} \to \mathbb{R}_+$ represent the full computation of a unit in the original and in the tinkered network, respectively, that can be bounded on their domain. For an arbitrary algorithm $\mathrm{Opt} : \{\mathcal{X} \to \mathbb{R}\} \times \mathcal{X} \to \mathcal{X}$ and distribution π_0 on $\mathcal{X}$ define the following sequence of random variables: $\forall n > 0 : X_{n+1} = \mathrm{Opt}(D, X_n)$ and $X_0 \sim \pi_0$. Denote the distribution over $\mathcal{X}$ induced by this process π. If $\mathcal{D}$ and π have disjoint support, then there exists a neural network implementing a function $N : \mathcal{X} \to \mathbb{R}$ such that*

$$\mathbb{P}_{\mathbf{x} \sim \pi}[N(\mathbf{x}) = D(\mathbf{x})] = 1 \text{ and } \mathbb{P}_{\mathbf{x} \sim \mathcal{D}}[N(\mathbf{x}) = F(\mathbf{x})] = 1.$$

Proof of Lemma 2. As $\mathcal{D}$ and π have disjoint support this means that there exists a function $E : \mathbb{R} \to \{0, 1\}$ such that

$$\mathbb{P}_{\mathbf{x} \sim \pi}[E(\mathbf{x}) = 0] = 1 \text{ and } \mathbb{P}_{\mathbf{x} \sim \mathcal{D}}[E(\mathbf{x}) = 1] = 1. \tag{8}$$

Let $k = \max\left(\max_{\mathbf{x} \in \mathcal{D}} D(\mathbf{x}), \max_{\mathbf{x} \in \pi} D(\mathbf{x})\right)$, which exists as both D and F are bounded. In line with Lemma 1, we construct N as $N(\mathbf{x}) = A(B(D(\mathbf{x}), E(\mathbf{x})), C(F(\mathbf{x}), \neg E(\mathbf{x})))$. Per the universal approximation theorem (Hornik et al., 1989), there exists a neural network implementing the assumed function E. As all other computations (i.e., A, B, C, D, F) are implemented by a neural network, we can conclude that the constructed function N can also be implemented by a neural network.

Applying Lemma 1 and Eq. (8) directly yields

$$\mathbb{P}_{\mathbf{x} \sim \pi}[N(\mathbf{x}) = D(\mathbf{x})] = 1 \text{ and } \mathbb{P}_{\mathbf{x} \sim \mathcal{D}}[N(\mathbf{x}) = F(\mathbf{x})] = 1, \tag{9}$$

concluding the proof. □

Remark 1. *In the case of feature visualizations, the assumption that π and $\mathcal{D}$ have disjoint support is plausible as demonstrated empirically in Section 2.1; this can also be visually appreciated from looking at Figure 6 showing a visualization trajectory which at no point resembles natural images.* ◁

B.2. Details on interpretation of Table 1

First, we elaborate on what No and Yes mean in Table 1.

The weakest form of answering No would be to find a *single* function f where feature visualization cannot be used to predict f. At the other extreme, one could hope to show that feature visualization cannot be used to predict f *for all* f. Unfortunately, this is trivially impossible to show: for every distinct value the feature visualization can take, one could pick a function f that agrees with this visualization (e.g., has the same $\arg\max$) and use this as the prediction. In light of this, we prove the next strongest impossibility result. When the answer is No, we show that *for all*[2] functions f (except for a handful of corner cases, like constant functions), there exists another function f' that gets the exact same feature visualization as f yet cannot be accurately predicted. Similarly, for the cells with **extra assumptions in orange**, this means that the answer is No (as defined in the previous sentence) *unless these extra assumptions are satisfied*.

We measure predictive accuracy using the sup norm for simplicity, but our results could be extended to any other strictly convex loss. This is essentially the strongest result one could hope for: by the intermediate value theorem, any continuous f must take every value in between $f_{\min}$ and $f_{\max}$, and hence it is impossible to prove that $f(x)$ can't be recovered *for every* x. On the contrary, for the cells where the answer is Yes, we can actually prove something much stronger than the converse of No: we prove that $f(x)$ can be predicted *for all* x and *for all* f. This hints at the necessity of such strong assumptions. Either the function class is so simple that feature visualization reveals everything about every function, or feature visualization reveals hardly anything about any function.

To find the precise results that correspond to each cell of Table 1, see Table 2.

Table 2: Theoretical results corresponding to each cell of Table 1.

Stronger assumptions about f ↓	Given feature visualization for a function f and an input x, can we reliably predict…		$f(x)$?	$f(x)$ ε-approx.?	if $f(x)$ is closer to $f_{\max}$ or $f_{\min}$?
	black-box	$\mathcal{F}$	Theorem 1	Theorem 1	Theorem 2
	neural network (NN)	$\mathcal{F}_{\text{NN}}$	Theorem 1	Theorem 1	Theorem 2
	NN trained with ERM	$\mathcal{F}_{\text{ERM}}$	Theorem 1	Theorem 1	Theorem 2
	$L-$Lipschitz (known L)	$\mathcal{F}_{\text{Lip}}^{L}$	Theorem 4	Theorem 4	Theorem 5
	piecewise affine	$\mathcal{F}_{\text{PAff}}$	Theorem 1	Theorem 1	Theorem 2
	monotonic	$\mathcal{F}_{\text{Mono}}$	Theorem 1	Theorem 1	Theorem 2
	convex	$\mathcal{F}_{\text{Convx}}$	Theorem 3	Theorem 3	Theorem 2
	affine (input dim. > 1)	$\mathcal{F}_{\text{Aff}}^{d>1}$	Theorem 6	Theorem 6	Theorem 7
	affine (input dim. = 1)	$\mathcal{F}_{\text{Aff}}^{d=1}$	Theorem 8	Theorem 8	Theorem 8
	constant	$\mathcal{F}_{\text{Const}}$	Theorem 8	Theorem 8	Theorem 8

Finally, we define precisely what each assumption means in Table 1. For any space $\mathcal{A}$, let $\mathcal{M}(\mathcal{A})$ denote the set of all

[2] Affine functions are the only exception, since there are more cases where an affine f can be exactly recovered from feature visualization. See Theorem 6 for the precise characterization.

probability measures on $\mathcal{A}$.

Neural Network: $\mathcal{F}_{\text{NN}} = \Big\{ f \in \mathcal{F} : \text{can be written as mat. mul. with scalar activations} \Big\}$

NN with ERM: $\mathcal{F}_{\text{ERM}} = \Big\{ f \in \mathcal{F}_{\text{NN}} : \exists \pi \in \mathcal{M}(\mathcal{I} \times [0,1]) \text{ s.t. } f = \arg\min_{f' \in \mathcal{F}} \mathop{\mathbb{E}}_{(X,Y)\sim\pi} \ell(f'(X), Y), \text{ where } \ell \text{ is a Bregman loss function (see Banerjee et al., 2005)} \Big\}$

Lipschitz: $\mathcal{F}^L_{\text{Lip}} = \Big\{ f \in \mathcal{F} : \sup_{x,x' \in \mathcal{I}} \frac{f(x) - f(x')}{\|x - x'\|_\infty} \leq L \Big\}$

Piecewise Affine: $\mathcal{F}_{\text{PAff}} = \Big\{ f \in \mathcal{F} : \text{can be written as affine on each piece of a partition of } \mathcal{I} \Big\}$

Monotone: $\mathcal{F}_{\text{Mono}} = \Big\{ f \in \mathcal{F} : \forall x \leq x', f(x) \leq f(x') \Big\} \cup \Big\{ f : \forall x \leq x', f(x) \geq f(x') \Big\}$ [3]

Convex: $\mathcal{F}_{\text{Convx}} = \Big\{ f \in \mathcal{F} : \forall x, x' \in \mathcal{I}\ \forall \alpha \in [0,1], f(\alpha x + (1-\alpha)x') \leq \alpha f(x) + (1-\alpha) f(x') \Big\}$

Affine: $\mathcal{F}^d_{\text{Aff}} = \Big\{ f \in \mathcal{F} : \exists A \in \mathbb{R}^d\ \exists b \in \mathbb{R} \text{ s.t. } \forall x \in \mathcal{I}, f(x) = A^{\mathrm{T}} x + b \Big\}$

Constant: $\mathcal{F}_{\text{Const}} = \Big\{ f \in \mathcal{F} : \forall x, x' \in \mathcal{I}, f(x) = f(x') \Big\}.$

B.3. Additional impossibility results

While Theorems 1 and 2 already provide impossibility for strong assumptions like monotonicity and piecewise affine, convexity is a particularly strong assumption since it restricts the output space of functions. In particular, no convex function can cross the diagonal line from $f_{\min}$ to $f_{\max}$, and hence it is possible that more information can be recovered from just these values. However, the next result shows that this information can only be used to possibly improve a constant $1/2$ to $1/4$, and arbitrary approximation is still impossible (unless the function is constant).

Theorem 3. *For all $D \in \mathcal{D}$ and $f \in \mathcal{F}_{\text{Convx}}$, there exists $f' \in \mathcal{F}_{\text{Convx}}$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ and*

$$\Big\| f' - D(\Phi_{\min\max}(f')) \Big\|_\infty \geq \frac{f'_{\max} - f'_{\min}}{4}.$$

Similarly, a known Lipschitz constant implies local stability of f, which may be possible for a decoder to exploit. However, we show that this is also not possible in general (our result is stated in 1-dimension for simplicity, but could be extended trivially to arbitrary dimension using the sup norm definition of Lipschitz).

Theorem 4. *For all $D \in \mathcal{D}$, $L > 0$, and $f \in \mathcal{F}^L_{\text{Lip}}$, there exists $f' \in \mathcal{F}^L_{\text{Lip}}$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ and if $2\,|f_{\max} - f_{\min}| \leq L\,|x_{\max} - x_{\min}|$ then*

$$\Big\| f' - D(\Phi_{\min\max}(f')) \Big\|_\infty \geq \frac{f'_{\max} - f'_{\min}}{2}.$$

Moreover, even if $2\,|f_{\max} - f_{\min}| > L\,|x_{\max} - x_{\min}|$,

$$\Big\| f' - D(\Phi_{\min\max}(f')) \Big\|_\infty \geq L \max\Big\{ \min\{x_{\min}, x_{\max}\}, 1 - \max\{x_{\min}, x_{\max}\} \Big\}.$$

First, for all $f \in \mathcal{F}^L_{\text{Lip}}$ it holds that $|f_{\max} - f_{\min}| \leq L\,|x_{\max} - x_{\min}|$, so the first condition nearly captures all cases. As already argued, under this condition our lower bound is tight by Proposition 1. Even when the condition fails, our lower

[3] For d-dimensional inputs, $x \leq x'$ if and only if $x_j \leq x'_j$ for all $j \in [d]$.

bound is zero if and only if $|x_{\max} - x_{\min}| = 1$ and $|f_{\max} - f_{\min}| > L/2$. This is nearly tight, since if $|f_{\max} - f_{\min}| = L$, then necessarily $|x_{\max} - x_{\min}| = 1$ and f is linear and uniquely identifiable from $\Phi_{\min\max}(f)$ (and hence the lower bound must be zero in this case).

A similar condition can be used to provide a Lipschitz analogue of Theorem 2.

Theorem 5. *For all $D \in \mathcal{D}$, $L > 0$, and $f \in \mathcal{F}_{\text{Lip}}^L$ such that $2\,|f_{\max} - f_{\min}| \leq L\,|x_{\max} - x_{\min}|$, there exists $f' \in \mathcal{F}_{\text{Lip}}^L$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ and*

$$\left\| \mathbf{1}_{f' > m_{f'}} - \mathbf{1}_{D(\Phi_{\min\max}(f')) > m_{f'}} \right\|_\infty \geq \mathbf{1}_{\sup_{x \in \mathcal{I}} f(x) \neq \inf_{x \in \mathcal{I}} f(x)}.$$

Finally, we have the following negative result for affine functions. Due to the extra structure imposed by an affine assumption, in more cases it is possible to fully recover f from just the feature visualization. However, in the worst case, f may still be completely unrecoverable. We show this for $d = 2$; a similar result can be shown in higher dimensions with more careful accounting of edge cases.

Theorem 6. *For all $D \in \mathcal{D}$ and $f \in \mathcal{F}_{\text{Aff}}^{d=2}$, there exists $f' \in \mathcal{F}_{\text{Aff}}^{d=2}$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ and*

$$\left\| f' - D(\Phi_{\min\max}(f')) \right\|_\infty \geq \mathbf{1}_{x_{\min,1} \neq x_{\min,2}} \mathbf{1}_{x_{\max,1} \neq x_{\max,2}} \frac{f'_{\max} - f'_{\min}}{2}.$$

The same can also be shown for the analogue of Theorem 2.

Theorem 7. *For all $D \in \mathcal{D}$ and $f \in \mathcal{F}_{\text{Aff}}^{d=2}$, there exists $f' \in \mathcal{F}_{\text{Aff}}^{d=2}$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f')$ and*

$$\left\| \mathbf{1}_{f' > m_{f'}} - \mathbf{1}_{D(\Phi_{\min\max}(f')) > m_{f'}} \right\|_\infty \geq \mathbf{1}_{x_{\min,1} \neq x_{\min,2}} \mathbf{1}_{x_{\max,1} \neq x_{\max,2}}.$$

Remark 2. *Our negative results for affine functions rely on the constrained nature of the inputs. Without such constraints, the task of feature visualization would generally become even more difficult (and in practice, inputs are always bounded). However, specifically for affine functions, on unbounded inputs one could take advantage of the fact that the* $\arg\max$ *will be proportional to the weight vector, and hence could more reliably predict f from $\Phi_{\min\max}(f)$. Similarly, adding regularization when computing $\Phi_{\min\max}$ could lead to reliably predicting f even with bounded inputs.* ◁

B.4. Positive results

Finally, we state our positive result for very simple functions.

Theorem 8. *For all $\mathcal{G} \in \{\mathcal{F}_{\text{Aff}}^{d=1}, \mathcal{F}_{\text{Const}}\}$ there exists $D \in \mathcal{D}$ such that for all $f \in \mathcal{G}$,*

$$\left\| f - D(\Phi_{\min\max}(f)) \right\|_\infty = 0.$$

B.5. Proofs for Section 4

Remark 3. *Throughout, we prove impossibility results for 1-dimensional functions. The extension to multiple dimensions follows from using our constructions componentwise and then applying Lemma 3 or Lemma 4 as appropriate, which hold for any input dimension.* ◁

B.5.1. Helper lemmas

To prove results for the first two columns on Table 1, we use the following lemma to characterize the performance of an arbitrary decoder $D \in \mathcal{D}$.

Lemma 3. *For any $D \in \mathcal{D}$ and $f_1, f_2 \in \mathcal{F}$ such that $\Phi_{\min\max}(f_1) = \Phi_{\min\max}(f_2)$, for some $f \in \{f_1, f_2\}$*

$$\left\| f - D(\Phi_{\min\max}(f)) \right\|_\infty \geq \frac{\|f_1 - f_2\|_\infty}{2}.$$

Proof of Lemma 3. Let $g = D(\Phi_{\min\max}(f_1)) = D(\Phi_{\min\max}(f_2))$ and let x be such that $|f_1(x) - f_2(x)| = \|f_1 - f_2\|_\infty$. Then, since mean is less than max,

$$\frac{1}{2}|f_1(x) - f_2(x)| \leq \frac{1}{2}|f_1(x) - g(x)| + \frac{1}{2}|g(x) - f_2(x)| \leq \max_{f\in\{f_1,f_2\}} |f(x) - g(x)|\,.$$

□

Then, for any $\mathcal{G} \subseteq \mathcal{F}$ of interest and any $f \in \mathcal{G}$, we simply have to find $f_1, f_2 \in \mathcal{G}$ such that $\Phi_{\min\max}(f) = \Phi_{\min\max}(f_1) = \Phi_{\min\max}(f_2)$ and $\|f_1 - f_2\|_\infty$ is appropriately large (where "large" will depend on f).

Similarly, we use the following lemma to prove results for the third column of Table 1.

Lemma 4. *For any $D \in \mathcal{D}$ and $f_1, f_2 \in \mathcal{F}$ such that $\Phi_{\min\max}(f_1) = \Phi_{\min\max}(f_2)$, for some $f \in \{f_1, f_2\}$*

$$\left\|\mathbf{1}_{f>m_f} - \mathbf{1}_{D(\Phi_{\min\max}(f))>m_f}\right\|_\infty \geq \|\mathbf{1}_{f_1>m} - \mathbf{1}_{f_2>m}\|_\infty\,,$$

where $m = m_{f_1} = m_{f_2}$.

Proof of Lemma 4. Let $g = D(\Phi_{\min\max}(f_1)) = D(\Phi_{\min\max}(f_2))$ and let x be such that $\left|\mathbf{1}_{f_1(x)>m} - \mathbf{1}_{f_2(x)>m}\right| = \|\mathbf{1}_{f_1>m} - \mathbf{1}_{f_2>m}\|_\infty$.

If $\|\mathbf{1}_{f_1>m} - \mathbf{1}_{f_2>m}\|_\infty = 0$ the result holds trivially, so suppose that $\|\mathbf{1}_{f_1>m} - \mathbf{1}_{f_2>m}\|_\infty = 1$. That is, $\mathbf{1}_{f_1(x)>m} \neq \mathbf{1}_{f_2(x)>m}$. If $\mathbf{1}_{g(x)>m} = \mathbf{1}_{f_1(x)>m}$, then

$$\left\|\mathbf{1}_{f_2>m} - \mathbf{1}_{D(\Phi_{\min\max}(f_2))>m}\right\|_\infty \geq \left|\mathbf{1}_{f_2(x)>m} - \mathbf{1}_{g(x)>m}\right| = 1.$$

Otherwise, if $\mathbf{1}_{g(x)>m} = \mathbf{1}_{f_2(x)>m}$, then

$$\left\|\mathbf{1}_{f_1>m} - \mathbf{1}_{D(\Phi_{\min\max}(f_1))>m}\right\|_\infty \geq \left|\mathbf{1}_{f_1(x)>m} - \mathbf{1}_{g(x)>m}\right| = 1.$$

That is,

$$\max_{f\in f_1,f_2} \left\|\mathbf{1}_{f>m} - \mathbf{1}_{D(\Phi_{\min\max}(f))>m}\right\|_\infty \geq 1 = \|\mathbf{1}_{f_1>m} - \mathbf{1}_{f_2>m}\|_\infty\,.$$

□

B.5.2. Proof of Proposition 1

Let $D(x_{\min}, x_{\max}, f_{\min}, f_{\max}) \equiv (1/2)(f_{\max} + f_{\min})$ and fix $f \in \mathcal{F}$. For any x,

$$f(x) - \frac{f_{\max} + f_{\min}}{2} \leq f_{\max} - \frac{f_{\max} + f_{\min}}{2} = \frac{f_{\max} - f_{\min}}{2}$$

and

$$\frac{f_{\max} + f_{\min}}{2} - f(x) \leq \frac{f_{\max} + f_{\min}}{2} - f_{\min} = \frac{f_{\max} - f_{\min}}{2}.$$

Thus,

$$\left|f(x) - \frac{f_{\max} + f_{\min}}{2}\right| \leq \frac{f_{\max} - f_{\min}}{2}.$$

Since x was arbitrary, the result holds. □

B.5.3. Proof of Theorem 1

First, suppose $0 \leq x_{\min} < x_{\max} \leq 1$.

Define

$$f_1(x) = \begin{cases} f_{\min} & x \in [0, (x_{\min} + x_{\max})/2] \\ \frac{2(f_{\max} - f_{\min})}{x_{\max} - x_{\min}} x + \frac{2 f_{\min} x_{\max} - f_{\max} x_{\min} - f_{\max} x_{\max}}{x_{\max} - x_{\min}} & x \in [(x_{\min} + x_{\max})/2, x_{\max}] \\ f_{\max} & x \in [x_{\max}, 1] \end{cases}$$

and

$$f_2(x) = \begin{cases} f_{\min} & x \in [0, x_{\min}] \\ \frac{2(f_{\max} - f_{\min})}{x_{\max} - x_{\min}} x + \frac{f_{\min} x_{\max} + f_{\min} x_{\min} - 2 f_{\max} x_{\min}}{x_{\max} - x_{\min}} & x \in [x_{\min}, (x_{\min} + x_{\max})/2] \\ f_{\max} & x \in [(x_{\min} + x_{\max})/2, 1]. \end{cases}$$

Since $\Phi_{\min \max}(f_1) = \Phi_{\min \max}(f_2) = \Phi_{\min \max}(f)$, and f_1 and f_2 are both monotone and piecewise affine, the result follows for $\mathcal{F}_{\text{Mono}}$ and $\mathcal{F}_{\text{PAff}}$ from applying Lemma 3 and observing that $\|f_1 - f_2\|_\infty \geq f_{\max} - f_{\min}$ (this occurs at $(x_{\min} + x_{\max})/2$).

If $0 \leq x_{\max} < x_{\min} \leq 1$, the same argument applies with $1 - f_1$ and $1 - f_2$.

Finally, when $x_{\min} = x_{\max}$, then $f_{\max} - f_{\min} = 0$ so the result holds trivially.

To prove the result for $\mathcal{F}_{\text{NN}}$, note that we imposed no constraints on $f_{\max}$ or $f_{\min}$. Thus, for any $f \in \mathcal{F}_{\text{NN}}$, we can construct f_1 and f_2. We then use that any piecewise affine function can be exactly represented by a sufficiently large neural network (Arora et al., 2018; Chen et al., 2022) to conclude $f_1, f_2 \in \mathcal{F}_{\text{NN}}$.

The same argument applies to prove the result for $\mathcal{F}$, since clearly $f_1, f_2 \in \mathcal{F}$.

Finally, for $\mathcal{F}_{\text{ERM}}$, we must construct appropriate distributions with conditional means f_1 and f_2 respectively (we already noted these are both elements of $\mathcal{F}_{\text{NN}}$). For simplicity, define the joint distribution by $X \sim \text{Unif}(\mathcal{I})$ and $Y|X \sim \text{Ber}(f_j(X))$ for $j \in \{1, 2\}$. □

B.5.4. Proof of Theorem 2

We use f_1 and f_2 from Theorem 1, and recall that $m = (f_{\min} + f_{\max})/2$. Consider when $0 \leq x_{\min} < x_{\max} \leq 1$. Then, at $x = (x_{\min} + x_{\max})/2$, $f_1(x) = f_{\min} < m$ and $f_2(x) = f_{\max} > m$, so $\|\mathbf{1}_{f_1 > m} - \mathbf{1}_{f_2 > m}\|_\infty = 1$. The result then follows by Lemma 4. If $0 \leq x_{\max} < x_{\min} \leq 1$, the same argument applies with $1 - f_1$ and $1 - f_2$.

For $\mathcal{F}_{\text{Convx}}$, we use f_1 and f_2 from the proof of Theorem 3 (Appendix B.5.5). Recall that $m = (f_{\min} + f_{\max})/2$. Consider when $x_{\max} = 1$ and $x_{\min} < 1$. Then, at $x = x_{\min}/4 + 3/4$, $f_1(x) = f_{\min}/4 + 3 f_{\max}/4 > m$ and $f_2(x) = m$, so $\|\mathbf{1}_{f_1 > m} - \mathbf{1}_{f_2 > m}\|_\infty = 1$. The result then follows by Lemma 4. If $x_{\min} = 0$ and $x_{\max} > 0$, the same argument applies using f_1' and f_2' as defined in Appendix B.5.5.

If $x_{\min} = x_{\max}$ then $f_{\min} = f_{\max}$ and hence f is constant, so the result holds trivially. □

B.5.5. Proof of Theorem 3

Note that for any $f \in \mathcal{F}_{\text{Convx}}$, one of $x_{\min}$ or $x_{\max}$ are in $\{0, 1\}$.

First, consider $x_{\max} = 1$ and $x_{\min} < 1$. Define

$$f_1(x) = \begin{cases} f_{\min} & x \in [0, x_{\min}] \\ \frac{f_{\max} - f_{\min}}{1 - x_{\min}} x + \frac{f_{\min} - f_{\max} x_{\min}}{1 - x_{\min}} & x \in [x_{\min}, 1] \end{cases} \tag{10}$$

and

$$f_2(x) = \begin{cases} f_{\min} & x \in [0, (x_{\min}+1)/2] \\ \frac{2(f_{\max}-f_{\min})}{1-x_{\min}} x + \frac{2f_{\min}-f_{\max}x_{\min}-f_{\max}}{1-x_{\min}} & x \in [(x_{\min}+1)/2, 1]. \end{cases} \tag{11}$$

Clearly, $f_1, f_2 \in \mathcal{F}_{\text{Convx}}$ (since they are flat then linear with positive slope) and $\|f_1 - f_2\|_\infty \geq (f_{\max} - f_{\min})/2$ (this occurs at $x = (x_{\min}+1)/2$). Since $x_{\min} = 0$ implies that $x_{\max} = 1$ by convexity, this case also covers $x_{\min} = 0$.

Second, consider $x_{\max} = 0$ and $x_{\min} > 0$. Using Eqs. (10) and (11), define $g_1(x) = f_1(1-x)$ and $g_2(x) = f_2(1-x)$. These also satisfy $g_1, g_2 \in \mathcal{F}_{\text{Convx}}$ (since they are linear with negative slope then flat) and $\|g_1 - g_2\|_\infty \geq (f_{\max} - f_{\min})/2$ (this occurs at $x = x_{\min}/2$). Since $x_{\min} = 1$ implies that $x_{\max} = 0$ by convexity, this case also covers $x_{\min} = 1$.

Finally, if $x_{\min} = x_{\max}$ then $f_{\min} = f_{\max}$ and the result holds trivially. □

B.5.6. PROOF OF THEOREM 4

When $2|f_{\max} - f_{\min}| \leq L|x_{\max} - x_{\min}|$, the proof of Theorem 1 applies since $f_1, f_2 \in \mathcal{F}^L_{\text{Lip}}$.

Otherwise, suppose that $0 \leq x_{\min} < x_{\max} \leq 1$. Define

$$f_1(x) = \begin{cases} f_{\min} & x \in [0, x_{\min}] \\ \frac{f_{\max}-f_{\min}}{x_{\max}-x_{\min}} x + \frac{f_{\min}x_{\max}-f_{\max}x_{\min}}{x_{\max}-x_{\min}} & x \in [x_{\min}, x_{\max}] \\ f_{\max} & x \in [x_{\max}, 1] \end{cases}$$

and

$$f_2(x) = \begin{cases} -L(x - x_{\min}) + f_{\min} & x \in [0, x_{\min}] \\ \frac{f_{\max}-f_{\min}}{x_{\max}-x_{\min}} x + \frac{f_{\min}x_{\max}-f_{\max}x_{\min}}{x_{\max}-x_{\min}} & x \in [x_{\min}, x_{\max}] \\ -L(x - x_{\max}) + f_{\max} & x \in [x_{\max}, 1]. \end{cases}$$

Recall that by definition of $f \in \mathcal{F}^L_{\text{Lip}}$, $|f_{\max} - f_{\min}| \leq L|x_{\max} - x_{\min}|$, so $f_1, f_2 \in \mathcal{F}^L_{\text{Lip}}$.

If $x_{\min} > 1 - x_{\max}$, then $\|f_1 - f_2\|_\infty \geq L x_{\min}$ (which is realized at $x = 0$), and otherwise $\|f_1 - f_2\|_\infty \geq L(1 - x_{\max})$ (which is realized at $x = 1$).

If $0 \leq x_{\max} < x_{\min} \leq 1$, the same argument applies with $1 - f_1$ and $1 - f_2$. □

B.5.7. PROOF OF THEOREM 5

Since $2|f_{\max} - f_{\min}| \leq L|x_{\max} - x_{\min}|$, the proof of Theorem 2 applies because $f_1, f_2 \in \mathcal{F}^L_{\text{Lip}}$. □

B.5.8. PROOF OF THEOREM 6

When $d = 2$, any $f \in \mathcal{F}^{d=2}_{\text{Aff}}$ satisfies $f(x) = ax_1 + bx_2 + c$ for some $a, b, c \in \mathbb{R}$. Given $\Phi_{\min\max}(f) = (x_{\min}, f_{\min}, x_{\max}, f_{\max})$, the compatible $f \in \mathcal{F}^{d=2}_{\text{Aff}}$ are those f_c such that

$$a = \frac{x_{\max,2}f_{\min} - x_{\min,2}f_{\max} + (x_{\min,2} - x_{\max,2})c}{x_{\min,1}x_{\max,2} - x_{\max,1}x_{\min,2}}$$
$$b = \frac{-x_{\max,1}f_{\min} + x_{\min,1}f_{\max} + (x_{\max,1} - x_{\min,1})c}{x_{\min,1}x_{\max,2} - x_{\max,1}x_{\min,2}},$$

where c is a free parameter (with the only constraint that $f_c(x) \in [0, 1]$ for all $x \in \mathcal{I}$).

Since f is affine, $x_{\min}$ and $x_{\max}$ must both occur at one of the four corners of $[0, 1]^2$. Note that a and b above are undefined for some of these combinations, which we now enumerate.

If $x_{\min} = x_{\max}$, then f is constant and can be recovered exactly. If $x_{\min} = (0,0)$, then necessarily $c = f_{\min}$, so f can be recovered exactly. Similarly, if $x_{\max} = (0,0)$ then necessarily $c = f_{\max}$.

Moreover, there are other cases where f can be recovered. If $x_{\min} = (1,1)$ and $x_{\max} \in \{(1,0),(0,1)\}$, then one of a or b do not depend on c and hence c can be directly recovered. The same is true when $x_{\max} = (1,1)$.

There are two possibilities left.

1) $x_{\min} = (0,1)$ and $x_{\max} = (1,0)$:

$$a = f_{\max} - c$$
$$b = f_{\min} - c.$$

Take $c_1 = f_{\min}$ and $c_2 = f_{\max}$ to get

$$f_1(x) = (f_{\max} - f_{\min})x_1 + f_{\min}$$

and

$$f_1(x) = (f_{\min} - f_{\max})x_2 + f_{\max}.$$

These still have $\Phi_{\min\max}(f) = \Phi_{\min\max}(f_1) = \Phi_{\min\max}(f_2)$, but $\|f_1 - f_2\|_\infty \geq f_{\max} - f_{\min}$ (this is realized at $x = (1,1)$).

2) $x_{\min} = (1,0)$ and $x_{\max} = (0,1)$:

$$a = f_{\min} - c$$
$$b = f_{\max} - c.$$

Take $c_1 = f_{\min}$ and $c_2 = f_{\max}$ to get

$$f_1(x) = (f_{\max} - f_{\min})x_2 + f_{\min}$$

and

$$f_1(x) = (f_{\min} - f_{\max})x_1 + f_{\max}.$$

These still have $\Phi_{\min\max}(f) = \Phi_{\min\max}(f_1) = \Phi_{\min\max}(f_2)$, but $\|f_1 - f_2\|_\infty \geq f_{\max} - f_{\min}$ (this is again realized at $x = (1,1)$). The result holds by then applying Lemma 3. □

B.5.9. Proof of Theorem 7

This follows directly from applying Lemma 4 to the functions constructed in the proof of Theorem 6 (Appendix B.5.8). □

B.5.10. Proof of Theorem 8

First suppose that $f \in \mathcal{F}_{\text{Aff}}^{d=1}$. That is, there exists a, b such that $f(x) = ax + b$ for all x. Given $\Phi_{\min\max}(f)$, define

$$a_f = \frac{f_{\max} - f_{\min}}{x_{\max} - x_{\min}}$$

and

$$b_f = \frac{x_{\max} f_{\min} - x_{\min} f_{\max}}{x_{\max} - x_{\min}}.$$

Set $D(\Phi_{\min\max}(f)) = [x \mapsto a_f x + b_f]$. Since there is a unique affine function passing through both $(x_{\min}, f_{\min})$ and $(x_{\max}, f_{\max})$, and $D(\Phi_{\min\max}(f))$ is an affine function passing through both of these, this implies that $D(\Phi_{\min\max}(f)) \equiv f$.

If $f \in \mathcal{F}_{\text{Const}}$, then there exists $y \in [0,1]$ such that $f_{\min} = f_{\max} = y$ and $f(x) = y$ for all x. Define $D(\Phi_{\min\max}(f)) \equiv f_{\min}$, which implies that $D(\Phi_{\min\max}(f)) \equiv f$. □

B.6. Can we move beyond worst-case analyses?

The theoretical results above are of a worst-case nature (for every claim, we construct a counterexample to refute the claim). Of course, another interesting question to ask is: How likely are these counterexamples to appear in reality when we generate the function f by training a neural net using SGD? Here, we briefly discuss our results in the context of this average-case perspective.

On the one hand, a strength of our counterexamples is that they can be realized by very simple functions, and hence we are not "cherry-picking" convoluted functions that SGD will not learn. Similarly, our counterexamples can be easily extended to a family of functions that are rich in the space of all functions in the model class, again in contrast to having only a single, unrealistic function that works as a counterexample. On the other hand, it is very possible that while these counterexamples are easily learned by SGD, they are more or less likely depending on, say, the specific training data. Unfortunately, to answer this question, it seems one would have to have a rather refined understanding of the distribution of models learned from training on realistic data via SGD, which would resolve some very large open problems in learning theory along the way. We pose it as an open problem to extend our approach of proving negative results for feature visualization by incorporating the learning algorithm.

B.7. Relationship of theoretical results to psychophysical experiments

Borowski et al. (2021) and Zimmermann et al. (2021) performed psychophysical experiments to investigate the tness of feature visualizations for human observers. Both papers find that natural highly activating images are more interpretable (as measured by human prediction performance) compared to feature visualizations. A candidate explanation for this behaviour is our analysis in Section 3, showing that for last-layer Inception-V1 feature visualizations, those visualizations are processed along very different paths for most of the network (compared to natural images as a baseline).

The task used by Borowski et al. (2021) is related to the third column of Table 1. They asked participants to predict which one of two natural images is strongly activating for a certain unit based on maximally and minimally activating feature visualizations for that unit. This can be seen as an easier version of the task in Table 1: Borowski et al. (2021) did not use random test samples but instead two curated samples, out of which one has extremely high and one has extremely low activations. They find that that humans are able to do this task above chance.

At first glance, this result seems to contradict our theoretical finding from Theorem 2, which states that reliable prediction is impossible unless additional assumptions about the function are known. However, there is no contradiction: our theory allows for the possibility of a specific function (e.g., a specific neural network unit) and a specific decoder (e.g., a human observer) to be aligned in the sense that predictions about the function can happen to be correct—however, for every function f for which the decoder is correct there is a function f' from the same function family for which the decoder is wrong (in spirit, a case of "no free lunch" for feature visualization). If an observer gets significantly above chance in one case, they would pay the price by being significantly below chance in the other case. To the best of our knowledge, there is currently no way for the observer to know in advance whether they're visualizing a function f for which their decoding is aligned or a function f' for which their decoding leads to the wrong conclusions. It is an interesting open question to develop a practical and rigorous approach to distinguish these cases, perhaps relying on additional information such as the data distribution (e.g., does ImageNet lead to benign f more often?) and the training procedure (e.g., does SGD lead to benign f more often?).

C. Method details

C.1. Classifier training (Section 2.1)

For classifier training, we create a dataset by combining $1,281,167$ images from the training set of the ImageNet 2012 dataset and $472,500$ synthetic images. These synthetic images are the (intermediate) results of the feature visualization optimization process. Specifically, for $1,000$ classification units in the last layer of an ImageNet-trained InceptionV1 network, we run the optimization process with the parameters used by (Olah et al., 2017) 35 times each, resulting in $35,000$ unique optimization trajectories. We logarithmically sample 15 (intermediate) steps from the optimization trajectory, resulting in $525,000$ synthetic images in total. Finally, we split the synthetic images into $472,500$ $(= 90\%)$ training and $52,500$ $(= 10\%)$ testing images. Note that we use different units for the two sets. We train a model implementing the simple six layer CNN architecture displayed in Table 3 for 8 epochs on the aforementioned dataset with an SGD optimizer using a learning rate of 0.01, momentum of 0.9 and weight decay of 0.00005. The classifier achieves a near-perfect accuracy of 99.49% on the held-out test set (99.66% and 99.31% for natural input and feature visualizations, respectively).

Type	Size/Channels	Activation	Stride
Conv 3×3	16	ReLU	3
Conv 5×5	16	ReLU	2
Conv 5×5	16	ReLU	2
Conv 5×5	16	ReLU	2
Conv 5×5	16	ReLU	2
Conv 3×3	16	ReLU	2
Flatten	-	-	-
Linear	1	-	-

Table 3: Architecture of the classifier used to detect feature visualizations.

C.2. Feature visualization figures (Section 2)

Throughout the paper, feature visualizations were generated using the `lucent` library (Greentfrapp, v0.1.8), version v0.1.8. Per default, we used `thresholds=(512,)` except for Figure 6 where the five images at different points in the optimization trajectory are shown (specifically, `thresholds=(1,8,32,128,512)`). For Figure 1, we used the thresholds that visually looked best (in line with existing literature: there is no principled way to determine the threshold); specifically `thresholds=(512,512,512,6,32,6)` for the six visualizations from left to right (for the three rightmost images, higher thresholds produced qualitatively similar yet oversaturated images). In terms of transformations during feature visualization, `transforms=lucent.optvis.transform.standard_transforms + [center_crop(224, 224)]` was used. The image was parameterized via `param_f=lambda: lucent.optvis.param.image(224, batch=1)`.

For Figure 1, a natural image was embedded into the weights of a single convolutional layer, `torch.nn.Conv2d`, with `kernel_size=224, stride=1, padding=0, dilation=1, groups=1, bias=True, padding_mode='zeros')`. To this end, the image was loaded and the layer weights were set to the corresponding image values, divided by 224^2 to avoid a potential overflow. Architecturally, the layer received the standard image input and its ouptut was appended to the output of the desired layer (e.g., `softmax2_pre_activation_matmul`) where it was used in the role of D from Figure 3.

Figure 6: **Natural vs. synthetic distribution shift.** There is a clear distribution shift between feature visualizations (left) and natural images (right). This can be exploited by a classifier when building a fooling circuit. Visualizations at different steps in the optimization process for a randomly selected unit in the last layer of standard, unmanipulated Inception-V1; randomly selected ImageNet validation samples (excluding images containing faces).

C.3. Sanity check (Section 3)

Motivation: relationship between path similarity and Spearman correlation. In Section 3, we describe that different processing paths lead to different activation similarity as measured through Spearman correlation. We here attempt to explain this relationship in a bit more detail. For the context of our analysis, we define a path as a (sub-)graph of a directed acyclic graph (DAG, a neural network or sub-network in our case), starting at the input nodes (first layer units) and ending at a single unit (the unit for which the analysis is performed). How can we quantify the overlap between two different paths, layer by layer? If a node is in the subgraph forming the path, the node is assigned a value of 1; if it is not, it is assigned a value of 0. Then, layer-wise overlap can be quantified by the Spearman correlation, which is exactly zero if there is only chance overlap, exactly 1.0 if there is perfect overlap, and exactly -1.0 if the units in a certain layer (corresponding to two different paths) are perfectly anticorrelated. Similarly, in the non-binary case (such as a path formed by activation patterns for natural images vs. feature visualization images, which is what we consider for the similarity analysis), the values assigned to the node are simply the activations, and the same analysis can be applied. Since we only care about

the path similarity and not about whether this similarity is a linear relationship, Spearman's rank-order correlation is the correct measure to use here (while the Pearson correlation, plotted in Figure 10 for comparison, is a measure of a linear relationship).

Methods. Performing a full comparison is computationally expensive: Inception-V1's largest layer (`conv2d0_pre_relu_conv`) contains 802,816 values; computing the Spearman correlation for this takes about one third of a second. Inception-V1 has 138 layers and sub-layers (see Figure 7 x-axis labels for a list). Even if we just consider the comparison between natural images vs. natural images of a different class, for the ImageNet-1K validation set this amounts to 138 (= number of layers) $\cdot$ 50 $\cdot$ 50 (= number of comparisons between two specific classes of the ImageNet validation split) $\cdot \frac{1000 \cdot 1001}{2} - 1000$ (= number of comparisons when comparing each class with each other class except for itself) $= 172,327,500,000$ comparisons. With 3 comparisons per second, this amounts to about *eighteen hundred years* required to do the full comparison. In order to make this more feasible, we chose the following approach. We randomly selected 10 classes via `numpy.random.seed(42); randomly_selected_class_indices = sorted(numpy.random.choice(1000, 10))`, resulting in `randomly_selected_class_indices=[20, 71, 102, 106, 121, 270, 435, 614, 700, 860]` and obtained 10 feature visualizations per class. Images that were not correctly classified by the model (wrong top-1 classification) were excluded from the comparison since those images do not constitute natural images for which the corresponding unit is selective for.

From this point onward, when computing the Spearman and Pearson correlations, we only performed every 10th comparison for natural images vs. natural images of the same class; every 100th comparison for natural images vs. natural images of a different class, and every 5th comparison for natural images vs. feature visualizations of the same class. For Cosine similarity (which is much faster), we performed every single comparison.

Raw and normalized plots. For each metric, *absolute* values are plotted in Figures 7, 9, and 11. For Figures 5, 8, and 10, *normalized* values are plotted. For these plots, we normalized the data according to the raw absolute values, i.e., such that natural images vs. natural images of the same class is set to 1.0 and natural images vs. natural images of a different class is set to 0.0 since it makes sense to interpret similarity results relative to those two extreme baselines. A tiny number of layers was excluded from the normalized comparison if the green and black points from the absolute plots differed by strictly less than a threshold of 0.01. To reduce noise in the orange curve (natural images vs. feature visualizations of the same class), we smoothed the curve by convolving it with `scipy.ndimage.convolve` using a window size of 7 and the following uniform weights: `np.ones(windowsize)/windowsize`. The shaded blue area corresponds to the standard deviation of the orange data points, convolved over a window of of size `std_windowsize`=5 which is then (for the lower bound of the blue area) subtracted from the orange curve, and (for the upper bound of the blue area) added to the orange curve; thus in total the blue area area vertically extends two standard deviations. The idea behind this is to give a rough visual estimate of the standard deviation range that the orange values have at certain points throughout the network. By itself, it does not provide an indication of statistical significance.

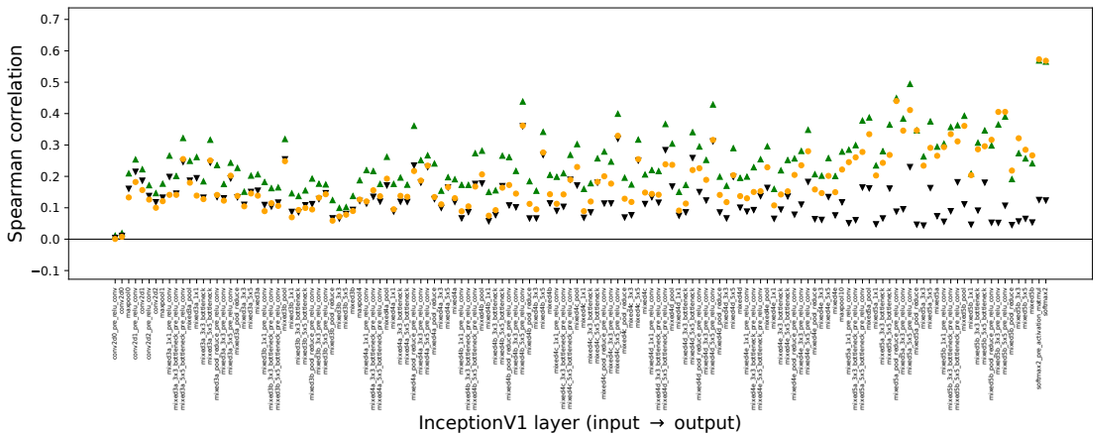


Figure 7: Absolute similarity (Spearman).

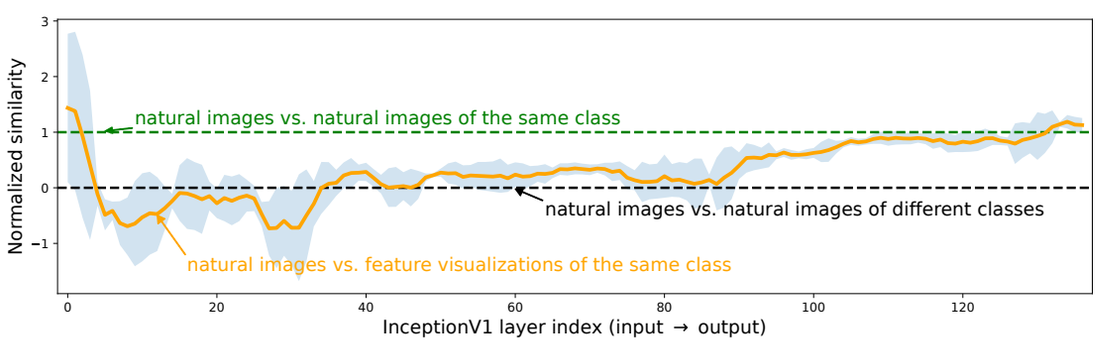


Figure 8: Normalized similarity (Cosine).

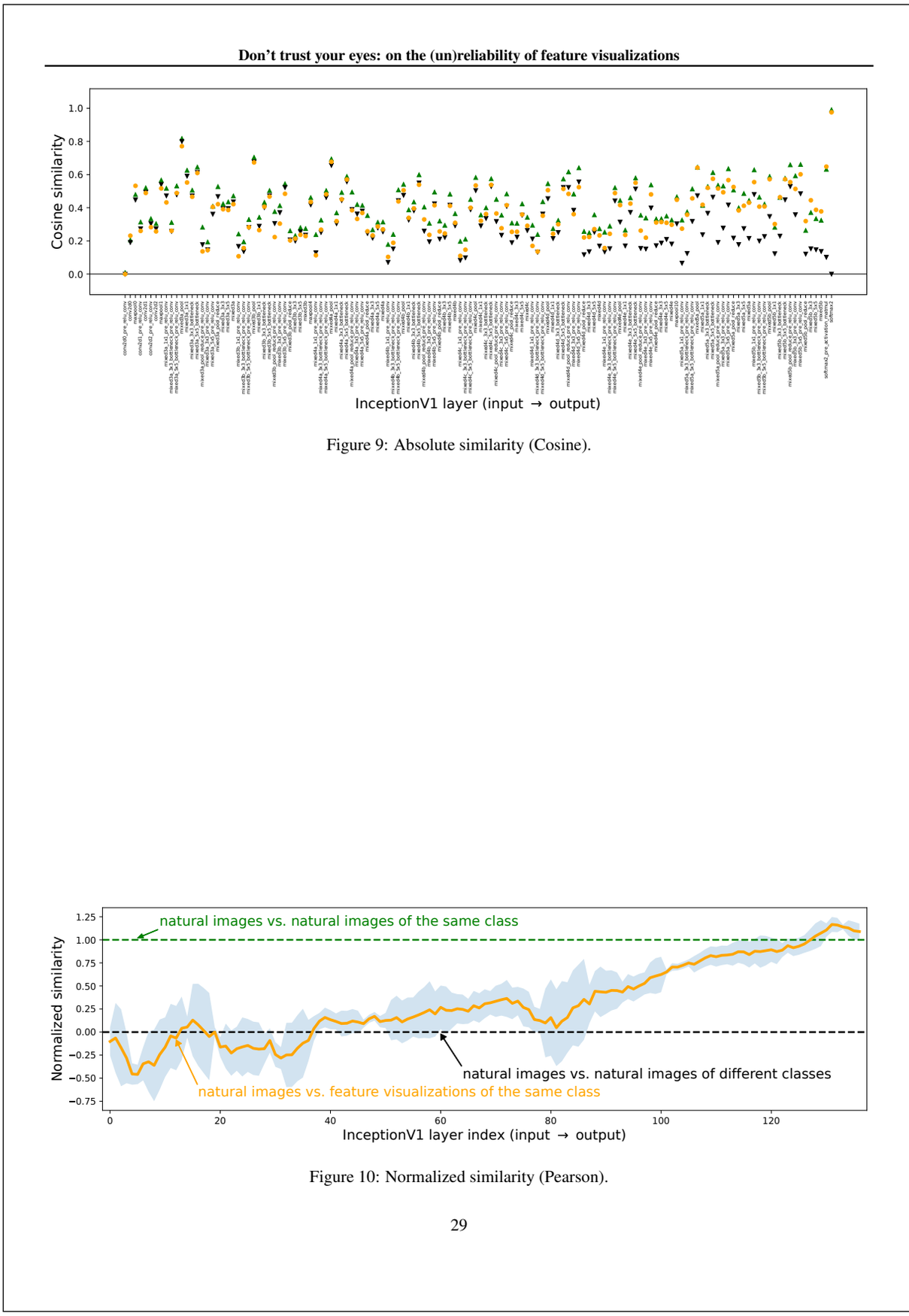


Figure 9: Absolute similarity (Cosine).

Figure 10: Normalized similarity (Pearson).

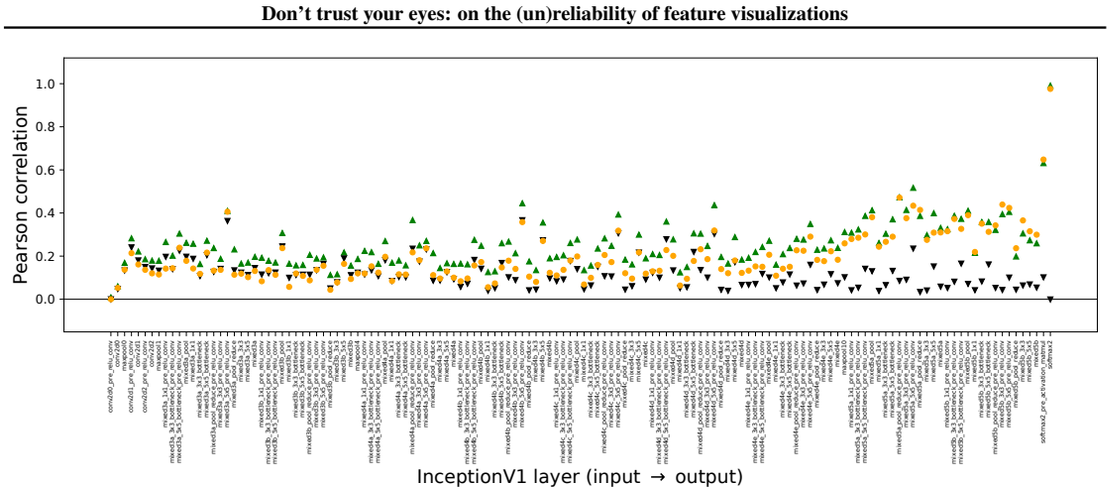


Figure 11: Absolute similarity (Pearson).

D. Are more linear units easier to interpret?

Our theory makes a prediction: the simpler a function is, the easier it should be to interpret given a feature visualization. We here empirically test this prediction. As a measure of simplicity, we use *path linearity*: based on a highly activating natural image (start point), we perform feature visualization to arrive at a highly activating optimized image (end point). We then analyze how much the optimization path deviates from linearity by measuring the angle between gradients of the first n steps of the path. This metric (across many such paths) is then compared to human experimental data from Zimmermann et al. (2024) for the same units, who measured how well humans can interpret different units in Inception-V1. Intriguingly, we find a significant correlation (Spearman's $r = -.36$, $p = .001$) between this human interpretability score and our path linearity measure (lower angle means higher interpretability) especially for the beginning of the trajectory (e.g., $n = 2$); a plot can be found below. Overall, we interpret this as preliminary evidence in favor of the hypothesis that more linear units, at least at the beginning of the optimization trajectory, might be easier to interpret. An interesting direction for future work would be to enforce higher degrees of linearity, for instance through regularization or the architecture. This is one example of how our theory might be used to develop hypotheses for better feature visualizations.

Methods. To measure the interpretability of a unit we use the experimental data provided by Zimmermann et al. (2024). Based on the experimental paradigm by Borowski et al. (2021) and Zimmermann et al. (2021), Zimmermann et al. (2024) tested how well humans can differentiate maximally and minimally activating images for individual units of a CNN when supported with explanations in the form of feature visualizations (see Appendix B.7 for details). We use the experimental data of 84 units as well as the $M = 20$ maximally activating natural dataset samples (from ImageNet) they provided.

Quantifying degree of nonlinearity through gradient angles. To measure the linearity of a unit we compute the following quantity for each unit: We start from a maximally activating dataset sample x_i^s and perform feature visualization to iteratively optimize this image to further increase the unit's activation. We use standard hyperparameters and optimize for $N = 512$ steps. During optimization we record the normalized gradients with respect to the current image $(\hat{g}_j(x_i^s))_{j=1,\dots,N}$ and compute the angle between successive steps:

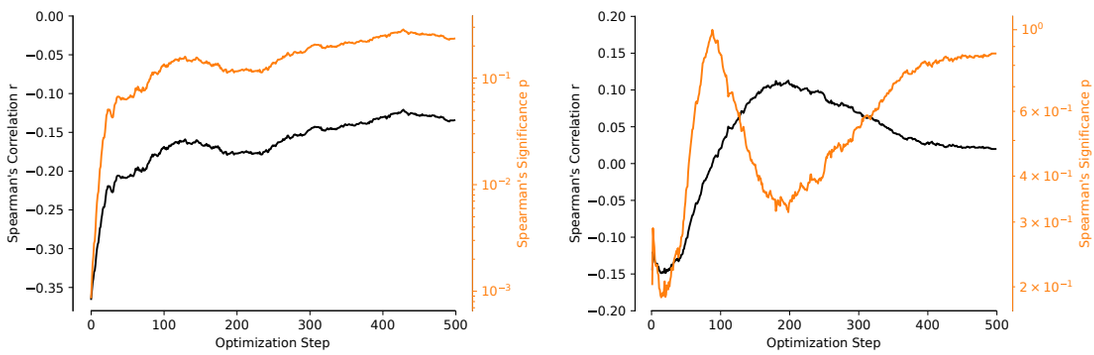
$$\forall j = 1, \dots, N-1: \quad a_j(x_i^s) := \measuredangle(\hat{g}_j(x_i), \hat{g}_{j+1}(x_s^s)). \tag{12}$$

We take the average over all M maximally activating images and denote the *average gradient path angle* as:

$$\forall j = 1, \dots, N-1: \quad \text{AGPA}_j := \frac{1}{M} \sum_{i=1}^{M} a_j(x_i^s). \tag{13}$$

Finally, we denote the average of the average gradient path angle AGPA as the average gradient angle:

$$\mathrm{AGA} = \frac{1}{N-1} \sum_{i=1}^{N-1} \mathrm{AGPA}_i \,. \tag{14}$$



(a) Average gradient path angle AGPA_k. Strikingly, only the first few gradient angles are strongly and significantly correlated with the units' interpretability.

(b) Average line distance path AGLD_k. Interestingly, while we see a anti-correlation at the beginning (cf. Figure 12a), this turns into a weak correlation. However, these correlations are not significant.

Figure 12: Development of Spearman's rank correlation between the units' interpretability score and the (a) average gradient path angle AGPA_k and (b) average line distance path ALDP_k as a function of the number of optimization steps k to consider.

To answer our initial question—whether simpler/more linear units are more interpretable—we now measured the rank correlation between the average gradient angle and the interpretability scores by Zimmermann et al. (2024) based on the paradigm of Zimmermann et al. (2021). Intriguingly, we find a significant correlation (Spearman's $r = -.36$, $p = .001$) between this human interpretability score and our path linearity measure (lower angle means higher interpretability) for the beginning of the trajectory (e.g., $n = 2$). Linearity at later steps in the trajectory does not seem to contribute much to human interpretability, thus increasing n to include all 512 steps decreases the overall correlation. The results depending on path length are plotted in Figure 12a.

Quantifying degree of nonlinearity through deviations from linear interpolation. There are many different ways that could be used to measure path or unit linearity. As a more global measure, we also tested another one: Here, we begin by computing a linear interpolation between the maximally activation data samples we initialize the optimization with x_i^s and the final visualization x_i^f:

$$\{\, z \mid x_i^s + \alpha(x_i^f - x_i^s) \quad \forall \alpha \in [0,1] \,\}. \tag{15}$$

Next, for each step j of the optimization process we compute the distance of the current image $x_j(x_i^s)$ to to the linear interpolation

$$d_j(x_i^s) = d\left(x_j((x_i^s)), \{\, z \mid x_i^s + \alpha(x_i^f - x_i^s) \quad \forall \alpha \in [0,1] \,\}\right), \tag{16}$$

where $d(\cdot,\cdot)$ represents the ℓ_2 distance. Analogously to the computation above, we then take the mean over the different start images and define this property as the average line distance path

$$\forall j = 1, \ldots, N-1: \quad \mathrm{ALDP}_j := \frac{1}{M} \sum_{i=1}^{M} \frac{d_j(x_i^s)}{d(x_i^2, x_i^f)}, \tag{17}$$

where we normalize the distances from the line with the distance between start and end point of the optimization trajectory to make these values scaleless. Finally, by averaging again over optimization steps final average line distance:

$$\mathrm{ALD} = \frac{1}{N-1} \sum_{i=1}^{N-1} \mathrm{ALDP}_i \,. \tag{18}$$

The lower the ALD value, the smaller is the deviation of the optimization path from a line. For this global measure, we see a non-significant relation between the average line distance and the interpretability scores (Spearman's $r = 0.02$, $p = 0.86$). Analogously to the local measure explained above (gradient angle), we zoom into these results again in Figure 12b. While there might be a weak anti-correlation at the beginning of the optimization path (which later turns into a weak correlation), none of these are significant.

Interpretation. We believe there is more to be understood: while it is intriguing that linearity at the beginning of the optimization trajectory is predictive of a human interpretability score, the global measure of linearity (through distance to linear interpolation) does not seem to be predictive and more investigations are needed to fully understand this phenomenon—as we say in the main paper, this can only be considered "very preliminary evidence". A promising way forward could be to enforce (or optimize for) different properties in neural networks and measure whether this improves human interpretability.

E. Do fooling methods change model behavior on OOD input?

We tested both methods from Section 2 on two OOD datasets and validated that neither of them changes the behavior of the models: On ImageNet-V2, the silent unit method shows 100.0% identical predictions to the unmanipulated model (10,000 out of 10,000); for the fooling circuit we get 99.52% identical predictions (9,952 out of 10,000). The 0.5% different responses approximately match the delta on the standard, unmodified ImageNet validation set (99.49% identical predictions as reported in Section 2.1). On NINCO ("No ImageNet Class Objects"), a dedicated dataset with a much stronger distribution shift (zero overlap with ImageNet classes), the silent unit method again achieves 100.0% identical predictions to the unmanipulated model (5,878 out of 5,878); the fooling circuit method achieves 98.62% identical predictions (5,797 out of 5,878). This indicates that both methods are fairly robust towards natural input from a different distribution compared to the ImageNet training dataset.

F. Fooling feature visualizations of adversarially robust models

In Section 2, we presented two proofs of concept that feature visualizations can manipulated: one based on a fooling circuit, and one leveraging silent units. We empirically demonstrated this for two standard vision models. Our theoretical results in Table 1 in Section 4 indicate that feature visualizations become more reliable for networks with a sufficiently small Lipschitz constant. At the same time, a lower Lipschitz constant is also connected with higher adversarial robustness of models (Hein & Andriushchenko, 2017). Therefore, a natural question is: Can feature visualizations of (more) robust models still be manipulated? We investigate this question for adversarially trained ResNet50 ($\ell_\infty, \epsilon = 4/255$) by (Salman et al., 2020).

Fooling circuit Integrating a fooling circuit is guaranteed to manipulate the visualizations of a model without changing its overall behavior as long as one can clearly distinguish between feature visualizations and standard dataset examples (see Section 2.1). Therefore, to demonstrate the this approach also works for an adversarial robust model, it suffices to show that we can almost surely detect whether an input is a feature visualization. Qualitatively, this is suggested by an analogy of Figure 6, i.e., a visual comparison of feature visualizations and dataset samples, in Figure 13. Quantitatively, we tested the *same* feature visualization detector that we used for a non-robust Inception-V1 model in Section 2.1 for the robust ResNet50 and see strong generalization results: the binary classifier's accuracy is still very high at 99.19 %.

Silent units To apply the methodology presented in Section 2.2 to an adversarial robust model, only a single change needs to be implemented: Namely, we noticed that the ranges of activations caused by feature visualizations and dataset examples, respectively, are closer for a robust than for a non-robust model. For some units, there even exist natural dataset examples that elicit slightly higher activation than feature visualizations. We, therefore, need to adjust how we choose the bias b in Eq. (11): Instead of using a value proportional to the maximal activation value recorded on any natural test sample, we use a value proportional to the 99th percentile of the activation range for test samples. While this change ensures that we can manipulate the visualizations of more/all units, it comes with a small price: Namely, the network's behavior on natural

Figure 13: **Natural vs. synthetic distribution shift for a robust model.** As shown above for the Inception-V1 model in Figure 6, there is still a clear distribution shift between feature visualizations (left) and natural images (right) an adversarial robust model (ResNet50). Visualizations at different steps in the optimization process for a randomly selected unit in the last layer of standard, unmanipulated but robust ResNet50; randomly selected ImageNet validation samples (excluding images containing faces).

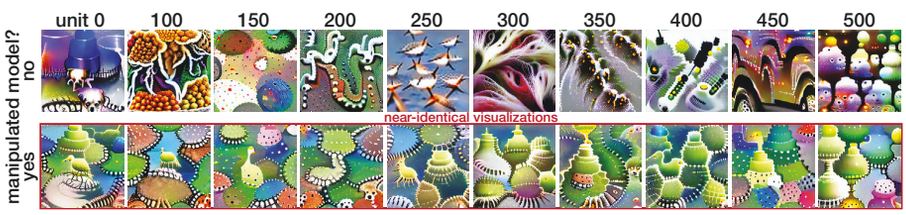


Figure 14: **Leveraging silent units to produce identical visualizations throughout a layer of a robust model.** Replication of Figure 4 for a robust ResNet50 model. The top row shows feature visualizations for units of a layer (block 4-1, conv 2) in a robust but unmanipulated ResNet-50. For the bottom row, we manipulate the model such that the feature visualizations of all units become near-identical (indicated by the red box).

samples does not remain unchanged (as for a non-robust ResNet50, cf., Section 2.2) but drops very slightly from $63.924\,\%$ to $63.784\,\%$. The analogue of Figure 4 for the robust model, Figure 14, shows that even a robust model can be manipulated to show near-identical feature visualizations.

G. Do feature visualizations fail the sanity check simply because they are out-of-distribution?

In Section 3, we showed that feature visualizations fail a sanity check. An anonymous reviewer asked whether this might be due to the case that they are out-of-distribution (compared to natural images)—an interesting question that we decided to investigate. We therefore performed the analysis for two complementary additional datasets: ImageNet-V2 (Recht et al., 2019) which has the same classes as original ImageNet and only a small distribution shift as evidenced by a roughly 10–15% accuracy drop of standard models, and NINCO (Bitterwolf et al., 2023), a dedicated out-of-distribution detection dataset that specifically contains no classes that are part of original ImageNet. We run the sanity check analysis with those datasets separately by computing the same baselines as before (natural images vs. natural images of the same class; natural images vs. natural images of different classes) with the difference that those natural images are now sampled from the respective OOD dataset, not from ImageNet. We then plot the previously computed similarity between feature visualizations vs. natural ImageNet images of the same class in relationship to those new baselines. The key idea is that if a simple out-of-distribution shift is responsible for strongly decreased processing similarity, then those new baselines should be drastically lower. At the same time, the normalized similarity between feature visualizations and natural ImageNet images (plotted in orange) should be a lot higher than it is in Figure 5 since it is now normalized with respect to the out-of-distribution baselines.

As we can see for ImageNet-V2 in Figure 15 and for NINCO in Figure 17, this is *not* the case: even though there is a substantial distribution shift, the similarity between same-class images of those OOD datasets is still a lot higher than the natural-vs-feature-visualization similarity on ImageNet throughout the network (except for the last few layers). We conclude from this analysis that simple distribution shifts are insufficient to explain the different processing paths of feature visualizations—either the distribution shift does not play a role, or the distribution shift from feature visualizations is much larger even than the ImageNet-NINCO distribution shift (keeping in mind that NINCO is a dedicated out-of-distribution detection dataset aimed at providing a much more systematic shift than many other datasets).

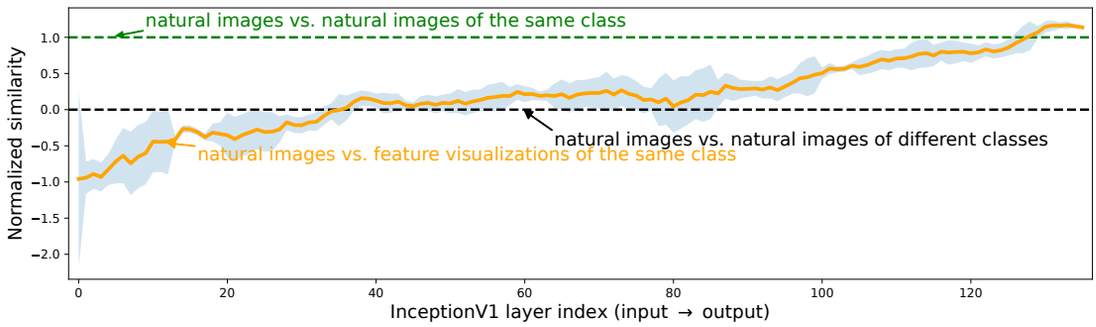


Figure 15: Normalized similarity (Spearman) for ImageNet-V2.

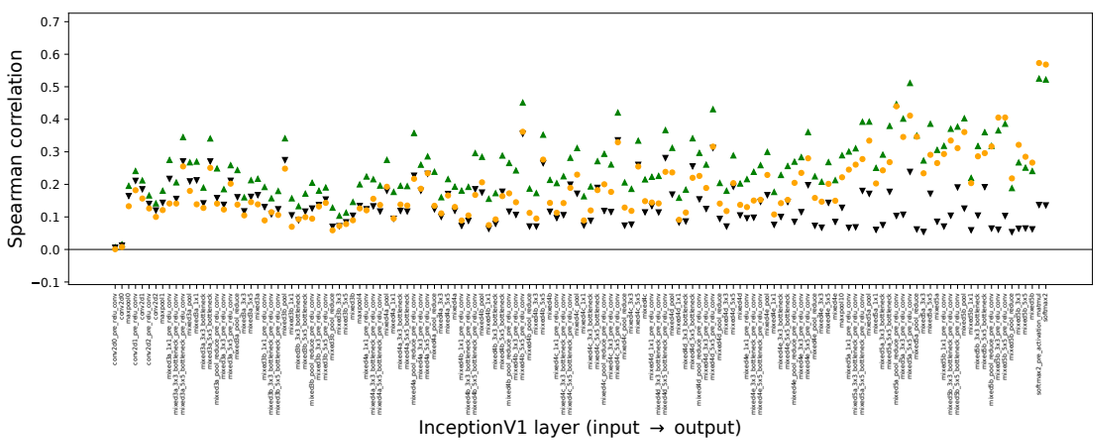


Figure 16: Absolute similarity (Spearman) for ImageNet-V2.

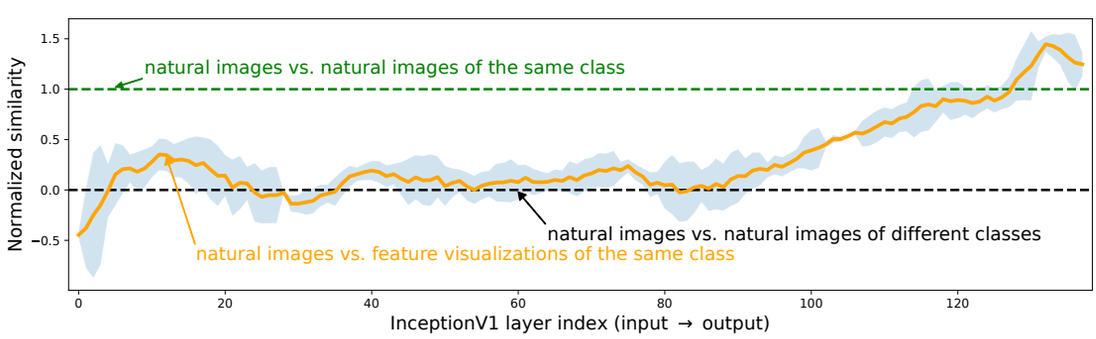


Figure 17: Normalized similarity (Spearman) for NINCO.

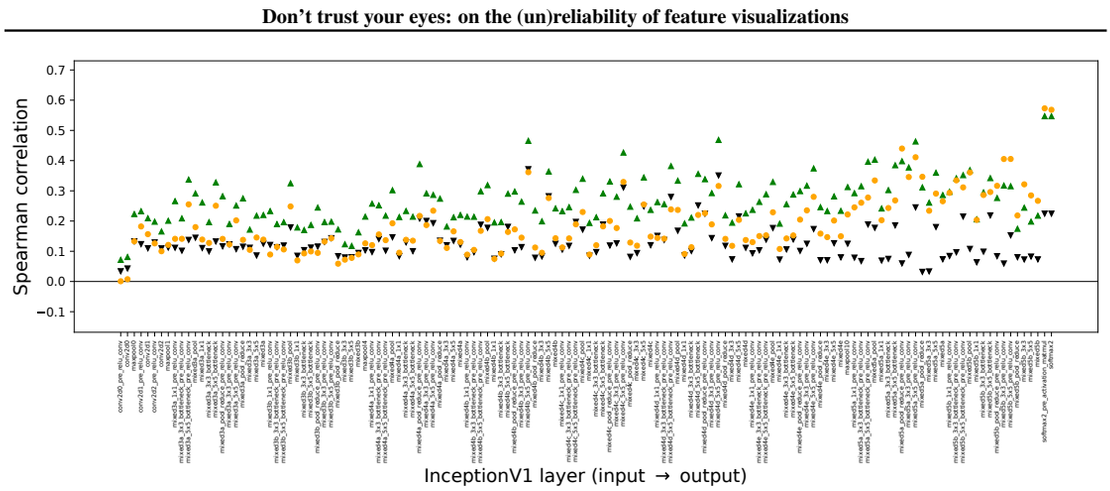


Figure 18: Absolute similarity (Spearman) for NINCO.

H. Silent unit manipulation: sensitivity studies

This section extends the empirical results above and investigates the sensitivity of the fooling method based on silent units with respect to different (external) choices: Does the fooling method work for multiple layers at once? Does it work for earlier layers? Does its success depend on how feature visualizations are initialized?

Fooling visualizations with different initial images. The feature visualization shown above are all generated through gradient descent on the input, starting with a random Gaussian noise image. As the choice of the initial image might influence the success rate of the proposed fooling method we here test another initialization: Figure 19 shows how visualizations for the same units as in Figure 4 look when initialized with a randomly selected natural seed image as initialization. As shown in the bottom row, for a manipulated model we still obtain the same result, i.e., near-identical visualizations for each unit in the entire layer.

Figure 19: Feature visualizations initialized with a natural seed image can also be fooled.

Fooling units from early layers. While Figure 4 demonstrates the success of the fooling method for a mid layer of a ResNet, Figure 20 shows it also works for an earlier layer.

Fooling units from multiple layers together. Moreover, we find that multiple (earlier) layers (layer3_2_conv2 and layer3_2_conv2) can be manipulated at the same time, as shown in Figure 21.

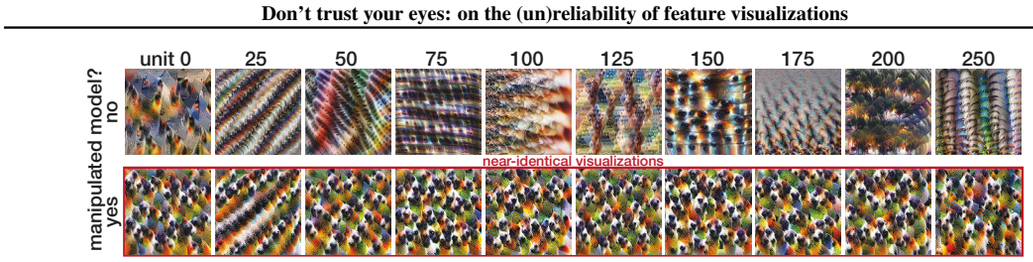


Figure 20: The proposed fooling method does not only work for mid/late layers but also for early ones.

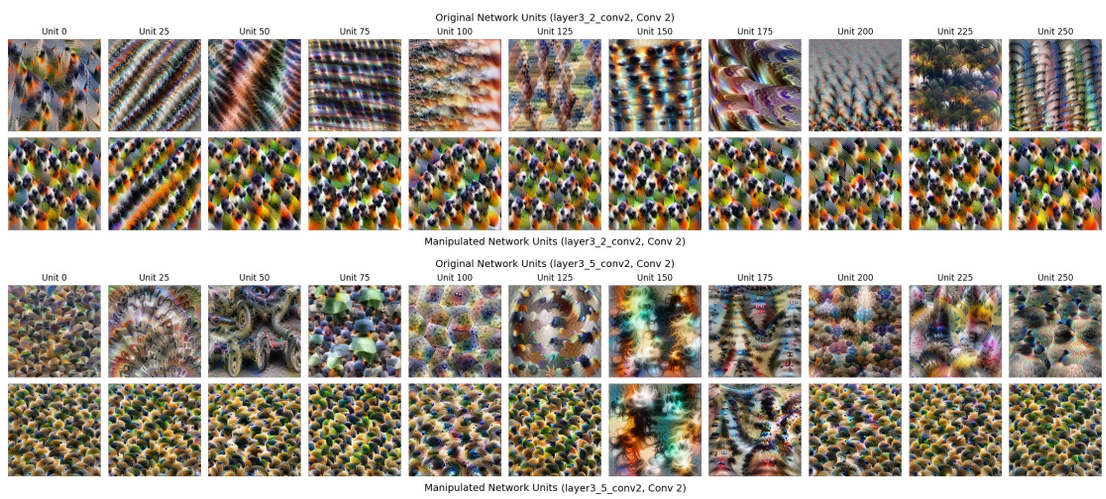


Figure 21: The fooling method can be successfully applied to multiple layers at once.

I. What happens when a slightly different method for visualization is used which is considered out of distribution by the classifier?

While it would be fairly easy to train the classifier with more data augmentation / different feature visualizations, our existing classifier is already robust enough to handle variations in the input distribution. We conducted an experiment where we kept the classifier fixed and systematically varied the distribution by changing the transformations applied during the feature visualization optimization approach. The lucent library uses a standard set of transformations including jitter(8), random_scale, random_rotation, followed by jitter(4). Even if we switch all of those off, which results in substantially different visualizations, the classifier still reliably detects the feature visualizations with the exact same level of accuracy. Therefore, those changes are insufficient to circumvent our fooling method.

J. Limitations

We see the following potential limitations:

1. We design methods that fool feature visualizations. Once it is known that a certain fooling method might be used, it is easy to develop a detection mechanism. That said, the space of potential fooling methods is vast. Therefore, developing a specific detection mechanism would probably lead to a pattern similar to adversarial attacks and defenses: after an attack is developed, a detection/circumvention method defends, which is then again circumvented by a revised attack/fooling method.

2. The fooling methods that we developed in Section 2 assume bad intent. Most models are developed with good intent. However, we believe that the reliability of interpretability methods should not rely on assuming good intentions. The

experiments in Section 3 and the theory from Section 4 are independent of good/bad intent assumptions.

3. No sanity check is perfect, and like any sanity check, ours is just a necessary but not a sufficient condition for a reliable feature visualization. For instance, if the training data contains spurious correlations (all cows are on grass); a unit in the network is selective for grass and not for cows but the feature visualization shows realistic-looking cows on grass (rather than just grass), then the visualizations would pass the sanity check without raising a warning about the spurious correlation present in the visualization. We would love to see more research on sanity checks—to the best of our knowledge we provide the first one which hopefully serves as a motivation for research on both better visualizations and more sanity checks.

4. The potential assumptions on the function space listed in Table 1 are not exhaustive. It is possible that other assumptions enable stronger prediction. Furthermore, our theory is a worst-case analysis. It may be possible to go beyond worst-case analyses—an aspect we discuss in Appendix B.6—but the strength of our theoretical counterexamples is that they can be realized by very simple functions, hence we are not "cherry-picking" complicated functions that SGD would never learn.

5. The investigated networks, Inception-V1 and ResNet-50, are of course not exhaustive either. None of our methods is specific to those networks. This means that other networks could be equipped with a fooling circuit, too. At the same time, the empirical results might look different for other networks, which would be an interesting direction to explore in future work.

6. One could argue that the fooling circuit in Figure 3 only deceives a user when looking at unit A, whereas the other units still have their original visualization. That's correct: the fooling circuit manipulates the visualization of A but not of e.g., units D or F. From a single unit perspective, this is already problematic since it means we can't trust a unit's visualization. It would be interesting avenue for future work to develop networks where every single unit's visualization is misleading.

7. There is no one definition of what it means to "understand" or "explain" a neural network, since those are very vague terms. We seek to be precise about our definition and motivate it with expectations about feature visualization stated in the literature (Appendix A.2), but we realize that this means not everyone's notion of "understanding" / "explaning" neural networks can be captured by our definition.

8. For the silent unit approach, it might be possible to make a feature visualization approach that does not rely on maximizing the neuron output but rather just increasing it to below some chosen threshold. We see a parallel to adversarial attacks and defenses here: given a fixed feature visualization, it is easy to come up with a fixed manipulated model (as we show). Given a fixed manipulated model, it is probably not difficult to circumvent the manipulation technique with an updated feature visualization, and the circle continues—just like for adversarial attacks, where it is easy to defend if the attack is kept fixed. That said, this wouldn't fix the underlying problem and can easily be circumvented through an adaptive attack. Therefore, the adversarial attack community has called for (and largely agreed on) an adaptive approach (Tramer et al., 2020). It is possible that we might see a similar pattern of cat-and-mouse-games for feature visualizations in the years to come.

K. Image sources

Figure 1. "Girl with a Pearl Earring" by Johannes Vermeer was downloaded from here and is public domain according to the website. "Puppies" (West Highland White Terrier puppies) by Lucie Tylová, Westik.cz was downloaded from here and is licensed under CC BY-SA 3.0 according to the website. "Mona Lisa" by Leonardo da Vinci was downloaded from here and is public domain according to the website. All three images were cropped to size 224×224 pixels. The photographs were then embedded in the weights of a single convolutional layer and to some degree recovered by the feature visualization method, subject to distortion by the method's transformations.

A.5 Measuring Interpretability at Scale Without Humans

The following 36 pages were published as:

Roland S. Zimmermann, David Klindt, and Wieland Brendel. "Measuring Interpretability at Scale Without Humans." *NeurIPS (2024)*

A summary is given in Section 2.4 on page 41. * Equal contribution.

Abstract

In today's era, whatever we can measure at scale, we can optimize. So far, measuring the interpretability of units in deep neural networks (DNNs) for computer vision still requires direct human evaluation and is not scalable. As a result, the inner workings of DNNs remain a mystery despite the remarkable progress we have seen in their applications. In this work, we introduce the first scalable method to measure the per-unit interpretability in vision DNNs. This method does not require any human evaluations, yet its prediction correlates well with existing human interpretability measurements. We validate its predictive power through an interventional human psychophysics study. We demonstrate the usefulness of this measure by performing previously infeasible experiments: (1) A large-scale interpretability analysis across more than 70 million units from 835 computer vision models, and (2) an extensive analysis of how units transform during training. We find an anticorrelation between a model's downstream classification performance and per-unit interpretability, which is also observable during model training. Furthermore, we see that a layer's location and width influence its interpretability.

Measuring Per-Unit Interpretability at Scale Without Humans

Roland S. Zimmermann
MPI-IS, Tübingen AI Center

David Klindt
Stanford

Wieland Brendel
MPI-IS, Tübingen AI Center

Abstract

In today's era, whatever we can measure at scale, we can optimize. So far, measuring the interpretability of units in deep neural networks (DNNs) for computer vision still requires direct human evaluation and is not scalable. As a result, the inner workings of DNNs remain a mystery despite the remarkable progress we have seen in their applications. In this work, we introduce the first scalable method to measure the per-unit interpretability in vision DNNs. This method does not require any human evaluations, yet its prediction correlates well with existing human interpretability measurements. We validate its predictive power through an interventional human psychophysics study. We demonstrate the usefulness of this measure by performing previously infeasible experiments: (1) A large-scale interpretability analysis across more than 70 million units from 835 computer vision models, and (2) an extensive analysis of how units transform during training. We find an anti-correlation between a model's downstream classification performance and per-unit interpretability, which is also observable during model training. Furthermore, we see that a layer's location and width influence its interpretability. Online version, code and interactive visualizations available at brendel-group.github.io/mis.

1 Introduction

With the arrival of the first non-trivial neural networks, researchers got interested in understanding their inner workings [24, 26]. For one, this can be motivated by scientific curiosity; for another, a better understanding might lead to building more reliable, efficient, or fairer models. While the performance of machine learning models has seen a remarkable improvement over the last few years, our understanding of information processing has progressed more slowly. Nevertheless, understanding how complex models — e.g., language models [7] or vision models [34, 50] — work is still an active and growing field of research, coined *mechanistic interpretability* [33]. A common approach in this field is to divide a network into atomic units, hoping they are easier to comprehend. Here, atomic units might refer to individual neurons or channels of (convolutional) layers [34], or general vectors in feature space [12, 23]. Besides this approach, mechanistic interpretability also includes the detection of neural circuits [8, 12] or analysis of global network properties [29].

The goal of understanding the inner workings of a neural network is inherently human-centric: Irrespective of what tools have been used, in the end, humans should have a better comprehension of the network. However, measuring interpretability through human evaluations is time-consuming and costly due to their reliance on human labor [50]. This results in slower research progress, as validating novel hypotheses takes longer. Removing the need for human labor by automating the interpretability measure can open up multiple high-impact research directions: First, it enables the creation of more interpretable networks by explicitly optimizing for interpretability — after all, what we can measure at scale, we can optimize. Second, it allows more efficient research on explanation methods and might increase our understanding of neural networks. Due to the lack of a reliable automated measure, previous work resorted to limited time-consuming human evaluations, partially producing inconclusive results [e.g., 7, 39], highlighting the urgency of finding an automated measure.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

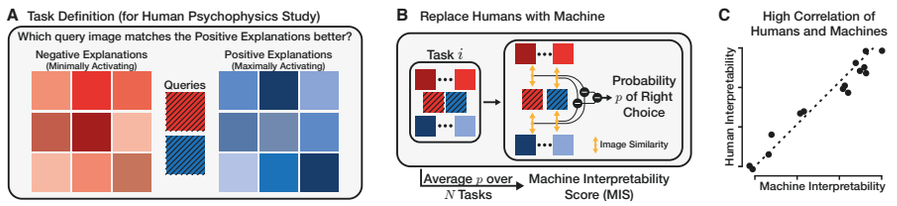


Fig. 1: **Definition of the Machine Interpretability Score. A.** We build on top of the established task definition for quantifying the per-unit interpretability via human psychophysics experiments [6]. The task measures how well participants understand the sensitivity of a unit by asking them to match strongly activating query images to strongly activating *visual* explanations of the unit. Red and blue squares illustrate the unit's minimally and maximally activating images; shaded and solid squares denote natural test images and explanations, respectively. See Fig. 9 for examples. **B.** Crucially, we remove the need for humans and fully automate the evaluation: We pass the explanations and query images through a feature encoder to compute pair-wise image similarities (DreamSim) before using a (hard-coded) binary classifier to solve the underlying task. Finally, the Machine Interpretability Score (MIS) is the average of the predicted probability of the correct choice over N tasks for the same unit. **C.** The MIS proves to be highly correlated with human interpretability ratings and allows fast evaluations of new hypotheses.

The present work is the first to introduce a fully automated interpretability measure (Fig. 1A & B) for vision models: the Machine Interpretability Score (MIS). By leveraging the latest advances in image similarity functions aligned with human perception, we obtain a measure that is strongly predictive of human-perceived interpretability (Fig. 1C). We verify our measure through both correlational and interventional experiments. By removing the need for human labor, we can scale existing evaluations up by multiple orders of magnitude. Finally, this work demonstrates potential workflows and use cases of our MIS.

2 Related Work

Mechanistic Interpretability While the overall field of explainable AI (XAI) tries to increase our understanding of neural networks, multiple subbranches with different foci exist [15]. One of these branches, *mechanistic interpretability*, tries to improve our understanding of neural networks by understanding their building blocks [33]. An even more fine-grained branch — per-unit mechanistic interpretability — aims to interpret individual units of vision models [3, 48, 4, 27, 34]. We focus exclusively on this branch of research in the present work. This line of research for artificial neural networks was, arguably, inspired by similar efforts in neuroscience for biological neural networks [20, 2, 37].

Different studies set out to understand the behavior and sensitivity of individual units of vision networks – here, a unit can, e.g., be (the spatial average of) a channel in a convolutional neural network (CNN) or a neuron in a multilayer perceptron (MLP). The level of understanding obtained for a unit is commonly called the *per-unit interpretability*; by averaging over a representative subset of units in the network, one obtains the *per-model interpretability* [50]. With the recent progress in vision-language modeling, a few approaches started using textual descriptions of a unit's behavior [18, 21]. However, the majority still uses visual explanations which are either synthesized by performing activation maximization through gradient ascent [34, 13, 26, 30, 28, 46, 31], or strongly activating dataset examples [34, 6]. With the increasing usage of large language models (LLM), there is also now an increasing interest in mechanistic interpretability of them [e.g., 11, 36, 7].

Quantifying Interpretability Rigorous evaluations, including falsifiable hypothesis testing, are critical for research on interpretability methods [25]. This also encompasses the need for human-centric evaluations [6, 22]. Nevertheless, such human-centric evaluations of interpretability methods are only available in some sub-fields. Specifically for the type of interpretability we are concerned about in this work, i.e., the per-unit interpretability of vision models, two methods for quantifying the helpfulness of explanations to humans were introduced before: Borowski et al. [6] presented a two-alternative-forced-choice (2-AFC) psychophysics task that requires participants to determine

which of two images elicits higher activation of the unit in question, given visual explanations (i.e., images that strongly activate or deactivate the unit, see Fig. 1A) of the unit's behavior. Zimmermann et al. [49] extended this paradigm to quantify how well participants can predict the influence of interventions in the form of occlusions in images. While these studies used their paradigms to evaluate the usefulness of different interpretability methods, Zimmermann et al. [50] leveraged them to compare the interpretability of models. Due to the reliance on human experiments, they could only probe the interpretability of 767 units from nine models. We now automatize this evaluation to scale it up by multiple orders of magnitude to more than 70 million units across 835 models.

Automating Interpretability Research To increase the efficiency of interpretability research and scale it to large modern-day networks, the concept of automated interpretability was proposed in the domain of natural language processing [5]. This approach uses an LLM to generate textual descriptions of the behavior of units in another LLM. Follow-up work by Huang et al. [19], however, pointed out potential problems regarding the correctness of the explanations. Besides automating interpretability research of individual units, there are also efforts for automating the discovery and interpretation of neural circuits and subnetworks [9, 43]. To benchmark future fully automated interpretability tools, acting as independent agents, Schwettmann et al. [41] introduced a synthetic benchmark suite inspired by the behavior of neural networks. In computer vision, there are also efforts to automate interpretability research [18, 50]. Hernandez et al. [18] and Oikarinen and Weng [32] map visual to textual explanations of a unit's behavior using automated tools, hoping to increase the efficiency of evaluations. Zimmermann et al. [50] introduced the ImageNet Mechanistic Interpretability (IMI) dataset, containing per-unit interpretability annotations from humans for 767 units, meant to foster research on automating interpretability evaluations.

3 Method

We now introduce our fully automated interpretability measure, Machine Interpretability Score (MIS), visualized in Fig. 1. Borowski et al. [6] proposed a psychophysical experiment for quantifying the per-unit interpretability of vision models, i.e., how well humans can infer the sensitivity of a unit in a vision model from visual explanations. Here, a unit can be a channel in a CNN, commonly averaged over space, a neuron in an MLP, or arbitrary linear combinations of different units. The experiment uses a 2-AFC task design (see Fig. 1A) to measure how well humans understand a unit by probing how well they can predict which of two extremely activating (query) images yields a higher activation, after seeing visual explanations. Specifically, two sets of explanations are displayed: highly and weakly activating images, called positive and negative explanations, respectively. See Appx. A.1 for a more detailed task description. We build on top of this paradigm but replace human participants with machines, resulting in a fully automated interpretability metric that requires no humans.

Definition of the Machine Interpretability Score Let $\mathcal{I}$ denote the space of valid input images for a model. For a specific explanation method and a unit in question, we denote the unit's positive and negative visual explanations as sets of images $\mathcal{E}^+ \subseteq \mathcal{I}$ and $\mathcal{E}^- \subseteq \mathcal{I}$, respectively. Further, let $\mathcal{Q}^+ \subseteq \mathcal{I}$ and $\mathcal{Q}^- \subseteq \mathcal{I}$ be the sets of query images with the most extreme (positive and negative) activations. The task by Borowski et al. [6] can now be expressed as: Given explanations $\mathcal{E}^+$ and $\mathcal{E}^-$ and two queries $\mathbf{q}^+ \in \mathcal{Q}^+$ and $\mathbf{q}^- \in \mathcal{Q}^-$, which of the two queries matches $\mathcal{E}^+$ and which $\mathcal{E}^-$ more closely? An intuitive way to solve this binary decision task is to compare each query with every explanation and match the query images to the sets of explanations based on the images' similarities.

To formalize this, we introduce a perceptual (image) similarity function $f : \mathcal{I} \times \mathcal{I} \to \mathbb{R}$ computing the scalar similarity of two images [47], and an aggregation function $a : \mathbb{R}^K \to \mathbb{R}$ reducing a set of K similarities to a single one. This allows us to define the function $s : \mathcal{I} \times \mathcal{I}^K \to \mathbb{R}$ that quantifies the similarity of a single query image to a set of explanations:

$$s(\mathbf{q}, \mathcal{E}) := a\left(\{\, f(\mathbf{q}, \mathbf{e}) \mid \mathbf{e} \in \mathcal{E} \,\}\right). \tag{1}$$

To decide whether a single query image is more likely to be the positive one, we can compute whether it is more similar to the positive than the negative explanations. We can compute this now for both the positive and the negative query images and get:

$$\Delta_+(\mathbf{q}^+, \mathcal{E}^+, \mathcal{E}^-) = s(\mathbf{q}^+, \mathcal{E}^+) - s(\mathbf{q}^+, \mathcal{E}^-), \tag{2}$$

$$\Delta_-(\mathbf{q}^-, \mathcal{E}^+, \mathcal{E}^-) = s(\mathbf{q}^-, \mathcal{E}^+) - s(\mathbf{q}^-, \mathcal{E}^-). \tag{3}$$

The classification problem will be solved correctly if the similarity of $\mathbf{q}^+$ to $\mathcal{E}^+$ relative to $\mathcal{E}^-$ is stronger than those of $\mathbf{q}^-$. This means we can define the probability of solving the binary classification problem correctly as

$$p(\mathbf{q}^+, \mathbf{q}^-, \mathcal{E}^+, \mathcal{E}^-) := \sigma\Big(\alpha \cdot \big(\Delta_+(\mathbf{q}^+, \mathcal{E}^+, \mathcal{E}^-) - \Delta_-(\mathbf{q}^-, \mathcal{E}^+, \mathcal{E}^-)\big)\Big), \tag{4}$$

where σ denotes the sigmoid function and α is a free parameter to calibrate the classifier's confidence.

We define the *Machine Interpretability Score* (MIS) as the predicted probability of making the right choice, averaged over N tasks for the same unit. Across these different tasks, the query images $\mathbf{q}^+, \mathbf{q}^-$ vary to cover a wider range of the unit's behavior. If the explanation method used is stochastic, it is advisable to also average over different explanations:

$$\text{MIS} = \frac{1}{N} \sum_{i}^{N} p(\mathbf{q}_i^+, \mathbf{q}_i^-, \mathcal{E}_i^+, \mathcal{E}_i^-). \tag{5}$$

Note that the MIS is not a general property of a unit but depends on the explanation method used. A general score can be defined by aggregating the MIS over multiple explanation methods.

Choice of Hyperparameters. We use the current state-of-the-art perceptual similarity, DreamSim [14], as f. See Appx. C for a sensitivity study on this choice. DreamSim models the perceptual similarity of two images as the cosine similarity of the images' representations from (multiple) computer vision backbones. These were first pre-trained with, e.g., CLIP-style training [38] and then fine-tuned to match human annotations for image similarities of pairs of images. We use the mean to aggregate the distances between a query image and multiple explanations to a single scalar, i.e., $a(x_1, \ldots, x_K) := 1/K \sum_i^K x_i$. To choose α, we use the interpretability annotations of IMI [50]: We optimize α over a randomly chosen subset of just 5% of the annotated units to approximately match the value range of human interpretability scores, resulting in $\alpha = 0.16$. Note that α is, in fact, the only free parameter of our metric, resulting in very low chances of overfitting the metric to the IMI dataset. We use the same strategy as Borowski et al. [6], Zimmermann et al. [49] and Zimmermann et al. [50] for generating new tasks (see Appx. A.2). As they used up to 20 tasks per unit, we average over $N = 20$. See Appx. D for a sensitivity study.

4 Results

This section is structured into two parts: First, we validate our Machine Interpretability Score (MIS) by showing that it is well correlated with existing interpretability annotations. Then, we demonstrate what type of experiments become feasible by having access to such an automated interpretability measure. Our experiments use the best-working — according to human judgements [6] — visual explanation method, dataset examples, for computing the MIS. We demonstrate the applicability of our method to other interpretability methods (e.g., feature visualizations) in Appx. E. Note that different explanation methods might require different hyperparameters for computing the MIS. Both query images and explanations are chosen from the training set of ImageNet-2012 [40]. When investigating layers whose feature maps have spatial dimensions, we consider the spatial mean over a channel as one unit [e.g., 6]. We ignore units with constant activations from our analysis as there is no behavior to understand (see Appx. F for details). The code for all experiments is included in the supplementary material and will be publicly released.

4.1 Validating the Machine Interpretability Score

We validate our MIS measure by using the interpretability annotations in the IMI dataset [50], which will be referred to as Human Interpretability Scores (HIS). The per-unit annotations are responses to the 2-AFC task described in Sec. 3, averaged over ≈ 30 participants. IMI contains scores for a subset of units for nine models.[1]

4.1.1 MIS Explains Existing Data

First, we reproduce the main result of Zimmermann et al. [50]: A comparison of nine models in terms of their per-unit interpretability. We plot the HIS and MIS values (averaged over all units in a

[1]Two models were tested in multiple settings, resulting in 14 distinct experimental conditions to compare.

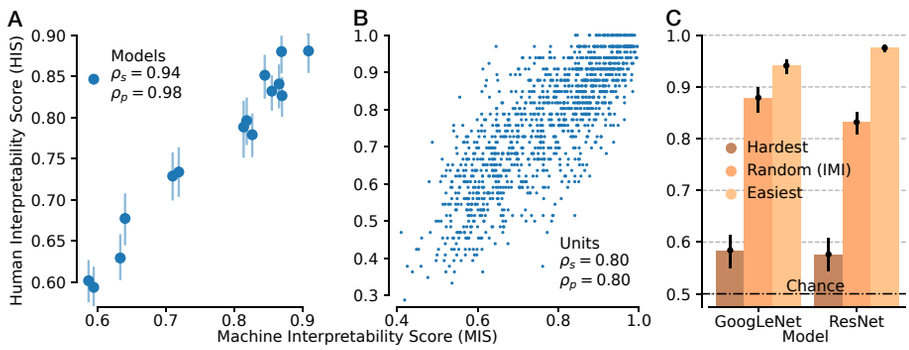


Fig. 2: **Validation of the MIS.** Our proposed Machine Interpretability Score (MIS) explains existing interpretability annotations (Human Interpretability Score, HIS) from IMI [50] well. **(A) MIS Explains Interpretability Model Rankings.** The MIS reproduces the ranking of models presented in IMI while being fully automated and not requiring any human labor, as evident by the strong correlation between MIS and HIS. Similar results are found for the interpretability afforded by another explanation method in Appx. E. **(B) MIS Explains Per-unit Interpretability Annotations.** The MIS also explains individual per-unit interpretability annotations. We show the calculated MIS and the recorded HIS for every unit in IMI and find a high correlation matching the noise ceiling at $\rho = 0.80$ (see Appx. C). **(C) MIS Allows Detection of (Non-) Interpretable Units.** We use the MIS to perform a causal intervention and determine the least (*hardest*) and most (*easiest*) interpretable units in a GoogLeNet and ResNet-50. Using the psychophysics setup of Zimmermann et al. [50], we measure their interpretability and compare them to randomly sampled units. Strikingly, the psychophysics results match the predicted properties: Units with the lowest MIS have significantly lower interpretability than random units, which have significantly lower interpretability than those with the highest MIS. Errorbars denote the 95 % confidence interval.

model) in Fig. 2A and find very strong correlations (Pearson's $r = 0.98$ and Spearman's $r = 0.94$). Reproducing the model ranking is strong evidence for the validity of the metric, as no information about these rankings was explicitly used to create our new measure.

Next, we can zoom in and look at individual units instead of per-model averages. Fig. 2B shows MIS and HIS for all units of IMI. It clearly shows a strong correlation (Pearson's and Spearman's $\rho_s = \rho_p = 0.80$). The interpretability scores in IMI are a (potentially noisy) estimate over a finite number of annotators. We estimate the ceiling performance due to noise (sampling 30 trials from a Bernoulli distribution) to equal Pearson's $\rho_p = 0.82$ (see Appx. C for details). We can conclude that the MIS explains existing interpretability annotations well - both on a per-unit and on a per-model level.

4.1.2 MIS Makes Novel Predictions

While the previous results show a strong relation between MIS and human-perceived interpretability, they are descriptive (correlational). To further test the match between MIS and HIS, we now turn to a causal (interventional) experiment: Instead of predicting the interpretability of units *after* a psychophysics evaluation produced their human scores, we now compute the MIS *before* conducting the psychophysics evaluation. We perform our experiment for two models: GoogLeNet and a ResNet-50. For each model, IMI contains interpretability scores for 96 randomly chosen units. We look at all the units not tested so far and find the 42 units yielding the highest (Easiest, average of 0.99 for both models) and lowest (Hardest, average of 0.63 and 0.59, respectively) MIS, respectively. Then, we use the same setup as Zimmermann et al. [50] and perform a psychophysical evaluation on Amazon Mechanical Turk with 236 participants (Appx. B). We compare the HIS for the random units from the IMI dataset and the two newly recorded groups (easy, hard) of units in Fig. 2C. The results are very clear again: As predicted by the MIS, the HIS is highest for the easiest and lowest for the hardest units. Further, the HIS is close to the *a priori* determined MIS given above. On this newly collected data, we again find a high correlation between MIS and HIS (Pearson's $\rho_p = 0.85$,

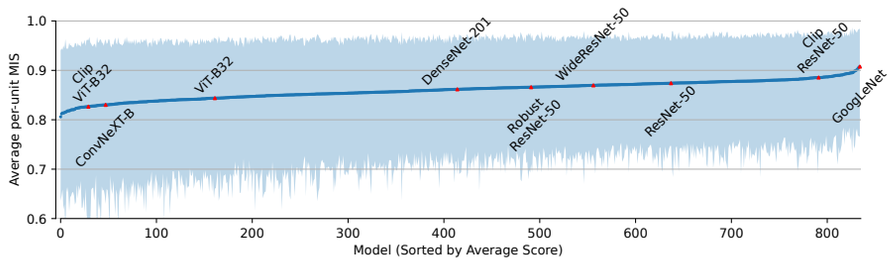


Fig. 3: **Comparison of the Average Per-unit MIS for Models.** We substantially extend the analysis of Zimmermann et al. [50] from a noisy average over a few units for a few models to all units of 835 models. The models are compared regarding their average per-unit interpretability (as judged by MIS); the shaded area depicts the 5th to 95th percentile over units. We see that all models fall into an intermediate performance regime, with stronger changes in interpretability at the tails of the model ranking. Models probed by Zimmermann et al. [50] are highlighted in red.

Spearman's $\rho_s = 0.81$)). This demonstrates the strong predictive power of the MIS and its ability to be used for formulating novel hypotheses.

4.2 Analyzing & Comparing Hundreds of Models

After confirming the validity of the MIS, we now change gears and show use cases for it, i.e., analyses that were truly infeasible before due to the high cost of human evaluations required for measuring the per-unit interpretability. These costs prevented fine-grained analyses. Crucially, our understanding of what influences a unit's interpretability is still fairly limited. For example, it is unclear whether units of specific layer types are more interpretable, or whether a layer's position or width influences its units interpretability. Equipped with the proposed MIS we can now investigate these relations.

4.2.1 Comparison of Models

Zimmermann et al. [50] investigated whether model or training design choices influence the interpretability of vision models. Although they invested a considerable amount of money in this investigation ($\geq 12\,000$ USD), they could only compare nine models via a subset of units. We now scale up this line of work by two orders of magnitude and investigate all units of 835 models, almost all of which come from the well-established computer vision library timm [44]. These models differ in architecture and training datasets but were all at least fine-tuned on ImageNet. See Appx. J for a list of models. Putting this scale into perspective, achieving the same scale by scaling up previous human psychophysics experiments would amount to the absurd costs of more than one billion USD. Following previous work we ignore the first and last layers of each model [50].

When sorting the models according to their average MIS (Fig. 3), they span a value range of $\approx 0.80 - 0.91$. The strongest differences across models are present at the tails of the ranking. Note that GoogLeNet is ranked as the most interpretable model, resonating with the community's interest in GoogLeNet as it is widely claimed to be more interpretable. The shaded area denotes the 5th to 95th percentile of the distribution across units. This reveals a strong difference in the variability of units for different models; further, as the upper end of the MIS is similar across models ($\approx 95\,\%$), most of the change in the average score seems to stem from a change in the lower end, with decreasing width of the per-unit distribution for higher model rank. Note that the MIS cannot only be computed for the most extremely activating query images (see Sec. 3) but also for less activating ones. Refer to Fig. 21 for a version of Fig. 3 that uses the 2nd/98th percentile instead of the most extremely activating query images.

To investigate the difference in how the MIS of units is distributed between different models, we select 15 exemplary models and visualize their per-unit MIS distribution in Fig. 4B. Those models were chosen according to the distance between 5th and 95th percentile (five with highest, average, and lowest distance). While models with low and medium variability have unimodal left-skewed distributions, the ones with high variability have a rather bimodal distribution. Note that the distribu-

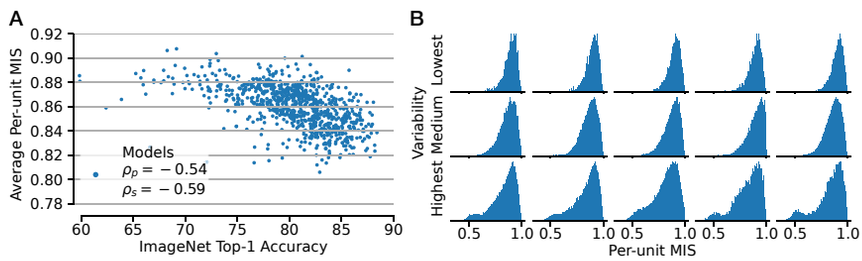


Fig. 4: **(A) Relation Between ImageNet Accuracy and MIS.** The average per-unit MIS of a model is anticorrelated with its ImageNet classification accuracy. Refer to Tab. 2 for a list of the Pareto-optimal models. **(B) Distribution of per-unit MIS.** Distribution of the per-unit MIS for 15 models, chosen based on the size of the error bar in Fig. 3: lowest (top row), medium (middle row), and highest variability (bottom row). While most models have an unimodal distribution, those with high variability have a second mode with lower MIS.

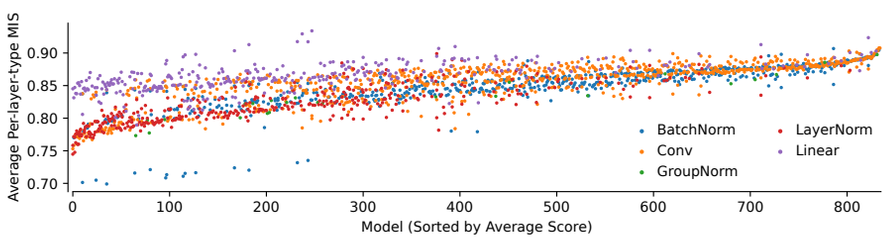


Fig. 5: **Comparison of the Average Per-unit MIS for Different Layer Types and Models.** We show the average interpretability of units from the most common layer types in vision models (BatchNorm, Conv, GroupNorm, LayerNorm, Linear). We follow Zimmermann et al. [50] and restrict our analysis of Vision Transformers to the linear layers in each attention head. While not every layer type is used by every model, we still see some separation between types (see Fig. 18 for significance results): Linear and convolutional layers mostly outperform normalization layers. Models are sorted by average per-unit interpretability, as in Fig. 3.

tion's second, stronger mode has a similar mean and shape to the overall distribution for models with low variability. The first mode is placed at a value range slightly above 0.5, close to the task's chance level, indicating mostly uninterpretable units. This suggests that a subset of uninterpretable units (see Fig. 28 for examples) can explain most of the models' differences in average MIS. We analyze this further in Fig. 22, where we compare the models in terms of their worst units. We see a similar shape as in Fig. 3, but with a larger value range used, resulting in stronger model differences.

Previous work analyzed a potential correlation between interpretability and downstream classification performance. However, in a limited evaluation, it was found that better classifiers are not necessarily more interpretable [50]. A re-evaluation of this question is performed in Fig. 4A and paints an even darker picture: Here, better performing ImageNet classifiers are less interpretable (Pearson's $r = -0.5$ and Spearman's $r = -0.55$). A similar analysis investigating the influence of a model's input resolution on its interpretability suggests no influence (see Fig. 19).

Besides analyzing the interpretability of models, one can also use the MIS to analyze interpretability tools. Above, we directly looked at the interpretability of a model's activations; however, recent work proposed leveraging sparse auto-encoders (SAE) to first transform a model's activations into a potentially more interpretable basis before analyzing it [e.g., 7]. While their application has been mostly limited to language models (with the exception of [23]), we now apply them to vision models in a first exploratory analysis: In Appx. I, we use the MIS to compare the interpretability of a model's original layer and of two competing SAE variants [39, 7] and find no systematic difference.

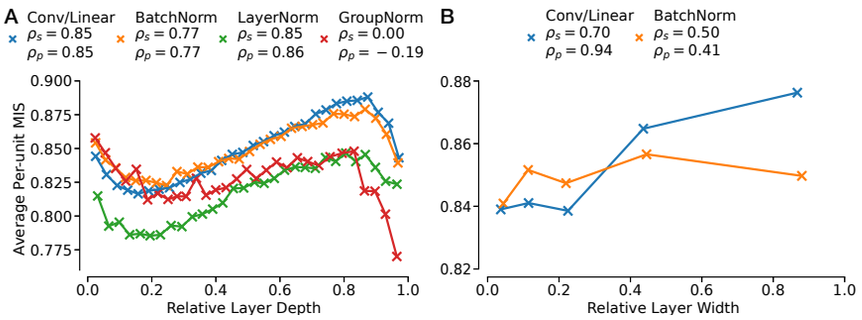


Fig. 6: **(A) Deeper Layers are More Interpretable.** Average MIS per layer as a function of the relative depth of the layer within the network, grouped by layer types. For each type, the values are grouped into 30 bins of equal count based on the relative depth. The crosses depict the bin averages (correlations are calculated for those, too); for a visualization including the bins' variance see Fig. 23. **(B) Wider Layers are More Interpretable.** Average MIS per layer as a function of their relative width, grouped by layer types. The values are grouped into 5 bins. See Fig. 24 for visualizations of how the median, 5th, or 95th percentile of MIS depend on the layer width.

4.2.2 Comparison of Layers

Next, we zoom into the results of Fig. 3 and investigate potential differences between layers. First, we are interested in testing whether the layer type is important, e.g., are convolutional more interpretable than normalization or linear layers? In Fig. 5, we sort the models by their average MIS over all layer types but show individual points for each of the five most common types (Conv, Linear, BatchNorm, LayerNorm, and GroupNorm). The number of points per model may vary, as not all models contain layers of all types. The figure shows a benefit of Conv over BatchNorm layers, which themselves are better than LayerNorm layers. Linear layers, if present, outperform both Batch- and LayerNorm as well as Conv layers. While the differences are small, they are statistically significant due to the large number of scores collected (see Fig. 18).

Second, we analyze whether the location of a layer inside a model plays a role, e.g., are earlier layers more interpretable than later ones? The average per-unit MIS (for each layer type) is shown in Fig. 6A as a function of the relative depth of the layer. A value of zero corresponds to the first and a value of one to the last layer analyzed. The scores are averaged in bins of equal count defined by the relative layer depth to enhance readability. The resulting curves all follow a similar pattern: They start high, decrease in the first fifth, then increase steadily until they drop in the last tenth again, resulting in an almost sinusoidal shape.

Third, it is interesting to probe the influence of the width of layers on their average interpretability. Based on the superposition hypothesis [12, 35, 1, 16], one might expect wider layers to be more interpretable as features do not have to form in superposition (i.e., as *polysemantic* units) but can arise in a disentangled form (i.e., as *monosemantic* units). Fig. 6B shows the relation between MIS and relative layer width. We use the relative rather than the absolute width to reduce the influence of the overall model and show the results of models with different architectures on the same axis. Note that, nevertheless, there might be other confounding factors correlated with the width, e.g., the layer depth. While we only see a weak correlation for BatchNorm layers, we find a stronger one for Conv/Linear layers. It is unclear what causes this difference in behavior. However, we see this as a hint that one way to increase a model's interpretability is to increase the width (and not the number) of layers.

4.3 How Does the MIS Change During Training?

In the last set of experiments, we demonstrate how the MIS can be used to analyze models in a fine-grained way and obtain insights into their training dynamics. For this, we train a ResNet-50 on ImageNet-2012, following the training recipe A3 of Wightman et al. [45], for 100 epochs.

Fig. 7 shows how the average per-unit MIS (left) changes during the training. Notably, the initial MIS (of the untrained network) is already above chance level. Visual explanations (see supplementary

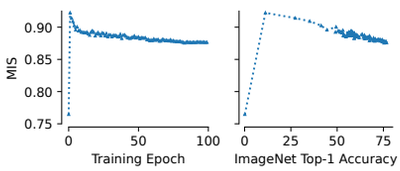


Figure 7: **Interpretability During Training.** For a ResNet-50 trained for 100 epochs on ImageNet, we track the MIS and accuracy after every epoch (epoch 0 refers to initialization). While the MIS improves drastically in the first epoch, it decays during the rest of the training (left). This results in an antiproportional relation between MIS and accuracy (right).

material) indicate a high color dependence of this network's units. However, during the first epoch, the MIS still increases drastically to values around 0.93, before it decays over the rest of the training. This indicates non-trivial dynamics of feature learning, which we analyze in Fig. 8. When showing the MIS as a function of ImageNet accuracy during training (right), a strong anticorrelation (ignoring the first points) becomes evident. This aligns with the anticorrelation shown in Fig. 4A. While we do not have a definite answer for why this is happening, we hypothesize the following: This could be a sign of learning dynamics and the order in which features are learned. After initialization, the network can improve the fastest by learning very simple feature detectors (e.g., colors, simple geometric shapes), as those are weakly correlated with certain classes (e.g., blue colors increase the chance of seeing a fish). Those features are easy for humans to understand. Throughout the training, these feature detectors are replaced with more complex ones that are harder to decode. Fig. 25 the least/most activating dataset examples for units with a strong MIS drop between the second and last training epoch, matching our hypothesis. To better understand the dynamics through the training

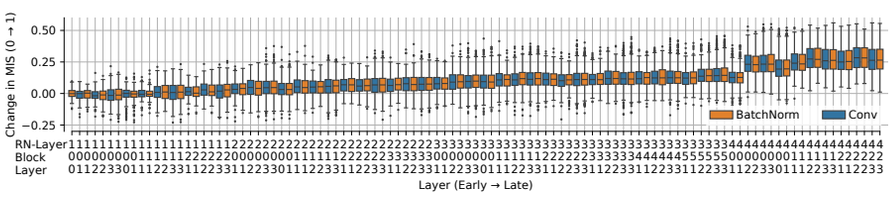


Fig. 8: **Change of Interpretability per Layer During Training.** To better understand the peak in interpretability after the first training epoch found in Fig. 7, we display the change in MIS during the first epoch, averaged over each layer. Layers are sorted by depth from left to right, and different colors encode different layer types. The change in interpretability appears moderately correlated with a layer's depth, such that deeper layers improve the strongest, whereas early layers show no improvement. For an extended visualization covering the full training, see Fig. 20.

— most importantly during the first epoch — we zoom in to find out which units cause this strong change in MIS. Fig. 8 shows the change in MIS during the first epoch for each layer separately (ordered by their depth within the network). We detect a trend of later layers improving more strongly than earlier ones: The change in MIS is heavily driven by the later layers in the network, whose MIS increases strongly while early laters show no improvement at first. In general, we do not see a difference between Conv and BatchNorm layers.

5 Conclusion

This paper presented the first fully automated intepretability metric for vision models: the machine interpretability score (MIS). We verified its alignment to human interpretability score (HIS) through both correlational and interventional experiments. We expect our MIS to enable experiments previously considered infeasible due to the costly reliance on human evaluations. To stress this, we demonstrated the metric's usefulness for formulating and testing new hypotheses about a network's behavior through a series of experiments: Based on the largest comparison of vision models in terms of their per-unit interpretability so far, we investigated potential influences on their interpretability, such as layer depth and width. Most importantly, we find an anticorrelation between a model's downstream performance and its per-unit interpretability. Further, we performed the first detailed analysis of how the interpretability changes during training.

While this paper considerably advances the state of interpretability evaluations, there are some open questions and potential future research directions. Most importantly, the performance of our MIS on a per-unit level is close to the noise ceiling determined by the limited number of human interpretability annotations available. This means that future changes in the MIS measure (e.g., based on other image perceptual similarities) might require additional human labels to determine the significance of performance improvements. Additional human labels could also be leveraged to improve the MIS by following Fu et al. [14] to fine-tune the image similarity directly on human judgments. In another direction, using vision language models for computing the MIS could be interesting as this might, in addition to a numerical score, also provide a textual description of a unit's sensitivity [18]. Finding a differentiable approximation of the MIS will be valuable for explicitly training models to be interpretable [50]. Note that while this paper looked at the interpretability of channels and neurons, it can also be used to analyze arbitrary directions in activation space. Thus, we expect the MIS to also be valuable for researchers generally looking for more interpretable representations of (artificial) neural activations [e.g., 17]. Finally, exploring whether this concept of interpretability quantification can be expanded to LLMs is an exciting direction.

Author Contributions

RSZ led the project, which DK initiated. DK proposed using perceptual similarity functions to build an interoperability metric. RSZ and WB conceived the final formulation of the metric. RSZ conducted all the experiments with suggestions from WB and feedback from DK. RSZ executed the data analysis, except for the estimation of the noise ceiling conducted by DK. RSZ created all the figures in the paper and wrote the manuscript with suggestions from DK and WB.

Acknowledgments

This work was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A. WB acknowledges financial support via an Emmy Noether Grant funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1 and via the Open Philantropy Foundation funded by the Good Ventures Foundation. WB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. This research utilized compute resources at the Tübingen Machine Learning Cloud, DFG FKZ INST 37/1057-1 FUGG. The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting RSZ.

References

[1] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. Linear Algebraic Structure of Word Senses, with Applications to Polysemy, December 2018. Cited on page 8.

[2] Horace Barlow. Single units and sensation: A neuron doctrine for perceptual psychology? *Perception*, 1:371–94, 02 1972. doi: 10.1068/p010371. Cited on page 2.

[3] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. Cited on page 2.

[4] David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, September 2020. doi: 10.1073/pnas.1907375117. URL `https://doi.org/10.1073/pnas.1907375117`. Cited on page 2.

[5] Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. Language models can explain neurons in language models. `https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html`, 2023. Cited on page 3.

[6] Judy Borowski, Roland S. Zimmermann, Judith Schepers, Robert Geirhos, Thomas S. A. Wallis, Matthias Bethge, and Wieland Brendel. Exemplary natural images explain cnn activations better than state-of-the-art feature visualization. In *Ninth International Conference on Learning Representations (ICLR 2021)*, 2021. Cited on pages 2, 3, 4, 15, 16, and 17.

[7] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html. Cited on pages 1, 2, 7, 20, and 21.

[8] Nick Cammarata, Shan Carter, Gabriel Goh, Chris Olah, Michael Petrov, Ludwig Schubert, Chelsea Voss, Ben Egan, and Swee Kiat Lim. Thread: Circuits. *Distill*, 2020. doi: 10.23915/distill.00024. https://distill.pub/2020/circuits. Cited on page 1.

[9] Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards Automated Circuit Discovery for Mechanistic Interpretability, October 2023. Cited on page 3.

[10] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli. Image Quality Assessment: Unifying Structure and Texture Similarity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2567–2581, May 2022. ISSN 1939-3539. doi: 10.1109/TPAMI.2020.3045810. Cited on page 17.

[11] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html. Cited on page 2.

[12] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition. *Transformer Circuits Thread*, 2022. Cited on pages 1 and 8.

[13] Dumitru Erhan, Y. Bengio, Aaron Courville, and Pascal Vincent. Visualizing higher-layer features of a deep network. *Technical Report, Univeristé de Montréal*, 01 2009. Cited on page 2.

[14] Stephanie Fu, Netanel Tamir, Shobhita Sundaram, Lucy Chai, Richard Zhang, Tali Dekel, and Phillip Isola. DreamSim: Learning New Dimensions of Human Visual Similarity using Synthetic Data, December 2023. Cited on pages 4, 10, and 17.

[15] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael A. Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 80–89, 2018. Cited on page 2.

[16] Gabriel Goh. Decoding the Thought Vector. https://gabgoh.github.io/ThoughtVectors/, 2016. Cited on page 8.

[17] Mara Graziani, Laura O'Mahony, An-phi Nguyen, Henning Müller, and Vincent Andrearczyk. Uncovering unique concept vectors through latent space decomposition. *Transactions on Machine Learning Research*, 2023. Cited on page 10.

[18] Evan Hernandez, Sarah Schwettmann, David Bau, Teona Bagashvili, Antonio Torralba, and Jacob Andreas. Natural Language Descriptions of Deep Visual Features, April 2022. Cited on pages 2, 3, and 10.

[19] Jing Huang, Atticus Geiger, Karel D'Oosterlinck, Zhengxuan Wu, and Christopher Potts. Rigorously assessing natural language explanations of neurons. *arXiv preprint arXiv:2309.10312*, 2023. Cited on page 3.

[20] D H Hubel and T N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physiol.*, 160(1):106–154, January 1962. Cited on page 2.

[21] Neha Kalibhat, Shweta Bhardwaj, C Bayan Bruss, Hamed Firooz, Maziar Sanjabi, and Soheil Feizi. Identifying interpretable subspaces in image representations. In *International Conference on Machine Learning*, pages 15623–15638. PMLR, 2023. Cited on page 2.

[22] Sunnie S. Y. Kim, Nicole Meister, Vikram V. Ramaswamy, Ruth Fong, and Olga Russakovsky. HIVE: Evaluating the human interpretability of visual explanations. In *European Conference on Computer Vision (ECCV)*, 2022. Cited on page 2.

[23] David Klindt, Sophia Sanborn, Francisco Acosta, Frédéric Poitevin, and Nina Miolane. Identifying interpretable visual features in artificial and biological neural systems. *arXiv preprint arXiv:2310.11431*, 2023. Cited on pages 1 and 7.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25: 26th Annual Conference on Neural Information Processing Systems 2012. Proceedings of a Meeting Held December 3-6, 2012, Lake Tahoe, Nevada, United States*, pages 1106–1114, 2012. Cited on pages 1 and 17.

[25] Matthew L. Leavitt and Ari S. Morcos. Towards falsifiable interpretability research. *CoRR*, abs/2010.12016, 2020. URL `https://arxiv.org/abs/2010.12016`. Cited on page 2.

[26] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5188–5196, Los Alamitos, CA, USA, jun 2015. IEEE Computer Society. doi: 10.1109/CVPR.2015.7299155. URL `https://doi.ieeecomputersociety.org/10.1109/CVPR.2015.7299155`. Cited on pages 1 and 2.

[27] Ari S. Morcos, David G.T. Barrett, Neil C. Rabinowitz, and Matthew Botvinick. On the importance of single directions for generalization. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=r1iuQjxCZ`. Cited on page 2.

[28] Alexander Mordvintsev, Chris Olah, and Mike Tyka. Inceptionism: Going deeper into neural networks, 2015. URL `https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html`. Cited on page 2.

[29] Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability, 2023. Cited on page 1.

[30] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 12 2014. Cited on page 2.

[31] Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017. Cited on page 2.

[32] Tuomas Oikarinen and Tsui-Wei Weng. Clip-dissect: Automatic description of neuron representations in deep vision networks. In *The Eleventh International Conference on Learning Representations*, 2022. Cited on page 3.

[33] Chris Olah. Mechanistic interpretability, variables, and the importance of interpretable bases, 2022. URL `https://transformer-circuits.pub/2022/mech-interp-essay/index.html`. Cited on pages 1 and 2.

[34] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. Feature visualization. *Distill*, 2017. doi: 10.23915/distill.00007. https://distill.pub/2017/feature-visualization. Cited on pages 1 and 2.

[35] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020. doi: 10.23915/distill.00024.001. https://distill.pub/2020/circuits/zoom-in. Cited on page 8.

[36] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html. Cited on page 2.

[37] R Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107, 2005. Cited on page 2.

[38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision, February 2021. Cited on page 4.

[39] Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. Improving Dictionary Learning with Gated Sparse Autoencoders, April 2024. Cited on pages 1, 7, 20, and 21.

[40] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. doi: 10.1007/s11263-015-0816-y. Cited on pages 4 and 34.

[41] Sarah Schwettmann, Tamar Rott Shaham, Joanna Materzynska, Neil Chowdhury, Shuang Li, Jacob Andreas, David Bau, and Antonio Torralba. FIND: A Function Description Benchmark for Evaluating Interpretability Methods, December 2023. Cited on page 3.

[42] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. Cited on page 17.

[43] Aaquib Syed, Can Rager, and Arthur Conmy. Attribution Patching Outperforms Automated Circuit Discovery, November 2023. Cited on page 3.

[44] Ross Wightman. Pytorch image models. `https://github.com/rwightman/pytorch-image-models`, 2019. Cited on pages 6 and 21.

[45] Ross Wightman, Hugo Touvron, and Hervé Jégou. Resnet strikes back: An improved training procedure in timm. *arXiv preprint arXiv:2110.00476*, 2021. Cited on page 8.

[46] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. In *Deep Learning Workshop, International Conference on Machine Learning (ICML)*, 2015. Cited on page 2.

[47] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 586–595. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00068. Cited on pages 3 and 17.

[48] Bolei Zhou, Yiyou Sun, David Bau, and Antonio Torralba. Revisiting the importance of individual units in cnns via ablation. *CoRR*, abs/1806.02891, 2018. URL `http://arxiv.org/abs/1806.02891`. Cited on page 2.

[49] Roland S. Zimmermann, Judy Borowski, Robert Geirhos, Matthias Bethge, Thomas Wallis, and Wieland Brendel. How well do feature visualizations support causal understanding of cnn activations? In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 11730–11744. Curran Associates, Inc., 2021. URL `https://proceedings.neurips.cc/paper_files/paper/2021/file/618faa1728eb2ef6e3733645273ab145-Paper.pdf`. Cited on pages 3, 4, 15, and 16.

[50] Roland S. Zimmermann, Thomas Klein, and Wieland Brendel. Scale alone does not improve mechanistic interpretability in vision models. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL `https://openreview.net/forum?id=0Z7aImD4uQ`. Cited on pages 1, 2, 3, 4, 5, 6, 7, 10, 15, 16, 21, and 34.

A Description of the 2-AFC Task

A.1 Task Design

Our proposed MIS builds on the 2-AFC task designed by Borowski et al. [6] to conduct human psychophysics experiments. An example of such a task is given in Fig. 9.

This task aims to probe how well (human) participants can detect the sensitivity of a unit of a neural network based on visual explanations of it. Understanding the unit's sensitivity should allow participants to distinguish between a stimulus eliciting high from one yielding low activation. Therefore, the task shows the participants two such images, called query images, and asks them to pick the image eliciting higher activation. To solve the task, participants also see two sets of visual explanations: Positive explanations describe the patterns the unit activates strongly for, while negative activations show patterns the unit weakly responds to. For solving this task, there are two potential strategies: Participants can either recognize a common pattern of the positive explanations in one of the query images, making this the correct choice. Or they detect a common pattern of the negative explanations in a query image, making the other one the right choice. See Borowski et al. [6], Zimmermann et al. [49] or Zimmermann et al. [50] for alternative descriptions and visualizations of the task.

Fig. 9: **Examples of the 2-AFC Task.** For two different units of GoogLeNet one task each is shown. Every task contains a set of negative (left) and positive (right) visual explanations describing which visual feature the unit is sensitive to. In the center, two query images in the form of strongly and weakly activating dataset examples are shown, respectively. This means that each one of the two query images corresponds to the positive and the other to the negative explanations. The task is now to choose which query image corresponds to the positive ones.

A.2 Task Construction

For constructing tasks, we follow Zimmermann et al. [50]. Specifically, this means that we use $K = 9$ (positive and negative) explanations in each task. We restrict explanations to natural dataset examples to reduce complexity but note that the same setup can also be applied to other visual explanations, such as feature visualizations. To choose query images and explanations, we proceed as follows: For each unit, we determine the $N \cdot (K + 1)$ most and least activating images, respectively. Out of these, the $N \cdot K$ most extreme images are used as explanations, the others as query images. The $N \cdot K$ potential explanation images are uniformly distributed across tasks according to their elicited activation level (see [6, 50] for more details).

B Psychophysical Evaluation

In Sec. 4.1.2, we validate the correctness of the proposed MIS in terms of estimating the interpretability for new, not previously analyzed, units. As described above, we use the MIS to create a set of the most and least interpretable units (42 units each). We then collect human annotations (i.e., human interpretability scores) for these units by running a psychophysical experiment. Specifically, we let humans solve the 2-AFC task described in Appx. A.1, originally introduced by [6]. We use the open-sourced setup of Zimmermann et al. [50] and recruit participants from Amazon Mechanical Turk. See Fig. 10 for screenshots of the task and the information given to participants on how their anonymized responses will be used. We leverage established quality/attention checks to control for high data quality and restrict the pool of potential participants to experienced workers [49, 50]. In total, we recruit 236 participants for our evaluation. The recruited workers are compensated at a targeted hourly rate of 15 USD, i.e., 2.79 USD per task based on the average task duration reported by Zimmermann et al. [50]. All procedures conform to Standard 8 of the American Psychological 405 Association's "Ethical Principles of Psychologists and Code of Conduct" (2016).

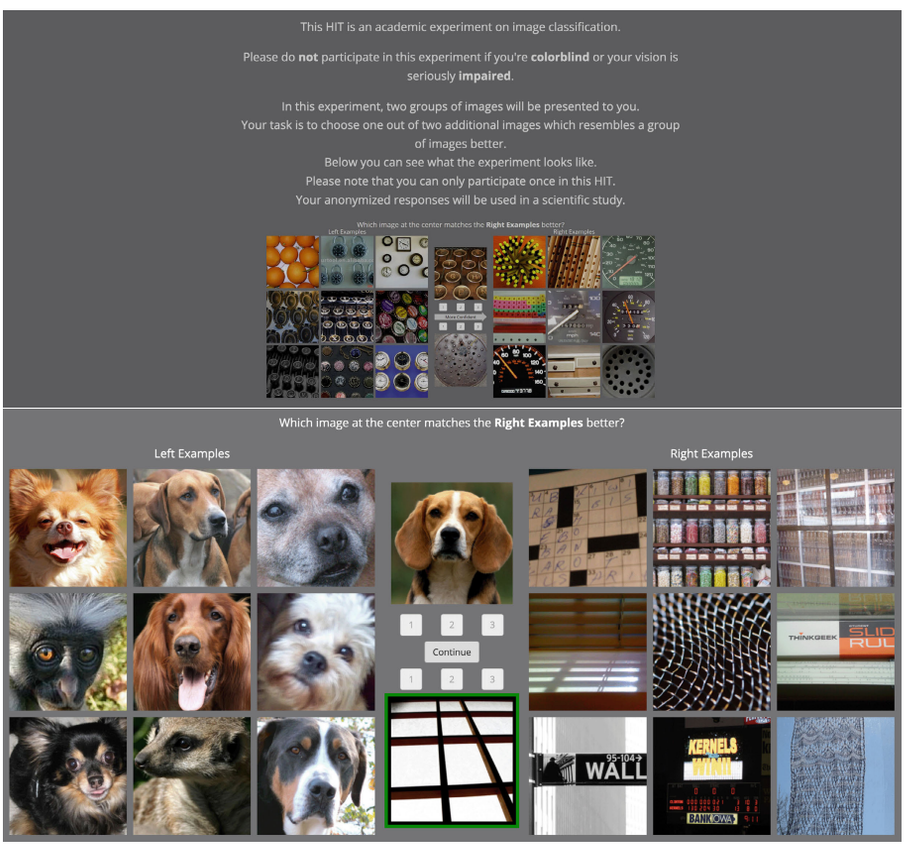


Fig. 10: Screenshot of the initial overview of the HIT presented to workers considering the task. We inform participants that they consent to their anonymized data being used for a scientific study.

C Influence of the Underlying Perceptual Similarity on the Machine Interpretability Score

As stated in Sec. 3, we used DreamSim [14] as the underlying perceptual similarity f for all experiments shown so far. We now repeat the experiments on IMI in Sec. 4.1.1 with two alternative similarity measures: LPIPS [47] and DISTS [10]. While all three measures are based on learned image features, DreamSim leverages an ensemble of modern vision models trained on larger datasets compared to LPIPS and DISTS, which use AlexNet [24] and VGG16 [42] trained on ImageNet, respectively. According to Fu et al. [14], DreamSim clearly outperforms LPIPS and DISTS on image similarity benchmarks.

When comparing MIS based on DreamSim with one based on LPIPS and DISTS on a per-model level (see Fig. 11) one sees very similar results and strong correlations between each MIS and HIS. This might suggest that the choice of the similarity function to use has little influence on the quality of MIS. The picture, however, changes when zooming in and looking at per-unit interpretability (see Fig. 13). Now, it becomes evident that the MIS based on DreamSim outperforms that based on LPIPS and DISTS, indicated by the higher correlation and smaller spread of the point cloud. We, therefore, conclude that DreamSim is the best perceptual similarity available for computing machine interpretability scores.

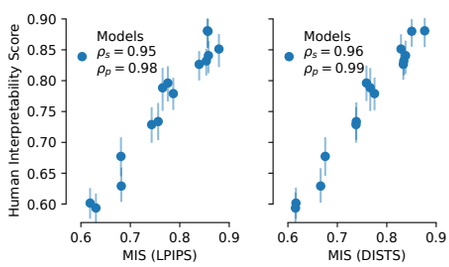


Fig. 11: **LPIPS and DISTS Perform Similarly as DreamSim when Comparing Models.** We compare DreamSim with two earlier perceptual similarity metrics, LPIPS and DISTS. All three lead to similar results on IMI (cf. Fig. 2A). See Fig. 13 for comparing these similarity functions on a per-unit level. standard deviation.

Noise Ceiling of Annotations in IMI To put the difference in performance between the perceptual similarities on a per-unit level into context, we estimate the noise ceiling of the data: As the HIS for a single unit is a (potentially) noisy estimate over (up to 30) human decisions, it has some uncertainty. To account for this, we run a statistical simulation in which we model individual human responses as binary decisions from a Bernoulli distribution whose mean equals the unit's HIS. We can now simulate human decisions by sampling from the distribution. Then, we compute the correlation between MIS and simulated HIS and repeat the process 1 000 times. The resulting *noise ceiling* is compared to the correlations obtained when using LPIPS, DISTS, and DreamSim in Fig. 12. DreamSim's performance is very close to the noise ceiling for estimating the per-unit human interpretability.

D Sensitivity of the MIS on the Number of Tasks

As described in Sec. 3, we compute the MIS by averaging over $N = 20$ tasks. This choice was initially motivated by previous work by Borowski et al. [6]. We investigate now how this choice influences the MIS. For this, we perform two experiments for GoogLeNet (see Fig. 14). First, we use the method for constructing tasks described before in Appx. A.2 to create 20 tasks per unit and then compute how the MIS changes when only using the first $i = 1, \ldots, 19$ tasks compared to all 20. While this setting is straightforward to analyze, it does not reflect how the number of tasks influences the MIS computation in practice: Using the task creation above, the chosen number of tasks influences the creation of all tasks, e.g., adding one more task changes which images are used for previous tasks. Therefore, in the second experiment, we again measure how the MIS changes

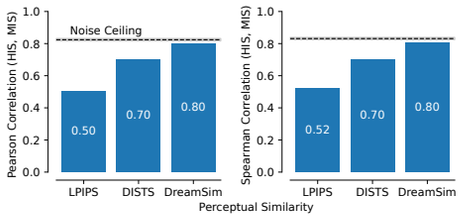


Fig. 12: **Best Perceptual Similarity Approaches Noise Ceiling.** Considering the noise ceiling, caused by the inherent uncertainty of the HIS, the best perceptual similarity (DreamSim) shows an almost perfect performance. The black bar and shaded area show the mean correlation and standard deviation over 1 000 simulations, respectively.

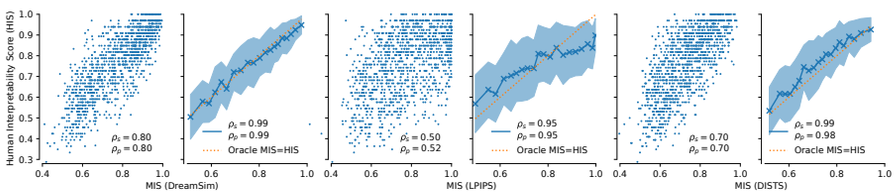
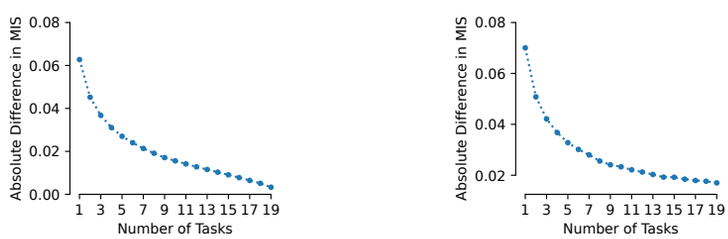


Fig. 13: **LPIPS and DISTS Perform Worse than DreamSim when Comparing Individual Units.** We compare DreamSim with two earlier perceptual similarity metrics, LPIPS and DISTS. While LPIPS and DISTS perform similarly to DreamSim on a per-model level of IMI (cf. Fig. 13), they lead to worse performance on a per-unit level.

when using $i = 1, \ldots, 19$ tasks compared to 20, but recreate all tasks when increasing their number. For both settings, we see that the residual converges to zero, with a slower convergence in the more realistic setting.



(a) New tasks do not influence earlier tasks.

(b) New tasks influence earlier tasks.

Fig. 14: **Convergence of MIS.** We investigate how MIS changes depending on the number of tasks N that it is computed over. Here, we distinguish between two settings. In (a), we simulate that adding another task does not change the selection of query images and explanations in earlier tasks; in (b), this is not the case. While the former is easier to analyze due to a reduced level of randomness, note that the latter is the more relevant setting in practice. For both cases, we visualize the average absolute difference in MIS estimated for < 20 and $N = 20$ tasks.

E Applying MIS for Different Explanation Methods

The experiments in Sec. 4 compute the MIS for one type of explanation, namely strongly activating dataset examples. We now demonstrate that the same approach easily generalizes to other visual explanations: feature visualizations. We do not tune any hyperparameters but re-use the same as presented in Sec. 3 for dataset examples as explanations. In Fig. 15 we repeat the experiment from Fig. 2A and again see a strong correlation between MIS and HIS.

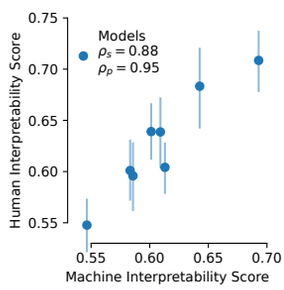


Fig. 15: **MIS Generalizes Well to Other Explanation Types.** We find a high correlation between MIS and HIS for other explanation types (feature visualizations). See Fig. 2A for the corresponding results for using natural dataset examples as explanations.

F Analysis of Constant Units

After training a network, it might happen that some of its units effectively become non-active/constant for any relevant image. We here call a unit *constant* if the difference between maximally and minimally elicited activation by the entire ImageNet-2012 training set is less than 10^{-8}. As mentioned at the beginning of Sec. 4, we excluded those units in our analysis, as they do not present any interesting behavior that is worth understanding. Note that this does not mean that it will not be interesting to understand why such units exist. In Fig. 16, we display the ratio of constant units for each model. For most models, we see a low number of constant units: Specifically, we see that out of the 835 models investigated, 256 do not contain any constant units, 89 contain more than $1\,\%$ and 22 more than $5\,\%$. Note that we here used the same notion of units as in the rest of the paper, meaning that we take the spatial mean of feature maps with spatial dimensions (e.g., for convolutional layers).

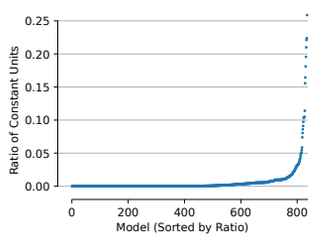


Fig. 16: **Ratio of Constant Units.** We compute the ratio of units constant with respect to the input (over the training set of ImageNet-2012) for all models considered. While the ratio is low for most models, it becomes large for a few models.

G Computational Resources

Complexity of MIS Computing the MIS of a unit consists of four steps: (1) determining its visual explanations, (2) finding the strongly and weakly activating dataset samples to be used as the query images of the 2-AFC task, (3) computing the pairwise image similarities, and (4) computing the

final MIS. Due to the simplicity of the MIS' computation, its cost is neglectable. The complexity of the first step depends on the visualization method used: Gradient-based search algorithms, e.g., feature visualizations, require hundreds of forward and backward passes (of small batches), while determining dataset examples requires only a single forward pass over a sufficiently large dataset. The second step also mostly requires a single forward pass over this large dataset. Thus, if dataset examples are used as explanations, this step is free. Performing the third step requires computing the pairwise similarities of the images used in the created tasks. However, as most perceptual similarities, most importantly the leveraged DreamSim metric, are computed as the cosine similarity of an image's features, the step can be greatly simplified: We first compute and store the features for every image in the dataset used to sample the tasks' images. Then, computing the similarities equals only querying two features from a hash map and computing their cosine similarity. While this caching approach is not necessary for computing the MIS of a single unit, it becomes important when computing it for thousands of units. In this case, computing and caching the similarities also becomes neglectable, meaning that the computational cost of the MIS is dominated by the first and second steps. In summary, computing the MIS mostly resorts to a single forward pass over a sufficiently large dataset and additional forward/backward passes only depending on the visualization technique used.

Resources Used Due to the aforementioned low computational complexity of the MIS, the experiments in Sec. 4 do not require much compute: Evaluating all units of a model takes, on average and varying depending on the model's size, less than one hour on a GPU (e.g., NVIDIA RTX 2080-TI or V100). Therefore, reproducing the experimental results of this paper requires approximately 1000 GPU hours.

H Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning, specifically the field of Interpretable Machine Learning. The main contribution of our work is the presentation of a more time- and cost-efficient approach for quantifying how well humans can understand neural activations. A potential risk in automating interpretability research is that we will start optimizing for metrics that are never fully aligned with human judgments. It is conceivable that this will encourage the design of models that ace our metric but whose inner workings and decision-making processes are still obscure to human observers. This would set false goalposts and potentially come with safety risks if a high score in MIS were mistaken for a white box model that comes with higher trustworthiness. Beyond that, we see many potential use cases for this result (see Sec. 5), that can all advance the state of machine learning. There are potential societal consequences of our work, however, none of which we feel must be specifically highlighted here.

I Analyzing SAEs

Sparse Auto-Encoders (SAE) have been recently proposed as a means to understand the behavior of a network's layer better [7]: By finding a new, sparser basis to represent the layer's original activation, one hopes to find new artificial computational units that are more monosemantic. These units are expected to be easier to understand, rendering the tasks of understanding the behavior of the entire layer easier, too. While conceptually simple, the implementation and evaluation of SAEs is intricate: Training them requires careful hyperparameter tuning and algorithmic design choices such that the final SAEs are as sparse as possible but still faithful to the layer's original activations. However, as no reliable automatic interpretability evaluation has existed so far, evaluating SAEs in terms of how much more interpretable their features are is difficult, resulting in potentially inconclusive results. For example, Rajamanoharan et al. [39] suggested a modification to the usual SAE architecture (Gated SAE) but could not find a statistically significant benefit over the default architecture due to the high and, thus, prohibitive cost of interpretability evaluations.

As the MIS enables cheap interpretability evaluations, we can now pick up this work: In the context of vision models, we train different SAEs and Gated SAEs and compare their interpretability. Specifically, we train them on activations of one layer of GoogLeNet (mixed4b_3x3) and use different expansion factors and weights of the sparsity loss to obtain different SAEs. In addition to their interpretability, we also evaluate models in terms of their sparsity (ℓ_0 count) and their reconstruction fidelity, i.e., how well they maintain the original model's classification cross-entropy compared to a

random model. In line with [39], Fig. 17 shows that Gated SAEs allow a better fidelity vs. sparsity trade-off. In terms of their MIS, we do not see a systematic difference between the two architectures. Moreover, in light of the high MIS of the original layer (i.e., $91.21\,\%$), we do not see a strong benefit of SAEs compared to analyzing the original layer yet.

In another experiment, we trained (vanilla) SAEs on another, less interpretable layer (layer2_2_conv2 of a ResNet50). While the units of the original layer have an average MIS of 0.854 we observe that MIS values of up to 0.922 for SAEs with ten times more units than the original layer. See Tab. 1 for a sensitivity study on the relationship of an SAE's sparsity and its MIS.

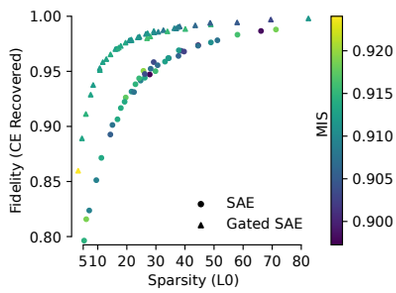


Fig. 17: **Comparable MIS for different SAE architectures**. We compare two types of SAEs used by Bricken et al. [7] and Rajamanoharan et al. [39] (SAE and Gated SAE, respectively), in terms of their sparsity, reconstruction fidelity and interpretability. While Gated SAEs allow a more optimal tradeoff between fidelity and sparsity, they are comparably interpretable as standard SAEs. The SAEs overall MIS is in a similar regime as the original layer's ($91.21\,\%$), while the sparsity is stronger than that of the original layer (≈ 75).

Tab. 17: **Sensitivity of SAE's MIS on its Hyperparameters.**

Sparsity Weight λ $[10^{-2}]$	1.125	2.5	3.75	5.0	6.25	7.5	8.75	10.0
L0 Count	233	138	99	75	60	49	41	35
MIS	0.892	0.908	0.916	0.915	0.919	0.918	0.922	0.918

J Details on Models

In addition to the 9 models investigated by Zimmermann et al. [50] (GoogLeNet, ResNet-50, Clip ResNet-50, Robust (L2) ResNet-50, DenseNet-101, WideResNet-50, Clip ViT-B32. ViT-B32), we include one more model suggested by them (Robust (L2) ResNet-50) and 825 models from timm [44]:

xcit_tiny_12_p16_224.fb_in1k, vit_tiny_patch16_384.augreg_in21k_ft_in1k, pit_xs_224.in1k, repghostnet_111.in1k, regnetz_c16_evos.ch_in1k, poolformer_m48.sail_in1k, repghostnet_080.in1k, volo_d3_448.sail_in1k, vit_base_patch16_224.augreg_in21k_ft_in1k, regnety_320.tv2_in1k, densenet121.ra_in1k, mobilenetv3_large_100.ra_in1k, repghostnet_150.in1k, seresnext26ts.ch_in1k, regnety_160.swag_ft_in1k, hrnet_w40.ms_in1k, convnext_small.in12k_ft_in1k, vit_base_patch16_224.sam_in1k, seresnextaa101d_32x8d.sw_in12k_ft_in1k_288, vit_tiny_r_s16_p8_384.augreg_in21k_ft_in1k, regnety_320.pycls_in1k, cs3darknet_m.c2ns_in1k, vit_tiny_patch16_224.augreg_in21k_ft_in1k, resnet101c.gluon_in1k, convnextv2_atto.fcmae_ft_in1k, flexivit_base.600ep_in1k, xcit_small_12_p16_384.fb_dist_in1k, mobilenetv2_050.lamb_in1k, flexivit_base.300ep_in1k, resnext50_32x4d.tv_in1k, resnet152.tv_in1k, seresnext26d_32x4d.bt_in1k, fbnetv3_g.ra2_in1k, poolformer_s36.sail_in1k, resnext101_32x8d.tv_in1k, rexnet_130.nav_in1k, efficientvit_b2.r224_in1k, convnext_small.fb_in22k_ft_in1k_384, resnet50_gn.a1h_in1k, eva02_small_patch14_336.mim_in22k_ft_in1k, regnety_032.ra_in1k, res2net50d.in1k, convit_small.fb_in1k, regnetx_160.pycls_in1k, convnextv2_large.fcmae_ft_in22k_in1k_384, tf_efficientnet_b0.ns_jft_in1k, pit_ti_224.in1k, volo_d1_384.sail_in1k, xcit_small_12_p8_384.fb_dist_in1k, dpn131.mx_in1k, resnext101_64x4d.gluon_in1k, densenet169.tv_in1k, resnet101d.ra2_in1k, repghostnet_200.in1k, resnet18.a2_in1k, xcit_small_12_p16_224.fb_in1k, pvt_v2_b3.in1k, dm_nfnet_f1.dm_in1k, vit_large_patch32_384.orig_in21k_ft_in1k, convnextv2_tiny.fcmae_ft_in22k_in1k_384, gcresnet50t.ra2_in1k, nf_regnet_b1.ra2_in1k, volo_d1_224.sail_in1k, resnet50.ram_in1k, hrnet_w18_small_v2.ms_in1k, convnext_base.clip_laion2b_augreg_ft_in1k, regnetx_160.tv2_in1k, sequencer2d_l.in1k, convnext_large.fb_in22k_ft_in1k, botnet26t_256.c1_in1k, gc_efficientnetv2_rw_t.agc_in1k, wide_resnet50_2.racm_in1k,

halonet50ts.a1h_in1k, cspresnext50.ra_in1k, resnetv2_50d_evos.ah_in1k, tf_efficientnetv2_b3.in21k_ft_in1k, resnet152.gluon_in1k, lambda_resnet26rpt_256.c1_in1k, fastvit_sa24.apple_dist_in1k, xcit_medium_24_p8_384.fb_dist_in1k, repvit_m0_9.dist_450e_in1k, regnetx_320.pycls_in1k, seresnextaa101d_32x8d.sw_in12k_ft_in1k, efficientvit_b2.r288_in1k, convnext_tiny.in12k_ft_in1k, xcit_large_24_p16_384.fb_dist_in1k, resnetv2_50.a1h_in1k, coatnet_0_rw_224.sw_in1k, efficientnet_es_pruned.in1k, dla60_res2net.in1k, efficientformer_l7.snap_dist_in1k, cait_xxs24_224.fb_dist_in1k, vit_small_patch16_224.augreg_in21k_ft_in1k, tf_efficientnet_cc_b1_8e.in1k, efficientvit_b1.r288_in1k, halonet26t.a1h_in1k, mixnet_m.ft_in1k, hrnet_w44.ms_in1k, regnety_160.tv2_in1k, xcit_nano_12_p8_384.fb_dist_in1k, seresnext101_32x8d.ah_in1k, efficientvit_b2.r256_in1k, vit_base_patch16_clip_224.laion2b_ft_in12k_in1k, tf_efficientnet_lite2.in1k, deit3_small_patch16_224.fb_in1k, hrnet_w18_ssld.paddle_in1k, tf_efficientnet_b2.aa_in1k, crossvit_15_dagger_240.in1k, deit3_small_patch16_224.fb_in22k_ft_in1k, haloregnetz_b.ra3_in1k, tf_efficientnetv2_b0.in1k, eca_nfnet_l0.ra2_in1k, twins_pcpvt_small.in1k, ecaresnet50t.ra2_in1k, fastvit_sa12.apple_dist_in1k, skresnext50_32x4d.ra_in1k, resnet50d.a2_in1k, vit_base_patch32_clip_224.laion2b_ft_in1k, resnetblur50.bt_in1k, vit_base_patch16_224.orig_in21k_ft_in1k, resnet50.a1h_in1k, hardcorenas_e.miil_green_in1k, coatnext_nano_rw_224.sw_in1k, convnext_base.clip_laiona_augreg_ft_in1k_384, tresnet_m.miil_in1k_448, resnet10t.c3_in1k, poolformerv2_m48.sail_in1k, tf_efficientnet_b1.aa_in1k, edgenext_base.usi_in1k, tf_efficientnet_es.in1k, tresnet_l.miil_in1k_448, resnet152.a1h_in1k, mixnet_s.ft_in1k, resnet50.am_in1k, rexnet_100.nav_in1k, xcit_large_24_p8_224.fb_dist_in1k, deit3_base_patch16_224.fb_in22k_ft_in1k, xcit_tiny_24_p8_384.fb_dist_in1k, coat_lite_medium_384.in1k, focalnet_small_srf.ms_in1k, vit_base_patch8_224.augreg_in21k_ft_in1k, convnext_tiny_hnf.a2h_in1k, visformer_small.in1k, vit_small_r26_s32_384.augreg_in21k_ft_in1k, vgg16_bn.tv_in1k, eca_nfnet_l1.ra2_in1k, xcit_small_12_p8_224.fb_in1k, beitv2_base_patch16_224.in1k_ft_in22k_in1k, cs3edgenet_x.c2_in1k, vit_base_patch16_clip_384.laion2b_ft_in12k_in1k, xcit_small_12_p16_224.fb_dist_in1k, convformer_b36.sail_in1k_384, bat_resnext26ts.ch_in1k, caformer_b36.sail_in1k, dla34.in1k, crossvit_18_dagger_240.in1k, tf_efficientnetv2_s.in21k_ft_in1k, focalnet_base_srf.ms_in1k, convformer_b36.sail_in22k_ft_in1k_384, resnet34.tv_in1k, resmlp_24_224.fb_distilled_in1k, convnext_base.clip_laion2b_augreg_ft_in12k_in1k, caformer_s18.sail_in1k_384, resnetaa50.a1h_in1k, beitv2_base_patch16_224.in1k_ft_in1k, convformer_m36.sail_in22k_ft_in1k, inception_resnet_v2.tf_ens_adv_in1k, mobilenetv2_110d.ra_in1k, resnext101_32x4d.fb_swsl_ig1b_ft_in1k, regnetx_008.tv2_in1k, convnext_small.in12k_ft_in1k_384, levit_conv_128.fb_dist_in1k, volo_d3_224.sail_in1k, nest_tiny_jx.goog_in1k, mobileone_s2.apple_in1k, fastvit_t8.apple_dist_in1k, halo2botnet50ts_256.a1h_in1k, mobilenetv2_140.ra_in1k, caformer_m36.sail_in1k, seresnet50.ra2_in1k, hardcorenas_d.miil_green_in1k, convformer_b36.sail_in1k, regnety_320.swag_ft_in1k, volo_d4_448.sail_in1k, tf_efficientnet_b2.ns_jft_in1k, sebotnet33ts_256.a1h_in1k, vit_small_patch32_224.augreg_in21k_ft_in1k, vit_base_patch32_224.sam_in1k, resnetv2_50d_gn.ah_in1k, mobileone_s4.apple_in1k, coat_small.in1k, tf_mixnet_l.in1k, resnet34.a2_in1k, regnetx_032.pycls_in1k, resnetaa101d.sw_in12k_ft_in1k, lcnet_100.ra2_in1k, repvgg_b1.rvgg_in1k, crossvit_15_240.in1k, edgenext_x_small.in1k, repvit_m1_5.dist_300e_in1k, hardcorenas_a.miil_green_in1k, efficientformer_l1.snap_dist_in1k, tf_mobilenetv3_large_075.in1k, hrnet_w18_small.ms_in1k, tf_efficientnet_b2.in1k, ghostnetv2_130.in1k, ecaresnet26t.ra2_in1k, fastvit_s12.apple_in1k, xcit_tiny_12_p8_224.fb_dist_in1k, tresnet_m.miil_in21k_ft_in1k, fastvit_sa24.apple_in1k, resnetrs200.tf_in1k, convnextv2_nano.fcmae_ft_in1k, resnet50.ra_in1k, resnet34.bt_in1k, regnety_002.pycls_in1k, focalnet_base_lrf.ms_in1k, dla102.in1k, regnetz_e8.ra3_in1k, pvt_v2_b0.in1k, xcit_medium_24_p8_224.fb_in1k, regnety_640.seer_ft_in1k, resnet200d.ra2_in1k, caformer_s36.sail_in1k_384, deit3_small_patch16_384.fb_in22k_ft_in1k, eca_resnext26ts.ch_in1k, vgg13.tv_in1k, tf_efficientnet_lite0.in1k, resnet50.b1k_in1k, dla60_res2next.in1k, repvit_m1_1.dist_300e_in1k, convnext_base.fb_in22k_ft_in1k, tf_efficientnet_cc_b0_4e.in1k, ese_vovnet19b_dw.ra_in1k, resnetv2_152x2_bit.goog_teacher_in21k_ft_in1k, deit_base_distilled_patch16_384.fb_in1k, resnet101d.gluon_in1k, convnext_large.fb_in22k_ft_in1k_384, darknet53.c2ns_in1k, poolformerv2_s36.sail_in1k, convformer_m36.sail_in22k_ft_in1k_384, gmlp_s16_224.ra3_in1k, convformer_s18.sail_in1k, efficientnet_em.ra2_in1k, inception_v3.gluon_in1k, resmlp_12_224.fb_in1k, tresnet_l.miil_in1k, ecaresnet101d_pruned.miil_in1k, resnet152.a2_in1k, vit_small_patch32_384.augreg_in21k_ft_in1k, inception_v3.tf_adv_in1k, repghostnet_130.in1k, levit_conv_384.fb_dist_in1k, repvit_m1_5.dist_450e_in1k, efficientnet_el.ra_in1k, seresnet50.a2_in1k, pit_s_distilled_224.in1k, cspdarknet53.ra_in1k, tf_efficientnet_cc_b0_8e.in1k, densenet201.tv_in1k, resnext50_32x4d.a1_in1k, cs3sedarknet_l.c2ns_in1k, cait_s24_384.fb_dist_in1k, spnasnet_100.rmsp_in1k, res2net50_14w_8s.in1k, repvgg_d2se.rvgg_in1k, regnetx_032.tv2_in1k, crossvit_18_dagger_408.in1k, pit_b_distilled_224.in1k, cs3darknet_focus_l.c2ns_in1k, resnet50.bt_in1k, vgg11.tv_in1k, convnextv2_femto.fcmae_ft_in1k, convnext_nano.in12k_ft_in1k, resnext101_64x4d.tv_in1k, convnext_nano.d1h_in1k, cspresnet50.ra_in1k, tf_mixnet_m.in1k, xcit_tiny_12_p16_384.fb_dist_in1k, seresnet50.a1_in1k, efficientnetv2_rw_t.ra2_in1k, resnet152d.gluon_in1k, regnety_032.tv2_in1k, inception_resnet_v2.tf_in1k, eva_large_patch14_196.in22k_ft_in1k, pvt_v2_b1.in1k, convformer_m36.sail_in1k_384, densenet161.tv_in1k, dla102x.in1k, edgenext_small_rw.sw_in1k, regnety_016.tv2_in1k, convnextv2_base.fcmae_ft_in1k, vit_large_patch14_clip_336.laion2b_ft_in12k_in1k, levit_conv_128s.fb_dist_in1k, hrnet_w48.ms_in1k, resnet101.a1h_in1k, xcit_medium_24_p8_224.fb_dist_in1k, resnetrs152.tf_in1k, convnextv2_nano.fcmae_ft_in22k_in1k, convnextv2_tiny.fcmae_ft_in22k_in1k, resnext50d_32x4d.bt_in1k, gernet_s.idstcv_in1k, selecsls42b.in1k, repvit_m3.dist_in1k, resnest50d_1s4x24d.in1k, dpn98.mx_in1k, xcit_nano_12_p16_224.fb_in1k, regnetx_016.pycls_in1k, xcit_medium_24_p16_224.fb_in1k, caformer_s18.sail_in1k, sehalonet33ts.ra2_in1k, tinynet_c.in1k, xcit_tiny_24_p16_224.fb_dist_in1k, flexivit_small.300ep_in1k, resnext101_32x8d.tv2_in1k, convnextv2_base.fcmae_ft_in22k_in1k_384, semnasnet_075.rmsp_in1k, res2net50_26w_4s.in1k, cait_xxs24_384.fb_dist_in1k, mobilenetv2_120d.ra_in1k, seresnext26t_32x4d.bt_in1k, flexivit_base.1200ep_in1k, res2net50_26w_6s.in1k, vit_base_patch16_clip_384.openai_ft_in12k_in1k, nest_base_jx.goog_in1k, ecaresnetlight.miil_in1k, repvgg_b0.rvgg_in1k, ecaresnet50t.a1_in1k, inception_next_tiny.sail_in1k, regnety_032.pycls_in1k, mixer_b16_224.miil_in21k_ft_in1k, poolformer_s12.sail_in1k, vit_base_patch32_clip_384.openai_ft_in12k_in1k, vit_base_patch32_384.augreg_in21k_ft_in1k, efficientvit_b1.r224_in1k, vit_base_patch16_clip_384.laion2b_ft_in1k, deit_small_distilled_patch16_224.fb_in1k, efficientvit_b0.r224_in1k, resnest50d.in1k, regnety_120.pycls_in1k, semnasnet_100.rmsp_in1k, wide_resnet50_2.tv_in1k, xcit_small_24_p16_224.fb_in1k,

resnet101.a3_in1k, fastvit_t12.apple_in1k, tf_efficientnet_lite1.in1k, tinynet_a.in1k, resmlp_big_24_224.fb_distilled_in1k, cs3se_edgenet_x.c2ns_in1k, resnetv2_152x2_bit.goog_teacher_in21k_ft_in1k_384, resnext50_32x4d.tv2_in1k, efficientnet_b2.ra_in1k, convformer_s18.sail_in22k_ft_in1k_384, caformer_s18.sail_in22k_ft_in1k_384, deit3_base_patch16_224.fb_in1k, vit_base_patch32_clip_384.laion2b_ft_in12k_in1k, vit_medium_patch16_gap_384.sw_in12k_ft_in1k, sequencer2d_s.in1k, mobileone_s0.apple_in1k, edgenext_base.in21k_ft_in1k, deit3_medium_patch16_224.fb_in1k, efficientformerv2_l.snap_dist_in1k, lambda_resnet50ts.a1h_in1k, xception41p.ra3_in1k, resnext50_32x4d.a3_in1k, crossvit_small_240.in1k, repvgg_a1.rvgg_in1k, resnet51q.ra2_in1k, xcit_small_24_p16_384.fb_dist_in1k, vit_base_patch32_clip_224.openai_ft_in1k, flexivit_large.300ep_in1k, repvgg_b3g4.rvgg_in1k, resnext50_32x4d.a1h_in1k, coat_lite_medium.in1k, vit_base_patch32_clip_448.laion2b_ft_in12k_in1k, resnext50_32x4d.gluon_in1k, repvgg_b2.rvgg_in1k, vit_base_patch16_rpn_224.sw_in1k, mixer_b16_224.goog_in21k_ft_in1k, resnet50.c2_in1k, lamhalobotnet50ts_256.a1h_in1k, tiny_vit_21m_512.dist_in22k_ft_in1k, xcit_large_24_p16_224.fb_dist_in1k, repvgg_a2.rvgg_in1k, gernet_l.idstcv_in1k, mobilevitv2_050.cvnets_in1k, convnextv2_base.fcmae_ft_in22k_in1k, resnet18.a3_in1k, ecaresnet50d.miil_in1k, coat_lite_small.in1k, convnext_xlarge.fb_in22k_ft_in1k, mobilevitv2_075.cvnets_in1k, cait_s36_384.fb_dist_in1k, efficientformerv2_s1.snap_dist_in1k, resnet18.fb_swsl_ig1b_ft_in1k, mobileone_s1.apple_in1k, resnet61q.ra2_in1k, tf_efficientnetv2_b3.in1k, mobilevitv2_175.cvnets_in1k, convnext_tiny.fb_in22k_ft_in1k_384, crossvit_tiny_240.in1k, caformer_b36.sail_in22k_ft_in1k_384, resnet152d.ra2_in1k, convit_base.fb_in1k, tinynet_b.in1k, deit3_large_patch16_384.fb_in22k_ft_in1k, regnetx_004_tv.tv2_in1k, cait_xxs36_384.fb_dist_in1k, convnext_nano_ols.d1h_in1k, efficientnet_lite0.ra_in1k, inception_v4.tf_in1k, hrnet_w18.ms_in1k, gernet_m.idstcv_in1k, convformer_s36.sail_in22k_ft_in1k_384, deit_tiny_distilled_patch16_224.fb_in1k, deit_small_patch16_224.fb_in1k, vit_large_patch14_clip_336.laion2b_ft_in1k, crossvit_18_240.in1k, resnet26.bt_in1k, resnet18.a1_in1k, deit3_base_patch16_384.fb_in22k_ft_in1k, convformer_s36.sail_in1k, convnext_small.fb_in22k_ft_in1k, selecsls60b.in1k, efficientnet_b0.ra_in1k, focalnet_tiny_srf.ms_in1k, ecaresnet101d.miil_in1k, regnetx_080.tv2_in1k, mobileone_s3.apple_in1k, mobilenetv3_rw.rmsp_in1k, poolformerv2_m36.sail_in1k, seresnextaa101d_32x8d.ah_in1k, levit_conv_192.fb_dist_in1k, focalnet_tiny_lrf.ms_in1k, regnety_320.swag_lc_in1k, tresnet_v2_l.miil_in21k_ft_in1k, seresnet50.a3_in1k, dla46x_c.in1k, cs3darknet_x.c2ns_in1k, tf_efficientnet_b0.ap_in1k, vit_base_patch16_224.augreg2_in21k_ft_in1k, resnext101_32x8d.fb_ssl_yfcc100m_ft_in1k, xcit_large_24_p8_384.fb_dist_in1k, tinynet_e.in1k, cait_xs24_384.fb_dist_in1k, fastvit_sa12.apple_in1k, hrnet_w64.ms_in1k, regnety_016.pycls_in1k, wide_resnet101_2.tv2_in1k, beitv2_large_patch16_224.in1k_ft_in22k_in1k, hrnet_w30.ms_in1k, resnet101.tv_in1k, repvit_m2.dist_in1k, coatnet_nano_rw_224.sw_in1k, flexivit_small.1200ep_in1k, tf_efficientnet_b0.in1k, tf_efficientnet_b1.in1k, efficientformer_l3.snap_dist_in1k, vit_base_patch16_384.augreg_in21k_ft_in1k, xcit_tiny_24_p8_224.fb_dist_in1k, dla102x2.in1k, hardcorenas_f.miil_green_in1k, regnety_064.ra3_in1k, resnext101_32x4d.gluon_in1k, tf_efficientnetv2_b2.in1k, resnet32ts.ra2_in1k, xcit_tiny_12_p8_384.fb_dist_in1k, inception_v3.tv_in1k, xcit_large_24_p16_224.fb_in1k, ecaresnet50t.a3_in1k, repvit_m2_3.dist_450e_in1k, fbnetv3_b.ra2_in1k, vit_base_patch8_224.augreg2_in21k_ft_in1k, cs3darknet_l.c2ns_in1k, convnext_base.clip_laion2b_augreg_ft_in12k_in1k_384, regnety_160.deit_in1k, regnety_160.pycls_in1k, dla60x.in1k, xcit_tiny_24_p16_384.fb_dist_in1k, eva02_tiny_patch14_336.mim_in22k_ft_in1k, volo_d2_224.sail_in1k, regnety_160.swag_lc_in1k, vit_base_patch32_clip_224.laion2b_ft_in12k_in1k, tf_mixnet_s.in1k, repvit_m1_0.dist_300e_in1k, convnextv2_large.fcmae_ft_in1k, resmlp_12_224.fb_distilled_in1k, xcit_medium_24_p16_384.fb_dist_in1k, regnety_080_tv.tv2_in1k, dpn107.mx_in1k, inception_v3.tf_in1k, dpn68.mx_in1k, efficientnet_es.ra_in1k, mnasnet_100.rmsp_in1k, resnet101.tv2_in1k, res2next50.in1k, vit_base_patch16_clip_384.openai_ft_in1k, tf_efficientnet_b1.ns_jft_in1k, flexivit_small.600ep_in1k, visformer_tiny.in1k, resnet50.a1_in1k, dla60.in1k, regnetz_d32.ra3_in1k, senet154.gluon_in1k, efficientnetv2_rw_s.ra2_in1k, focalnet_small_lrf.ms_in1k, seresnet33ts.ra2_in1k, fbnetc_100.rmsp_in1k, resnet18d.ra2_in1k, resnet34.a3_in1k, dla60x_c.in1k, efficientnet_b1_pruned.in1k, efficientformerv2_s2.snap_dist_in1k, resnet50s.gluon_in1k, resnet101.a2_in1k, regnety_040.ra3_in1k, convmixer_1536_20.in1k, regnety_008_tv.tv2_in1k, resnet152.a1_in1k, mixnet_l.ft_in1k, gcresnext26ts.ch_in1k, vit_base_patch16_clip_224.openai_ft_in1k, fastvit_ma36.apple_in1k, vgg16.tv_in1k, gcresnext50ts.ch_in1k, xcit_tiny_12_p16_224.fb_dist_in1k, regnety_008.pycls_in1k, resmlp_36_224.fb_distilled_in1k, regnetz_040_h.ra3_in1k, inception_next_base.sail_in1k, dm_nfnet_f0.dm_in1k, resnet50.d_in1k, efficientnet_b2_pruned.in1k, resnet18.tv_in1k, rexnet_150.nav_in1k, convnext_large_mlp.clip_laion2b_soup_ft_in12k_in1k_320, ghostnetv2_160.in1k, vit_small_patch16_384.augreg_in21k_ft_in1k, convnext_xlarge.fb_in22k_ft_in1k_384, mobilenetv3_small_075.lamb_in1k, regnetz_d8_evos.ch_in1k, dm_nfnet_f3.dm_in1k, repvgg_b3.rvgg_in1k, convnext_large_mlp.clip_laion2b_augreg_ft_in1k_384, dpn68b.mx_in1k, resnext101_32x8d.fb_wsl_ig1b_ft_in1k, deit3_large_patch16_384.fb_in1k, convformer_s18.sail_in1k_384, repghostnet_058.in1k, fastvit_sa36.apple_dist_in1k, resnext50_32x4d.a2_in1k, regnetx_040.pycls_in1k, vit_base_r50_s16_384.orig_in21k_ft_in1k, vit_base_patch16_clip_224.laion2b_ft_in1k, deit3_base_patch16_384.fb_in1k, tf_efficientnetv2_s.in1k, ecaresnet50t.a2_in1k, resnetrs50.tf_in1k, gmixer_24_224.ra3_in1k, resnetaa50d.sw_in12k_ft_in1k, tresnet_xl.miil_in1k, resnest101e.in1k, regnetx_004.pycls_in1k, mnasnet_small.lamb_in1k, repvgg_a0.rvgg_in1k, resnetv2_50x1_bit.goog_in21k_ft_in1k, cait_s24_224.fb_dist_in1k, regnety_004.tv2_in1k, convnext_base.fb_in22k_ft_in1k_384, convnext_tiny.fb_in22k_ft_in1k, convnext_tiny.in12k_ft_in1k_384, eca_halonext26ts.c1_in1k, resnet18.gluon_in1k, fastvit_s12.apple_dist_in1k, deit_base_patch16_224.fb_in1k, hrnet_w18.ms_aug_in1k, resnet33ts.ra2_in1k, seresnext101_64x4d.gluon_in1k, convnext_small.fb_in1k, convformer_s36.sail_in1k_384, pit_ti_distilled_224.in1k, resnet50.tv2_in1k, nest_small_jx.goog_in1k, resmlp_36_224.fb_in1k, hrnet_w18_small.gluon_in1k, vit_base_patch16_384.augreg_in1k, resnet50.fb_swsl_ig1b_ft_in1k, poolformer_m36.sail_in1k, tf_mobilenetv3_small_100.in1k, regnety_040.pycls_in1k, gcresnet33ts.ra2_in1k, resnet101s.gluon_in1k, darknetaa53.c2ns_in1k, poolformerv2_s12.sail_in1k, resnext50_32x4d.fb_ssl_yfcc100m_ft_in1k, poolformerv2_s24.sail_in1k, eca_resnet33ts.ra2_in1k, repvit_m2_3.dist_300e_in1k, nf_resnet50.ra2_in1k, convnext_pico_ols.d1_in1k, caformer_s36.sail_in1k, regnetz_040.ra3_in1k, vit_small_r26_s32_224.augreg_in21k_ft_in1k, resnext26ts.ra2_in1k, mixnet_xl.ra_in1k, deit_base_patch16_384.fb_in1k, repvit_m1_0.dist_450e_in1k, convmixer_1024_20_ks9_p14.in1k, regnety_064.pycls_in1k,

resnet34.gluon_in1k, res2net101_26w_4s.in1k, nfnet_l0.ra2_in1k, resnet34d.ra2_in1k, convnextv2_nano.fcmae_ft_in22k_in1k_384, twins_pcpvt_base.in1k, resnetv2_101.a1h_in1k, xcit_nano_12_p8_224.fb_dist_in1k, xcit_small_24_p8_224.fb_dist_in1k, resnet50.b2k_in1k, deit3_small_patch16_384.fb_in1k, hardcorenas_c.miil_green_in1k, coat_lite_mini.in1k, resnet152.tv2_in1k, densenetblur121d.ra_in1k, hrnet_w18_small_v2.gluon_in1k, vit_base_patch16_384.orig_in21k_ft_in1k, xcit_small_12_p8_224.fb_dist_in1k, convformer_m36.sail_in1k, xcit_nano_12_p16_384.fb_dist_in1k, resnet34.a1_in1k, convnext_atto_ols.a2_in1k, resnet14t.c3_in1k, twins_pcpvt_large.in1k, resnest26d.gluon_in1k, mobilenetv3_small_100.lamb_in1k, efficientnet_b3_pruned.in1k, vit_small_patch16_224.augreg_in1k, convnext_tiny.fb_in1k, resnet50d.a3_in1k, mobilevitv2_175.cvnets_in22k_ft_in1k_384, deit3_medium_patch16_224.fb_in22k_ft_in1k, seresnext101_32x4d.gluon_in1k, hardcorenas_b.miil_green_in1k, caformer_m36.sail_in22k_ft_in1k, ghostnetv2_100.in1k, ecaresnet50d_pruned.miil_in1k, caformer_s36.sail_in22k_ft_in1k_384, deit_tiny_patch16_224.fb_in1k, fastvit_sa36.apple_in1k, regnety_320.seer_ft_in1k, edgenext_small.usi_in1k, resmlp_big_24_224.fb_in22k_ft_in1k, regnety_160.lion_in12k_ft_in1k, regnety_160.sw_in12k_ft_in1k, tf_efficientnet_b1.ap_in1k, res2net50_48w_2s.in1k, eca_botnext26ts_256.c1_in1k, xcit_small_24_p8_224.fb_in1k, crossvit_9_dagger_240.in1k, coat_lite_tiny.in1k, resnetv2_101x1_bit.goog_in21k_ft_in1k, convnext_large_mlp.clip_laion2b_augreg_ft_in1k, xcit_nano_12_p16_224.fb_dist_in1k, cs3darknet_focus_m.c2ns_in1k, wide_resnet50_2.tv2_in1k, vit_base_patch16_clip_224.openai_ft_in12k_in1k, skresnet34.ra_in1k, repvgg_b1g4.rvgg_in1k, vgg19_bn.tv_in1k, repghostnet_100.in1k, regnetv_064.ra3_in1k, mobilenetv2_100.ra_in1k, convnext_femto.d1_in1k, resnet26t.ra2_in1k, regnetv_040.ra3_in1k, skresnet18.ra_in1k, caformer_m36.sail_in22k_ft_in1k_384, vit_base_patch32_384.augreg_in1k, regnetz_b16.ra3_in1k, hrnet_w48_ssld.paddle_in1k, resnest50d_4s2x40d.in1k, cait_xxs36_224.fb_dist_in1k, regnetx_016.tv2_in1k, xcit_small_24_p8_384.fb_dist_in1k, vit_tiny_r_s16_p8_224.augreg_in21k_ft_in1k, coat_mini.in1k, xcit_small_24_p16_224.fb_dist_in1k, caformer_s36.sail_in22k_ft_in1k, poolformer_s24.sail_in1k, resmlp_big_24_224.fb_in1k, regnetx_120.pycls_in1k, regnetz_d8.ra3_in1k, resnet50d.ra2_in1k, repvit_m1.dist_in1k, eca_nfnet_l2.ra3_in1k, resnet50d.gluon_in1k, seresnext50_32x4d.racm_in1k, vit_small_patch16_384.augreg_in1k, coat_tiny.in1k, xcit_nano_12_p8_224.fb_in1k, crossvit_base_240.in1k, resnet50d.a1_in1k, convformer_s36.sail_in22k_ft_in1k, convnextv2_large.fcmae_ft_in22k_in1k, resnet50.tv_in1k, resnet50.c1_in1k, pit_xs_distilled_224.in1k, efficientnet_b1.ft_in1k, tf_efficientnet_el.in1k, hrnet_w32.ms_in1k, vit_base_patch16_224_miil.in21k_ft_in1k, cs3sedarknet_x.c2ns_in1k, dpn68b.ra_in1k, tf_efficientnetv2_b1.in1k, regnety_004.pycls_in1k, tf_mobilenetv3_large_minimal_100.in1k, resnetrs101.tf_in1k, ese_vovnet39b.ra_in1k, mixer_l16_224.goog_in21k_ft_in1k, repghostnet_050.in1k, repvgg_b2g4.rvgg_in1k, repvit_m1_1.dist_450e_in1k, vit_base_patch32_224.augreg_in21k_ft_in1k, tf_mobilenetv3_large_100.in1k, pit_s_224.in1k, caformer_s18.sail_in22k_ft_in1k, wide_resnet101_2.tv_in1k, fastvit_t12.apple_dist_in1k, convmixer_768_32.in1k, vit_base_patch32_224.augreg_in1k, efficientformerv2_s0.snap_dist_in1k, resnest200e.in1k, levit_conv_256.fb_dist_in1k, resnet18.fb_ssl_yfcc100m_ft_in1k, vgg13_bn.tv_in1k, resnet152c.gluon_in1k, dla169.in1k, pvt_v2_b4.in1k, crossvit_15_dagger_408.in1k, convnext_femto_ols.d1_in1k, convnext_large.fb_in1k, regnetx_064.pycls_in1k, fastvit_t8.apple_in1k, seresnet152d.ra2_in1k, vgg19.tv_in1k, vgg11_bn.tv_in1k, dm_nfnet_f2.dm_in1k, seresnext101d_32x8d.ah_in1k, inception_next_base.sail_in1k_384, lambda_resnet26t.c1_in1k, resnetv2_152x2_bit.goog_in21k_ft_in1k, fastvit_ma36.apple_dist_in1k, regnety_006.pycls_in1k, regnety_080.pycls_in1k, resnet50.fb_ssl_yfcc100m_ft_in1k, tf_mobilenetv3_small_075.in1k, regnetz_c16.ra3_in1k, edgenext_xx_small.in1k, crossvit_9_240.in1k, xcit_tiny_24_p8_224.fb_in1k, regnety_080.ra3_in1k, efficientvit_b1.r256_in1k, tinynet_d.in1k, caformer_b36.sail_in1k_384, pvt_v2_b2.in1k, resnet26d.bt_in1k, convnext_pico.d1_in1k, pit_b_224.in1k, convnextv2_pico.fcmae_ft_in1k, fbnetv3_d.ra2_in1k, flexivit_large.1200ep_in1k, resnet50c.gluon_in1k, regnetx_080.pycls_in1k, convnext_base.fb_in1k, tf_efficientnet_em.in1k, vit_base_patch16_224.augreg_in1k, convit_tiny.fb_in1k, resnext50_32x4d.fb_swsl_ig1b_ft_in1k, dm_nfnet_f4.dm_in1k, resnet50.a3_in1k, convnext_atto.d2_in1k, efficientnet_el_pruned.in1k, volo_d2_384.sail_in1k, resnext101_32x4d.fb_ssl_yfcc100m_ft_in1k, repvit_m0_9.dist_300e_in1k, regnety_120.sw_in12k_ft_in1k, beit_base_patch16_384.in22k_ft_in22k_in1k, mobilenetv3_large_100.miil_in21k_ft_in1k, tf_efficientnet_b0.aa_in1k, inception_next_small.sail_in1k, deit_base_distilled_patch16_224.fb_in1k, lcnet_075.ra2_in1k, xcit_tiny_12_p8_224.fb_in1k, resnet101.gluon_in1k, dpn92.mx_in1k, resnet101.a1_in1k, selecsls60.in1k, beit_base_patch16_224.in22k_ft_in22k_in1k, convnextv2_tiny.fcmae_ft_in1k, res2net50_26w_8s.in1k, sequencer2d_m.in1k, vit_medium_patch16_gap_256.sw_in12k_ft_in1k, regnetx_008.pycls_in1k, resnet50.a2_in1k, res2net101d.in1k, vit_large_patch16_384.augreg_in21k_ft_in1k, pvt_v2_b2_li.in1k, regnetx_006.pycls_in1k, xcit_tiny_24_p16_224.fb_in1k, pvt_v2_b5.in1k, resnext50_32x4d.ra_in1k, resnest14d.gluon_in1k, caformer_m36.sail_in1k_384, resnet50.gluon_in1k, resnet152s.gluon_in1k, flexivit_large.600ep_in1k, resnetv2_50x1_bit.goog_distilled_in1k, resmlp_24_224.fb_in1k, deit3_large_patch16_224.fb_in1k, seresnext50_32x4d.gluon_in1k, densenet121.tv_in1k, resnet152.a3_in1k, ghostnet_100.in1k, tf_efficientnet_b2.ap_in1k, regnetx_002.pycls_in1k.

K Additional Results

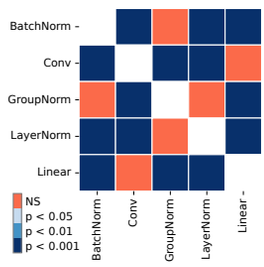


Fig. 18: **Differences Between Layer Types are Significant.** We analyze and test for statistical significances in the differences in MIS between different layer types (see Fig. 5. The reported significance levels were computed using Conover's test over the per-model and per-layer-type means with Holm's correction for multiple comparisons.

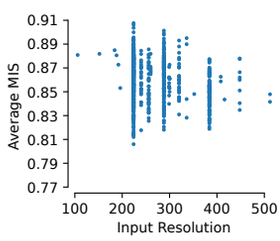


Fig. 19: **Influence of Input Resolution of MIS.** We show the average MIS per model as a function of the model's input resolution. No trend is apparent; models with the same resolution yield different interpretability levels.

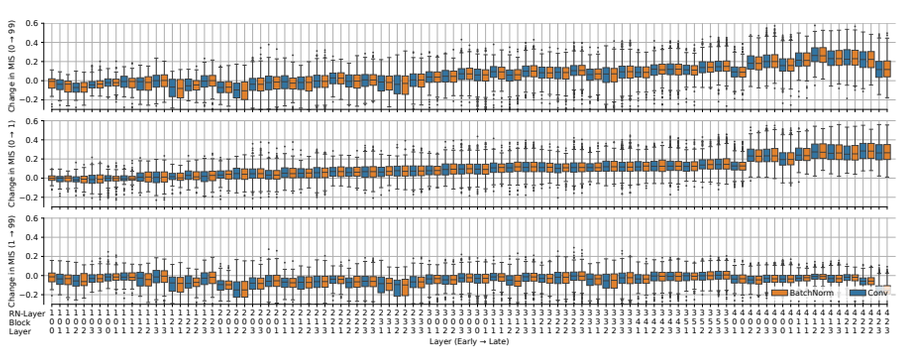


Fig. 20: **Change of Interpretability per Layer During Training.** Detailed version of Fig. 8.

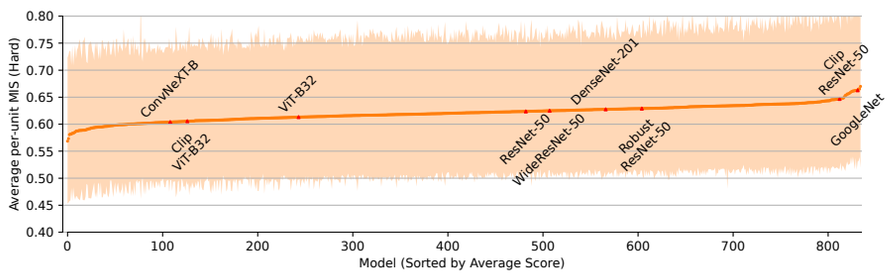


Fig. 21: **Comparison of the Average Per-unit MIS for Models for a Different Task Difficulty.** Our proposed MIS can easily be extended to test more than just the extrema of the activation distribution: Instead of choosing the most extremely activating samples as query images, we can sample less strongly activating ones from other parts of the activation distribution. By sampling from the 2nd/98th percentile, we can recompute Fig. 3 on a more challenging version of the underlying 2-AFC task.

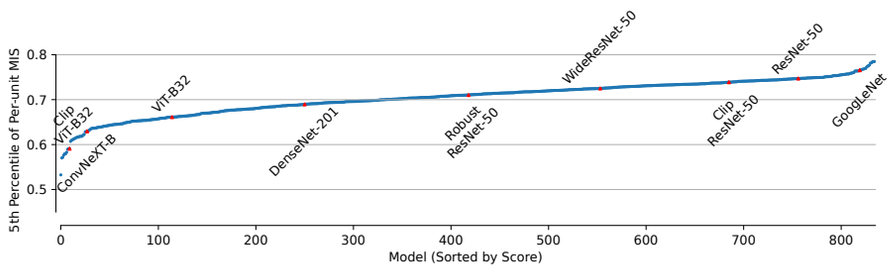


Fig. 22: **Comparison of the Minimum of the Per-unit MIS for Models.** While the mean of the per-unit interpretability varies in a rather narrow value range (see Fig. 3), we investigate differences in the distribution of scores. Specifically, we are interested in the effective width of the distribution, i.e., how low does the minimal MIS per model go? To make the analysis robust against outliers, we do not use the minimum but instead the 5th percentile. Note that this corresponds to the lower end of the shaded area in Fig. 3. Compared to the average MIS, we see higher variability across models.

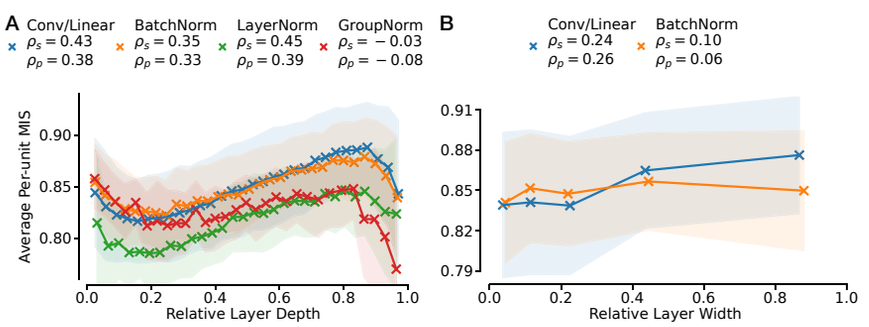


Fig. 23: **(A) Deeper Layers are More Interpretable.** Average MIS per layer as a function of the relative depth of the layer within the network, grouped by layer types. For each type, the values are grouped into 30 bins of equal count based on the relative depth. The markers shown correspond to the bin average, the shaded areas indicate the standard deviation. Correlations are computed for the ungrouped data points. While the standard deviation appears moderately high, note that the found trends are consistent over many bins of various layer types. **(B) Wider Layers are More Interpretable.** Average MIS per layer as a function of the relative width of the layer compared to all layers of the same type in the network, grouped by layer types. The values are grouped into 5 bins.

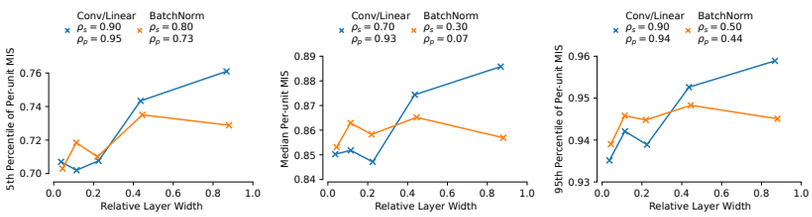


Fig. 24: **Wider Layers are More Interpretable.** Wider layers within a network are moderately more interpretable based on the computed MIS. This trend holds for both the per-layer-average (see Fig. 6B) as well as the 5th percentile, median, or 95th percentile of the per-layer distribution as shown here from left to right. This suggests that the overall distribution is shifted to higher MIS values for wider layers, compared to just a few outliers that positively influence the average value.

Tab. 24: **Pareto-optimal Models for Optimizing ImageNet Accuracy and MIS.** As Fig. 4A shows an anticorrelation between ImageNet top-1 accuracy and MIS, we here list the Pareto-optimal models for optimizing both accuracy and MIS at the same time.

Model	ImageNet top-1 Accuracy [%]	MIS
GoogLeNet	69.15	0.908
timm:resnet34.a3_in1k	72.97	0.904
timm:resnet50_gn.a1h_in1k	81.22	0.901
timm:ecaresnet101d_pruned.miil_in1k	82.00	0.985
timm:eva02_small_patch14_336.mim_in22k_ft_in1k	85.72	0.890
timm:vit_base_patch8_224.augreg_in21k_ft_in1k	85.8	0.871
timm:caformer_b36.sail_in1k_384	86.41	0.870
timm:caformer_s36.sail_in22k_ft_in1k_384	86.86	0.870
timm:caformer_b36.sail_in22k_ft_in1k_384	88.06	0.864
timm:beitv2_large_patch16_224.in1k_ft_in22k_in1k	88.39	0.39

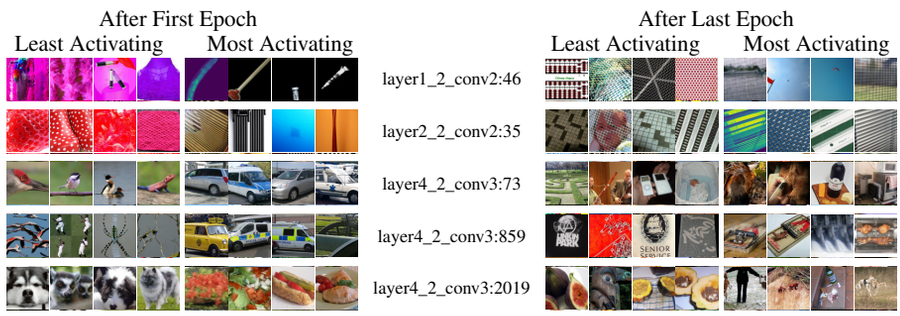
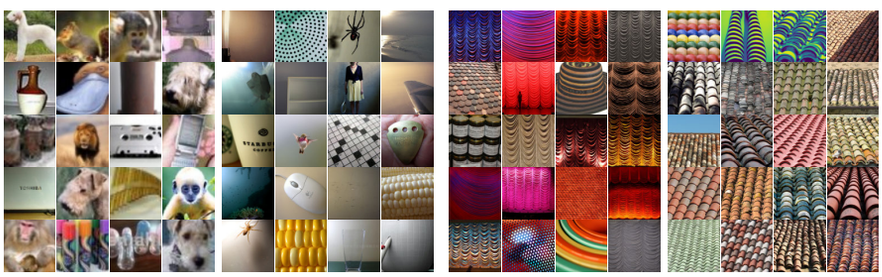
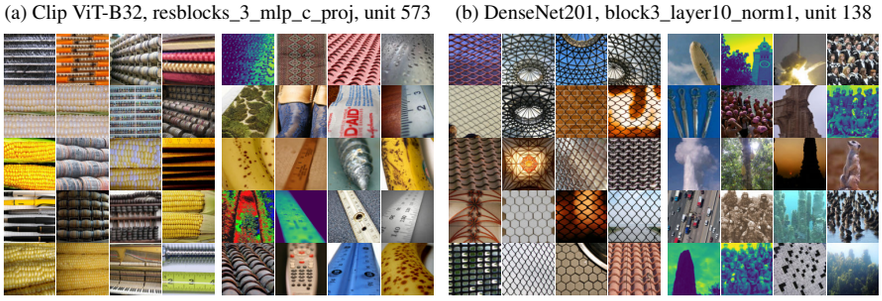


Fig. 25: **How do Dataset Exemplars for Units with Strong MIS Drop Change?** To gain a better understanding of why the MIS of a ResNet50 drops during training after the first epoch, we display the least/most activating dataset exemplars of four units from the model after the first (left) and after the last (right) epoch. While the explanations after the first epoch seem to focus on easy-to-grasp visual features, the units on the right react to less clear-cut concepts. The units are among the units with the strongest MIS drop in the convolutional layers with the strongest MIS drop.

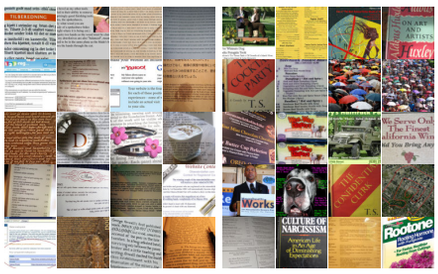


(a) Clip ViT-B32, resblocks_3_mlp_c_proj, unit 573 (b) DenseNet201, block3_layer10_norm1, unit 138

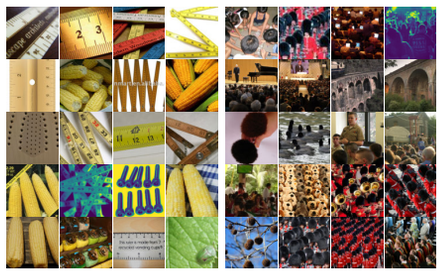


(c) DenseNet201, block3_layer29_conv1, unit 39 (d) DenseNet201, block3_layer35_norm2, unit 123

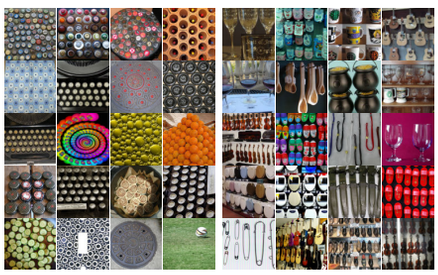
Fig. 26: **Visualization of Units for which MIS overestimates HIS.** To showcase the shortcomings of the MIS, we visualize four units for which the MIS predicts an interpretability that is higher than the measured HIS in Fig. 2B. See Fig. 27 for the opposite direction. For each unit, we show the 20 most (right) and 20 least (left) activating dataset exemplars.



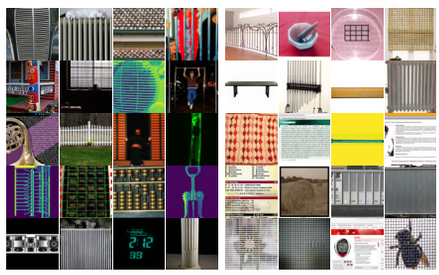
(a) Clip ResNet-50, layer4_1_conv2, unit 430



(b) DenseNet201, block3_layer48_conv2, unit 21

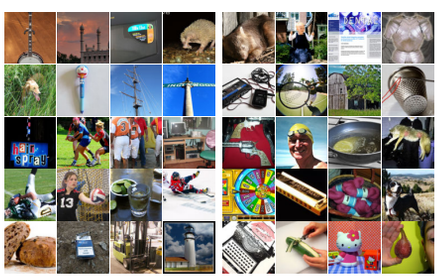


(c) DenseNet201, block3_layer48_norm1, unit 1369

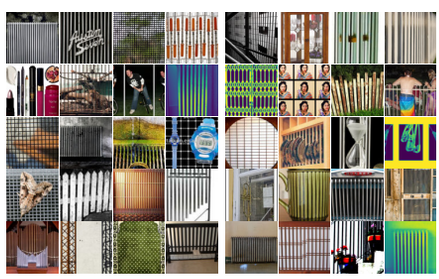


(d) ViT-B32, block0_norm2, unit 358

Fig. 27: **Visualization of Units for which MIS underestimates HIS.** To showcase the shortcomings of the MIS, we visualize four units for which the MIS predicts an interpretability that is lower than the measured HIS in Fig. 2B. See Fig. 26 for the opposite direction. For each unit, we show the 20 most (right) and 20 least (left) activating dataset exemplars.

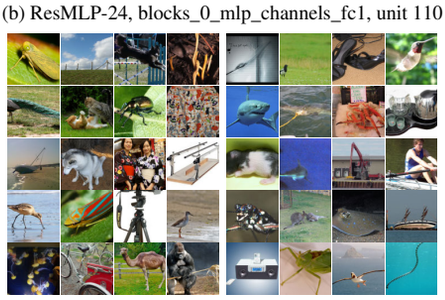


(a) ResMLP-36, blocks_10_linear_tokens, unit 61



(b) ResMLP-24, blocks_0_mlp_channels_fc1, unit 110

(c) GMixer-24, blocks_5_mlp_tokens_fc1, unit 166



(d) ResMLP-12, blocks_7_linear_tokens, unit 127

Fig. 28: **Visualization of Hard Units from Models with High Variability.** For the four models with the highest variability in MIS (see Fig. 4B), we visualize one of the units with the lowest MIS each. For each unit, we show the 20 most (right) and 20 least (left) activating dataset exemplars.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The contributions claimed in the abstract and introduction are backed up by experimental results in Sec. 4.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We mention the limitations of our work throughout the paper, e.g., when we introduce it in Sec. 3 or in Sec. 5, as well in the appendix in Appx. H.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: This paper presents no theoretical results but only empirical findings. Thus, this question does not apply.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: A detailed description of how our proposed metric is computed is given in Sec. 3. The conducted experiments are described in the first paragraph of each subsection in Sec. 4. The experimental settings are stated in Sec. 3, Appx. A.2 and Appx. B.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We grant open access to this paper's experimental code. It is shared, along with its documentation, in the supplementary material.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental settings are stated in Sec. 3, Appx. A.2 and Appx. B. Furthermore, specific experiments are always described in the first paragraph of each subsection in Sec. 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We visualize the uncertainty of our experimental results with error bars, unless this severely degrades the accessibility of a figure due to cluttering (e.g., Fig. 2B). unless stated otherwise, the error bars shown in this paper depict the difference between the 5 % and 95 % percentile of the per-unit distribution.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We explain the computational complexity of our proposed method and the resources required for reproducing our experiments in Appx. G.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We read the Code of Ethics and ensured our work follows its guiding principles.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We outline potential positive impacts of our work in Sec. 5 and potential negative impact in Appx. H.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This study presents an analysis tool and no new dataset or powerful model. Therefore, this question does not apply.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: This study uses two datasets (ImageNet [40] and IMI [50]) that are introduced and cited in Sec. 4 and Sec. 3, respectively.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: This paper introduces a new analysis tool/metric. Its implementation and further experimental code are published, along with its documentation in the supplementary material.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: We describe the setup of the conducted psychophysical experiment in Appx. B, where we also describe the workers' compensation and show screenshots of the experiment.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

Justification: Our experiments did not represent any larger risk than normal computer use. For pure psychophysical experiments with non-offensive stimuli, a choice task, and mouse clicks, we did not consider sending a request to our IRB. Participants were informed that they consent to their anonymized data being used for a scientific study before agreeing to participate.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

A.6 Contrastive Learning Inverts the Data Generating Process

The following 21 pages were published as:

Roland S. Zimmermann*, Yash Sharma*, Steffen Schneider*, Matthias Bethge, and Wieland Brendel. "Contrastive learning inverts the data generating process." *ICML (2021)*

A summary is given in Section 3.1 on page 45. * Equal contribution.

Abstract

Contrastive learning has recently seen tremendous success in self-supervised learning. So far, however, it is largely unclear why the learned representations generalize so effectively to a large variety of downstream tasks. We here prove that feedforward models trained with objectives belonging to the commonly used InfoNCE family learn to implicitly invert the underlying generative model of the observed data. While the proofs make certain statistical assumptions about the generative model, we observe empirically that our findings hold even if these assumptions are severely violated. Our theory highlights a fundamental connection between contrastive learning, generative modeling, and nonlinear independent component analysis, thereby furthering our understanding of the learned representations as well as providing a theoretical foundation to derive more effective contrastive losses.

Contrastive Learning Inverts the Data Generating Process

Roland S. Zimmermann [*1 2] **Yash Sharma** [*1 2] **Steffen Schneider** [*1 2 3] **Matthias Bethge** [†1] **Wieland Brendel** [†1]

Abstract

Contrastive learning has recently seen tremendous success in self-supervised learning. So far, however, it is largely unclear why the learned representations generalize so effectively to a large variety of downstream tasks. We here prove that feedforward models trained with objectives belonging to the commonly used InfoNCE family learn to implicitly invert the underlying generative model of the observed data. While the proofs make certain statistical assumptions about the generative model, we observe empirically that our findings hold even if these assumptions are severely violated. Our theory highlights a fundamental connection between contrastive learning, generative modeling, and nonlinear independent component analysis, thereby furthering our understanding of the learned representations as well as providing a theoretical foundation to derive more effective contrastive losses.[1]

1. Introduction

With the availability of large collections of unlabeled data, recent work has led to significant advances in self-supervised learning. In particular, contrastive methods have been tremendously successful in learning representations for visual and sequential data (Logeswaran & Lee, 2018; Wu et al., 2018; Oord et al., 2018; Hénaff, 2020; Tian et al., 2019; Hjelm et al., 2019; Bachman et al., 2019; He et al., 2020a; Chen et al., 2020a; Schneider et al., 2019; Baevski et al., 2020a;b; Ravanelli et al., 2020). While a number of explanations have been provided as to why contrastive learning leads to such informative representations, existing theoretical predictions and empirical observations appear to be at odds with each other (Tian et al., 2019; Bachman et al., 2019; Wu et al., 2020; Saunshi et al., 2019).

In a nutshell, contrastive methods aim to learn representations where related samples are aligned (positive pairs, e.g. augmentations of the same image), while unrelated samples are separated (negative pairs) (Chen et al., 2020a). Intuitively, this leads to invariance to irrelevant details or transformations (by decreasing the distance between positive pairs), while preserving a sufficient amount of information about the input for solving downstream tasks (by increasing the distance between negative pairs) (Tian et al., 2020). This intuition has recently been made more precise by (Wang & Isola, 2020), showing that a commonly used contrastive loss from the InfoNCE family (Gutmann & Hyvärinen, 2012; Oord et al., 2018; Chen et al., 2020a) asymptotically converges to a sum of two losses: an *alignment* loss that pulls together the representations of positive pairs, and a *uniformity* loss that maximizes the entropy of the learned latent distribution.

We show that an encoder learned with a contrastive loss from the InfoNCE family can recover the true generative factors of variation (up to rotations) if the process that generated the data fulfills a few weak statistical assumptions. This theory bridges the gap between contrastive learning, nonlinear independent component analysis (ICA) and generative modeling (see Fig. 1). Our theory reveals implicit assumptions encoded in the InfoNCE objective about the generative process underlying the data. If these assumptions are violated, we show a principled way of deriving alternative contrastive objectives based on assumptions regarding the positive pair distribution. We verify our theoretical findings with controlled experiments, providing evidence that our theory holds true in practice, even if the assumptions on the ground-truth generative model are partially violated.

To the best of our knowledge, our work is the first to analyze under what circumstances representation learning methods used in practice provably represent the data in terms of its underlying factors of variation. Our theoretical and empirical results suggest that the success of contrastive learning in many practical applications is due to an implicit and approximate inversion of the data generating process, which explains why the learned representations are useful in a wide range of downstream tasks.

In summary, our contributions are:

[*]Equal contribution. [†]Joint supervision [1]University of Tübingen, Tübingen, Germany [2]IMPRS for Intelligent Systems, Tübingen, Germany [3]EPFL, Geneva, Switzerland. Correspondence to: Roland S. Zimmermann <roland.zimmermann@uni-tuebingen.de>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

[1]Online version and code: brendel-group.github.io/cl-ica/

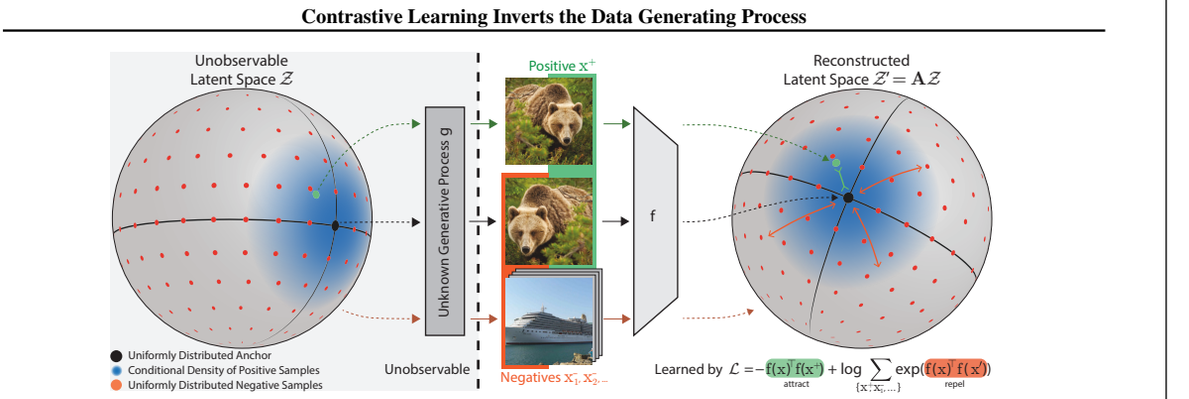


Figure 1. We analyze the setup of contrastive learning, in which a feature encoder f is trained with the InfoNCE objective (Gutmann & Hyvärinen, 2012; Oord et al., 2018; Chen et al., 2020a) using positive samples (green) and negative samples (orange). We assume the observations are generated by an (unknown) injective generative model g that maps unobservable latent variables from a hypersphere to observations in another manifold. Under these assumptions, the feature encoder f implictly learns to invert the ground-truth generative process g up to linear transformations, i.e., $f = \mathbf{A}g^{-1}$ with an orthogonal matrix $\mathbf{A}$, if f minimizes the InfoNCE objective.

- We establish a theoretical connection between the InfoNCE family of objectives, which is commonly used in self-supervised learning, and nonlinear ICA. We show that training with InfoNCE inverts the data generating process if certain statistical assumptions on the data generating process hold.
- We empirically verify our predictions when the assumed theoretical conditions are fulfilled. In addition, we show successful inversion of the data generating process even if these theoretical assumptions are partially violated.
- We build on top of the CLEVR rendering pipeline (Johnson et al., 2017b) to generate a more visually complex disentanglement benchmark, called *3DIdent*, that contains hallmarks of natural environments (shadows, different lighting conditions, a 3D object, etc.). We demonstrate that a contrastive loss derived from our theoretical framework can identify the ground-truth factors of such complex, high-resolution images.

2. Related Work

Contrastive Learning Despite the success of contrastive learning (CL), our understanding of the learned representations remains limited, as existing theoretical explanations yield partially contradictory predictions. One way to theoretically motivate CL is to refer to the InfoMax principle (Linsker, 1988), which corresponds to maximizing the mutual information (MI) between different views (Oord et al., 2018; Bachman et al., 2019; Hjelm et al., 2019; Chen et al., 2020a; Tian et al., 2020). However, as optimizing a tighter bound on the MI can produce worse representations (Tschannen et al., 2020), it is not clear how accurate this motivation describes the behavior of CL.

Another approach aims to explain the success by introducing latent classes (Saunshi et al., 2019). While this theory has some appeal, there exists a gap between empirical observations and its predictions, e.g. the prediction that an excessive number of negative samples decreases performance does not corroborate with empirical results (Wu et al., 2018; Tian et al., 2019; He et al., 2020a; Chen et al., 2020a). However, recent work has suggested some empirical evidence for said theoretical prediction, namely, issues with the commonly used sampling strategy for negative samples, and have proposed ways to mitigate said issues as well (Robinson et al., 2020; Chuang et al., 2020).

More recently, the behavior of CL has been analyzed from the perspective of *alignment* and *uniformity* properties of representations, demonstrating that these two properties are correlated with downstream performance (Wang & Isola, 2020). We build on these results to make a connection to cross-entropy minimization from which we can derive identifiability results.

Nonlinear ICA Independent Components Analysis (ICA) attempts to find the underlying sources for multidimensional data. In the nonlinear case, said sources correspond to a well-defined nonlinear generative model g, which is assumed to be invertible (i.e., injective) (Hyvärinen et al., 2001; Jutten et al., 2010). In other words, nonlinear ICA solves a demixing problem: Given observed data $\mathbf{x} = g(\mathbf{z})$, it aims to find a model f that equals the inverse generative model g^{-1}, which allows for the original sources $\mathbf{z}$ to be recovered.

Hyvärinen et al. (2019) show that the nonlinear demixing problem can be solved as long as the independent compo-

nents are conditionally mutually independent with respect to some auxiliary variable. The authors further provide practical estimation methods for solving the nonlinear ICA problem (Hyvärinen & Morioka, 2016; 2017), similar in spirit to noise contrastive estimation (NCE; Gutmann & Hyvärinen, 2012). Recent work has generalized this contribution to VAEs (Khemakhem et al., 2020a; Locatello et al., 2020; Klindt et al., 2021), as well as (invertible-by-construction) energy-based models (Khemakhem et al., 2020b). We here extend this line of work to more general feed-forward networks trained using InfoNCE (Oord et al., 2018).

In a similar vein, Roeder et al. (2020) build on the work of Hyvärinen et al. (2019) to show that for a model family which includes InfoNCE, distribution matching implies parameter matching. In contrast, we associate the learned latent representation with the ground-truth generative factors, showing under what conditions the data generating process is inverted, and thus, the true latent factors are recovered.

3. Theory

We will show a connection between contrastive learning and identifiability in the form of nonlinear ICA. For this, we introduce a feature encoder f that maps observations $\mathbf{x}$ to representations. We consider the widely used *InfoNCE* loss, which often assumes L^2 normalized representations (Wu et al., 2018; He et al., 2020b; Tian et al., 2019; Bachman et al., 2019; Chen et al., 2020a),

$$\mathcal{L}_{\mathsf{contr}}(f;\tau,M) := \tag{1}$$
$$\mathop{\mathbb{E}}_{\substack{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathrm{pos}} \\ \{\mathbf{x}_i^-\}_{i=1}^M \overset{\text{i.i.d.}}{\sim} p_{\mathrm{data}}}} \left[-\log \frac{e^{f(\mathbf{x})^\top f(\tilde{\mathbf{x}})/\tau}}{e^{f(\mathbf{x})^\top f(\tilde{\mathbf{x}})/\tau} + \sum_{i=1}^{M} e^{f(\mathbf{x})^\top f(\mathbf{x}_i^-)/\tau}} \right].$$

Here $M \in \mathbb{Z}_+$ is a fixed number of negative samples, p_{data} is the distribution of all observations and p_{pos} is the distribution of positive pairs. This loss was motivated by the InfoMax principle (Linsker, 1988), and has been shown to be effective by many recent representation learning methods (Logeswaran & Lee, 2018; Wu et al., 2018; Tian et al., 2019; He et al., 2020a; Hjelm et al., 2019; Bachman et al., 2019; Chen et al., 2020a; Baevski et al., 2020b). Our theoretical results also hold for a loss function whose denominator only consists of the second summand across the negative samples (e.g., the SimCLR loss (Chen et al., 2020a)).

In the spirit of existing literature on nonlinear ICA (Hyvärinen & Pajunen, 1999; Harmeling et al., 2003; Sprekeler et al., 2014; Hyvärinen & Morioka, 2016; 2017; Gutmann & Hyvärinen, 2012; Hyvärinen et al., 2019; Khemakhem et al., 2020a), we assume that the observations $\mathbf{x} \in \mathcal{X}$ are generated by an invertible (i.e., injective) generative process $g : \mathcal{Z} \to \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^K$ is the space of observations and $\mathcal{Z} \subseteq \mathbb{R}^N$ with $N \leq K$ denotes the space of latent factors. Influenced by the commonly used feature normalization in InfoNCE, we further assume that $\mathcal{Z}$ is the unit hypersphere $\mathbb{S}^{N-1}$ (see Appx. A.1.1). Additionally, we assume that the ground-truth marginal distribution of the latents of the generative process is uniform and that the conditional distribution (under which positive pairs have high density) is a von Mises-Fisher (vMF) distribution:

$$p(\mathbf{z}) = |\mathcal{Z}|^{-1}, \qquad p(\mathbf{z}|\tilde{\mathbf{z}}) = C_p^{-1} e^{\kappa \mathbf{z}^\top \tilde{\mathbf{z}}} \quad \text{with} \tag{2}$$
$$C_p := \int e^{\kappa \mathbf{z}^\top \tilde{\mathbf{z}}} \, \mathrm{d}\tilde{\mathbf{z}} = \text{const.}, \quad \mathbf{x} = g(\mathbf{z}), \quad \tilde{\mathbf{x}} = g(\tilde{\mathbf{z}}).$$

Given these assumptions, we will show that if f minimizes the contrastive loss $\mathcal{L}_{\mathsf{contr}}$, then f solves the demixing problem, i.e., inverts g up to orthogonal linear transformations.

Our theoretical approach consists of three steps: (1) We demonstrate that $\mathcal{L}_{\mathsf{contr}}$ can be interpreted as the cross-entropy between the (conditional) ground-truth and inferred latent distribution. (2) Next, we show that encoders minimizing $\mathcal{L}_{\mathsf{contr}}$ maintain distance, i.e., two latent vectors with distance α in the ground-truth generative model are mapped to points with the same distance α in the inferred representation. (3) Finally, we leverage distance preservation to show that minimizers of $\mathcal{L}_{\mathsf{contr}}$ invert the generative process up to orthogonal transformations. Detailed proofs are given in Appx. A.1.2.

Additionally, we will present similar results for general convex bodies in $\mathbb{R}^{\mathbb{N}}$ and more general similarity measures, see Sec. 3.3. For this, the detailed proofs are given in Appx. A.2.

3.1. Contrastive learning is related to cross-entropy minimization

From the perspective of nonlinear ICA, we are interested in understanding how the representations $f(\mathbf{x})$ which minimize the contrastive loss $\mathcal{L}_{\mathsf{contr}}$ (defined in Eq. (1)) are related to the ground-truth source signals $\mathbf{z}$. To study this relationship, we focus on the map $h = f \circ g$ between the recovered source signals $h(\mathbf{z})$ and the true source signals $\mathbf{z}$. Note that this is merely for mathematical convenience; it does not necessitate knowledge regarding neither g nor the ground-truth factors during learning (beyond the assumptions stated in the theorems).

A core insight is a connection between the contrastive loss and the cross-entropy between the ground-truth latent distribution and a certain model distribution. For this, we expand the theoretical results obtained by Wang & Isola (2020):

Theorem 1 ($\mathcal{L}_{\mathsf{contr}}$ converges to the cross-entropy between latent distributions)**.** *If the ground-truth marginal distribution p is uniform, then for fixed $\tau > 0$, as the number of negative samples $M \to \infty$, the (normalized) contrastive*

loss converges to

$$\lim_{M\to\infty} \mathcal{L}_{\text{contr}}(f;\tau,M) - \log M + \log|\mathcal{Z}| = \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[H(p(\cdot|\mathbf{z}), q_h(\cdot|\mathbf{z}))\right] \tag{3}$$

where H is the cross-entropy between the ground-truth conditional distribution p over positive pairs and a conditional distribution q_{h} parameterized by the model f,

$$q_{\text{h}}(\tilde{\mathbf{z}}|\mathbf{z}) = C_h(\mathbf{z})^{-1} e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau} \quad \text{with} \quad C_h(\mathbf{z}) := \int e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau}\, \mathrm{d}\tilde{\mathbf{z}}, \tag{4}$$

where $C_h(\mathbf{z}) \in \mathbb{R}^+$ is the partition function of q_{h} (see Appx. A.1.1).

Next, we show that the minimizers h^* of the cross-entropy (4) are isometries in the sense that $\kappa\mathbf{z}^\top\tilde{\mathbf{z}} = h^*(\mathbf{z})^\top h^*(\tilde{\mathbf{z}})$ for all $\mathbf{z}$ and $\tilde{\mathbf{z}}$. In other words, they preserve the dot product between $\mathbf{z}$ and $\tilde{\mathbf{z}}$.

Proposition 1 (Minimizers of the cross-entropy maintain the dot product)**.** *Let $\mathcal{Z} = \mathbb{S}^{N-1}$, $\tau > 0$ and consider the ground-truth conditional distribution of the form $p(\tilde{\mathbf{z}}|\mathbf{z}) = C_p^{-1}\exp(\kappa\tilde{\mathbf{z}}^\top\mathbf{z})$. Let h map onto a hypersphere with radius $\sqrt{\tau\kappa}$.[2] Consider the conditional distribution q_h parameterized by the model, as defined above in Theorem 1, where the hypothesis class for h (and thus f) is assumed to be sufficiently flexible such that $p(\tilde{\mathbf{z}}|\mathbf{z})$ and $q_{\text{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ can match. If h is a minimizer of the cross-entropy $\mathbb{E}_{p(\tilde{\mathbf{z}}|\mathbf{z})}[-\log q_{\text{h}}(\tilde{\mathbf{z}}|\mathbf{z})]$, then $p(\tilde{\mathbf{z}}|\mathbf{z}) = q_{\text{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ and $\forall\mathbf{z},\tilde{\mathbf{z}} : \kappa\mathbf{z}^\top\tilde{\mathbf{z}} = h(\mathbf{z})^\top h(\tilde{\mathbf{z}})$.*

3.2. Contrastive learning identifies ground-truth factors on the hypersphere

From the strong geometric property of isometry, we can now deduce a key property of the minimizers h^*:

Proposition 2 (Extension of the Mazur-Ulam theorem to hyperspheres and the dot product)**.** *Let $\mathcal{Z} = \mathbb{S}^{N-1}$ and $\mathcal{Z}' = \mathbb{S}_r^{N-1}$ be the hyperspheres with radius 1 and $r > 0$, respectively. If $h : \mathbb{R}^N \to \mathcal{Z}'$ is differentiable in the vicinity of $\mathcal{Z}$ and its restriction to $\mathcal{Z}$ maintains the dot product up to a constant factor, i.e., $\forall\mathbf{z},\tilde{\mathbf{z}} \in \mathcal{Z} : r^2\mathbf{z}^\top\tilde{\mathbf{z}} = h(\mathbf{z})^\top h(\tilde{\mathbf{z}})$, then h is an orthogonal linear transformation scaled by r for all $\mathbf{z} \in \mathcal{Z}$.*

In the last step, we combine the previous propositions to derive our main result: the minimizers of the contrastive loss $\mathcal{L}_{\text{contr}}$ solve the demixing problem of nonlinear ICA up to linear transformations, i.e., they identify the original sources $\mathbf{z}$ for observations $g(\mathbf{z})$ up to orthogonal linear transformations. For a hyperspherical space $\mathcal{Z}$ these correspond to combinations of permutations, rotations and sign flips.

[2] Note that in practice this can be implemented as a learnable rescaling operation as the last operation of the network f.

Theorem 2. *Let $\mathcal{Z} = \mathbb{S}^{N-1}$, the ground-truth marginal be uniform, and the conditional a vMF distribution (cf. Eq. 2). Let the restriction of the mixing function g to $\mathcal{Z}$ be injective and h be differentiable in a vicinity of $\mathcal{Z}$. If the assumed form of q_{h}, as defined above, matches that of p, and if f is differentiable and minimizes the CL loss as defined in Eq. (1), then for fixed $\tau > 0$ and $M \to \infty$, $h = f \circ g$ is linear, i.e., f recovers the latent sources up to an orthogonal linear transformation and a constant scaling factor.*

Note that we do not assume knowledge of the ground-truth generative model g; we only make assumptions about the conditional and marginal distribution of the latents. On real data, it is unlikely that the assumed model distribution q_{h} can exactly match the ground-truth conditional. We do, however, provide empirical evidence that h is still an affine transformation even if there is a severe mismatch, see Sec. 4.

3.3. Contrastive learning identifies ground-truth factors on convex bodies in $\mathbb{R}^N$

While the previous theoretical results require $\mathcal{Z}$ to be a hypersphere, we will now show a similar theorem for the more general case of $\mathcal{Z}$ being a convex body in $\mathbb{R}^N$. Note that the hyperrectangle $[a_1, b_1] \times \ldots \times [a_N, b_N]$ is an example of such a convex body.

We follow a similar three step proof strategy as for the hyperspherical case before: (1) We begin again by showing that a properly chosen contrastive loss on convex bodies corresponds to the cross-entropy between the ground-truth conditional and a distribution parametrized by the encoder. For this step, we additionally extend the results of Wang & Isola (2020) to this latent space and loss function. (2) Next, we derive that minimizers of the loss function are isometries of the latent space. Importantly, we do not limit ourselves to a specific metric, thus the result is applicable to a family of contrastive objectives. (3) Finally, we show that these minimizers must be affine transformations. For a special family of conditional distributions (rotationally asymmetric generalized normal distributions (Subbotin, 1923)), we can further narrow the class of solutions to permutations and sign-flips. For the detailed proofs, see Appx. A.2.

As earlier, we assume that the ground-truth marginal distribution of the latents is uniform. However, we now assume that the conditional distribution is exponential:

$$p(\mathbf{z}) = |\mathcal{Z}|^{-1}, \qquad p(\mathbf{z}|\tilde{\mathbf{z}}) = C_p^{-1} e^{-\delta(\mathbf{z},\tilde{\mathbf{z}})} \quad \text{with} \quad C_p(\mathbf{z}) := \int e^{-\delta(\mathbf{z},\tilde{\mathbf{z}})}\, \mathrm{d}\tilde{\mathbf{z}}, \quad \mathbf{x} = g(\mathbf{z}), \quad \tilde{\mathbf{x}} = g(\tilde{\mathbf{z}}), \tag{5}$$

where δ is a metric induced by a norm (see Appx. A.2.1).

To reflect the differences between this conditional distribution and the one assumed for the hyperspherical case, we need to introduce an adjusted version of the contrastive loss

in (1):

Definition 1 ($\mathcal{L}_{\delta\text{-contr}}$ objective)**.** Let $\delta : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ be a metric on $\mathcal{Z}$. We define the general InfoNCE loss, which uses δ as a similarity measure, as

$$\mathcal{L}_{\delta\text{-contr}}(f; \tau, M) := \mathop{\mathbb{E}}_{\substack{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\text{pos}} \\ \{\mathbf{x}_i^-\}_{i=1}^M \overset{\text{i.i.d.}}{\sim} p_{\text{data}}}} \left[-\log \frac{e^{-\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))/\tau}}{e^{-\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))/\tau} + \sum_{i=1}^{M} e^{-\delta(f(\mathbf{x}),f(\mathbf{x}_i^-))/\tau}} \right]. \tag{6}$$

Note that this is a generalization of the InfoNCE criterion in Eq. (1). In contrast to the objective above, the representations are no longer assumed to be L^2 normalized, and the dot-product is replaced with a more general similarity measure δ.

Analogous to the previously demonstrated case for the hypersphere, for convex bodies $\mathcal{Z}$, minimizers of the adjusted $\mathcal{L}_{\delta\text{-contr}}$ objective solve the demixing problem of nonlinear ICA up to invertible linear transformations:

Theorem 5. *Let $\mathcal{Z}$ be a convex body in $\mathbb{R}^N$, $h = f \circ g : \mathcal{Z} \to \mathcal{Z}$, and δ be a metric or a semi-metric (cf. Lemma 1 in Appx. A.2.4), induced by a norm. Further, let the ground-truth marginal distribution be uniform and the conditional distribution be as Eq. (5). Let the mixing function g be differentiable and injective. If the assumed form of q_{h} matches that of p, i.e.,*

$$q_{\text{h}}(\tilde{\mathbf{z}}|\mathbf{z}) = C_q^{-1}(\mathbf{z}) e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau} \quad \text{with} \quad C_q(\mathbf{z}) := \int e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau} \, \mathrm{d}\tilde{\mathbf{z}}, \tag{7}$$

and if f is differentiable and minimizes the $\mathcal{L}_{\delta\text{-contr}}$ objective in Eq. (6) for $M \to \infty$, we find that $h = f \circ g$ is invertible and affine, i.e., we recover the latent sources up to affine transformations.

Note that the model distribution q_{h}, which is implicitly described by the choice of the objective, must be of the same form as the ground-truth distribution p, i.e., both must be based on the same metric. Thus, identifying different ground-truth conditional distributions requires different contrastive $\mathcal{L}_{\delta\text{-contr}}$ objectives. This result can be seen as a generalized version of Theorem 2, as it is valid for any convex body $\mathcal{Z} \subseteq \mathbb{R}^N$, allowing for a larger variety of conditional distributions.

Finally, under the mild restriction that the ground-truth conditional distribution is based on an L^p similarity measure for $p \geq 1, p \neq 2$, h identifies the ground-truth generative factors up to generalized permutations. A generalized permutation matrix $\mathbf{A}$ is a combination of a permutation and element-wise sign-flips, i.e., $\forall \mathbf{z} : (\mathbf{A}\mathbf{z})_i = \alpha_i \mathbf{z}_{\sigma(i)}$ with $\alpha_i = \pm 1$ and σ being a permutation.

Theorem 6. *Let $\mathcal{Z}$ be a convex body in $\mathbb{R}^N$, $h : \mathcal{Z} \to \mathcal{Z}$, and δ be an L^α metric or semi-metric (cf. Lemma 1 in Appx. A.2.4) for $\alpha \geq 1, \alpha \neq 2$. Further, let the ground-truth marginal distribution be uniform and the conditional distribution be as Eq. (5), and let the mixing function g be differentiable and invertible. If the assumed form of $q_{\text{h}}(\cdot|\mathbf{z})$ matches that of $p(\cdot|\mathbf{z})$, i.e., both use the same metric δ up to a constant scaling factor, and if f is differentiable and minimizes the $\mathcal{L}_{\delta\text{-contr}}$ objective in Eq. (6) for $M \to \infty$, we find that $h = f \circ g$ is a composition of input independent permutations, sign flips and rescaling.*

4. Experiments

4.1. Validation of theoretical claim

We validate our theoretical claims under both perfectly matching and violated conditions regarding the ground-truth marginal and conditional distributions. We consider source signals of dimensionality $N = 10$, and sample pairs of source signals in two steps: First, we sample from the marginal $p(\mathbf{z})$. For this, we consider both uniform distributions which match our assumptions and non-uniform distributions (e.g., a normal distribution) which violate them. Second, we generate the positive pair by sampling from a conditional distribution $p(\tilde{\mathbf{z}}|\mathbf{z})$. Here, we consider matches with our assumptions on the conditional distribution (von Mises-Fisher for $\mathcal{Z} = \mathbb{S}^{N-1}$) as well as violations (e.g. normal, Laplace or generalized normal distribution for $\mathcal{Z} = \mathbb{S}^{N-1}$). Further, we consider spaces beyond the hypersphere, such as the bounded box (which is a convex body) and the unbounded $\mathbb{R}^N$.

We generate the observations with a multi-layer perceptron (MLP), following previous work (Hyvärinen & Morioka, 2016; 2017). Specifically, we use three hidden layers with leaky ReLU units and random weights; to ensure that the MLP g is invertible, we control the condition number of the weight matrices. For our feature encoder f, we also use an MLP with leaky ReLU units, where the assumed space is denoted by the normalization, or lack thereof, of the encoding. Namely, for the hypersphere (denoted as *Sphere*) and the hyperrectangle (denoted as *Box*) we apply an L^2 and L^∞ normalization, respectively. For flexibility in practice, we parameterize the normalization magnitude of the *Box*, including it as part of the encoder's learnable parameters. On the hypersphere we optimize $\mathcal{L}_{\text{contr}}$ and on the hyperrectangle as well as the unbounded space we optimize $\mathcal{L}_{\delta\text{-contr}}$. For further details, see Appx. A.3.

To test for identifiability up to affine transformations, we fit a linear regression between the ground-truth and recovered sources and report the coefficient of determination (R^2). To test for identifiability up to generalized permutations, we leverage the mean correlation coefficient (MCC), as used

in previous work (Hyvärinen & Morioka, 2016; 2017). For further details, see Appx. A.3.

We evaluate both identifiability metrics for three different model types. First, we ensure that the problem requires nonlinear demixing by considering the identity function for model f, which amounts to scoring the observations against the sources (**Identity Model**). Second, we ensure that the problem is solvable within our model class by training our model f with supervision, minimizing the mean-squared error between $f(g(\mathbf{z}))$ and $\mathbf{z}$ (**Supervised Model**). Third, we fit our model without supervision using a contrastive loss (**Unsupervised Model**).

Tables 1 and 2 show results evaluating identifiability up to affine transformations and generalized permutations, respectively. When assumptions match (see column M.), CL recovers a score close to the empirical upper bound. Mismatches in assumptions on the marginal and conditional do not lead to a significant drop in performance with respect to affine identifiability, but do for permutation identifiability compared to the empirical upper bound. In many practical scenarios, we use the learned representations to solve a downstream task, thus, identifiability up to affine transformations is often sufficient. However, for applications where identification of the individual generative factors is desirable, some knowledge of the underlying generative process is required to choose an appropriate loss function and feature normalization. Interestingly, we find that for convex bodies, we obtain identifiability up to permutation even in the case of a normal conditional, which likely is due to the axis-aligned box geometry of the latent domain. Finally, note that the drop in performance for identifiability up to permutations in the last group of Tab. 2 is a natural consequence of either the ground-truth or the assumed conditional being rotationally symmetric, e.g., a normal distribution, in an unbounded space. Here, rotated versions of the latent space are indistinguishable and, thus, the model cannot align the axes of the reconstruction with that of the ground-truth latent space, resulting in a lower score.

To zoom in on how violations of the uniform marginal assumption influence the identifiability achieved by a model in practice, we perform an ablation on the marginal distribution by interpolating between the theoretically assumed uniform distribution and highly locally concentrated distributions. In particular, we consider two cases: (1) a sphere ($\mathcal{S}^9$) with a vMF marginal around its north pole for different concentration parameters κ; (2) a box ($[0,1]^{10}$) with a normal marginal around the box's center for different standard deviations σ. For both cases, Fig. 2 shows the R^2 score as a function of the concentration κ and $1/\sigma^2$ respectively (black). As a reference, the concentration of the used conditional distribution is highlighted as a dashed line. In addition, we also display the probability mass (0–100%)

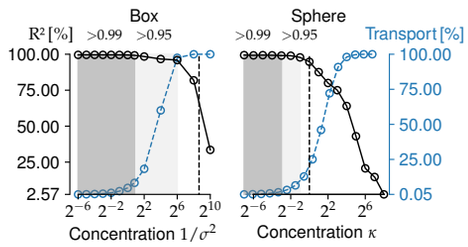


Figure 2. Varying degrees of violation of the uniformity assumption for the marginal distribution. The figure shows the R^2 score measuring identifiability up to linear transformations (black) as well as the difference between the used marginal and assumed uniform distribution in terms of probability mass (blue) as a function of the marginal's concentration. The black dotted line indicates the concentration of the used conditional distribution.

that needs to be moved for converting the used marginal distribution (i.e., vMF or normal) into the assumed uniform marginal distribution (blue) as an intuitive measure of the mismatch (i.e., $\frac{1}{2}\int |p(\mathbf{z}) - p_{\text{uni}}| \, \mathrm{d}\mathbf{z}$). While, we observe significant robustness to mismatch, in both cases, we see performance drop drastically once the marginal distribution is more concentrated than the conditional distribution of positive pairs. In such scenarios, positive pairs are indistinguishable from negative pairs.

4.2. Extensions to image data

Previous studies have demonstrated that representation learning using constrastive learning scales well to complex natural image data (Chen et al., 2020a;b; Hénaff, 2020). Unfortunately, the true generative factors of natural images are inaccessible, thus we cannot evaluate identifiability scores.

We consider two alternatives. First, we evaluate on the recently proposed benchmark *KITTI Masks* (Klindt et al., 2021), which is composed of segmentation masks of natural videos. Second, we contribute a novel benchmark (*3DIdent*; cf. Fig. 3) which features aspects of natural scenes, e.g. a complex 3D object and different lighting conditions, while still providing access to the continuous ground-truth factors. For further details, see Appx. A.4.1. *3DIdent* is available at zenodo.org/record/4502485.

4.2.1. KITTI Masks

KITTI Masks (Klindt et al., 2021) is composed of pedestrian segmentation masks extracted from an autonomous driving vision benchmark KITTI-MOTS (Geiger et al., 2012), with natural shapes and continuous natural transitions. We compare to SlowVAE (Klindt et al., 2021), the state-of-the-art on the considered dataset. In our experiments, we use the same training hyperparameters (for details see Appx. A.3) and (encoder) architecture as Klindt et al. (2021). The positive

Contrastive Learning Inverts the Data Generating Process

Table 1. Identifiability up to affine transformations. Mean ± standard deviation over 5 random seeds. Note that only the first row corresponds to a setting that matches (✓) our theoretical assumptions, while the others show results for violated assumptions (✗; see column *M.*). Note that the identity score only depends on the ground-truth space and the marginal distribution defined for the generative process, while the supervised score additionally depends on the space assumed by the model.

Generative process g			Model f			R^2 Score [%]		
Space	$p(\cdot)$	$p(\cdot\|\cdot)$	Space	$q_\mathrm{h}(\cdot\|\cdot)$	M.	Identity	Supervised	Unsupervised
Sphere	Uniform	vMF(κ=1)	Sphere	vMF(κ=1)	✓	66.98 ± 2.79	99.71 ± 0.05	99.42 ± 0.05
Sphere	Uniform	vMF(κ=10)	Sphere	vMF(κ=1)	✗	—"—	—"—	99.86 ± 0.01
Sphere	Uniform	Laplace(λ=0.05)	Sphere	vMF(κ=1)	✗	—"—	—"—	99.91 ± 0.01
Sphere	Uniform	Normal(σ=0.05)	Sphere	vMF(κ=1)	✗	—"—	—"—	99.86 ± 0.00
Box	Uniform	Normal(σ=0.05)	Unbounded	Normal	✗	67.93 ± 7.40	99.78 ± 0.06	99.60 ± 0.02
Box	Uniform	Laplace(λ=0.05)	Unbounded	Normal	✗	—"—	—"—	99.64 ± 0.02
Box	Uniform	Laplace(λ=0.05)	Unbounded	GenNorm(β=3)	✗	—"—	—"—	99.70 ± 0.02
Box	Uniform	Normal(σ=0.05)	Unbounded	GenNorm(β=3)	✗	—"—	—"—	99.69 ± 0.02
Sphere	Normal(σ=1)	Laplace(λ=0.05)	Sphere	vMF(κ=1)	✗	63.37 ± 2.41	99.70 ± 0.07	99.02 ± 0.01
Sphere	Normal(σ=1)	Normal(σ=0.05)	Sphere	vMF(κ=1)	✗	—"—	—"—	99.02 ± 0.02
Unbounded	Laplace(λ=1)	Normal(σ=1)	Unbounded	Normal	✗	62.49 ± 1.65	99.65 ± 0.04	98.13 ± 0.14
Unbounded	Normal(σ=1)	Normal(σ=1)	Unbounded	Normal	✗	63.57 ± 2.30	99.61 ± 0.17	98.76 ± 0.03

Table 2. Identifiability up to generalized permutations, averaged over 5 runs. Note that while Theorem 6 requires the model latent space to be a convex body and $p(\cdot|\cdot) = q_\mathrm{h}(\cdot|\cdot)$, we find that empirically either is sufficient. The results are grouped in four blocks corresponding to different types and degrees of violation of assumptions of our theory showing identifiability up to permutations: (1) no violation, violation of the assumptions on either the (2) space or (3) the conditional distribution, or (4) both.

Generative process g			Model f			MCC Score [%]		
Space	$p(\cdot)$	$p(\cdot\|\cdot)$	Space	$q_\mathrm{h}(\cdot\|\cdot)$	M.	Identity	Supervised	Unsupervised
Box	Uniform	Laplace(λ=0.05)	Box	Laplace	✓	46.55 ± 1.34	99.93 ± 0.03	98.62 ± 0.05
Box	Uniform	GenNorm(β=3; λ=0.05)	Box	GenNorm(β=3)	✓	—"—	—"—	99.90 ± 0.06
Box	Uniform	Normal(σ=0.05)	Box	Normal	✗	—"—	—"—	99.77 ± 0.01
Box	Uniform	Laplace(λ=0.05)	Box	Normal	✗	—"—	—"—	99.76 ± 0.02
Box	Uniform	GenNorm(β=3; λ=0.05)	Box	Laplace	✗	—"—	—"—	98.80 ± 0.02
Box	Uniform	Laplace(λ=0.05)	Unbounded	Laplace	✗	—"—	99.97 ± 0.03	98.57 ± 0.02
Box	Uniform	GenNorm(β=3; λ=0.05)	Unbounded	GenNorm(β=3)	✗	—"—	—"—	99.85 ± 0.01
Box	Uniform	Normal(σ=0.05)	Unbounded	Normal	✗	—"—	—"—	58.26 ± 3.00
Box	Uniform	Laplace(λ=0.05)	Unbounded	Normal	✗	—"—	—"—	59.67 ± 2.33
Box	Uniform	Normal(σ=0.05)	Unbounded	GenNorm(β=3)	✗	—"—	—"—	43.80 ± 2.15

pairs consist of nearby frames with a time separation $\overline{\Delta t}$.

As argued and shown in Klindt et al. (2021), the transitions in the ground-truth latents between nearby frames is sparse. Unsurprisingly then, Table 3 shows that assuming a Laplace conditional as opposed to a normal conditional in the contrastive loss leads to better identification of the underlying factors of variation. SlowVAE also assumes a Laplace conditional (Klindt et al., 2021) but appears to struggle if the frames of a positive pair are too similar ($\overline{\Delta t} = 0.05s$). This degradation in performance is likely due to the limited expressiveness of the decoder deployed in SlowVAE.

Table 3. **KITTI Masks**. Mean ± standard deviation over 10 random seeds. $\overline{\Delta t}$ indicates the average temporal distance of frames used.

	Model	Model Space	MCC [%]
$\overline{\Delta t} = 0.05s$	SlowVAE	Unbounded	66.1 ± 4.5
	Laplace	Unbounded	77.1 ± 1.0
	Laplace	Box	74.1 ± 4.4
	Normal	Unbounded	58.3 ± 5.4
	Normal	Box	59.9 ± 5.5
$\overline{\Delta t} = 0.15s$	SlowVAE	Unbounded	79.6 ± 5.8
	Laplace	Unbounded	79.4 ± 1.9
	Laplace	Box	80.9 ± 3.8
	Normal	Unbounded	60.2 ± 8.7
	Normal	Box	68.4 ± 6.7

4.2.2. 3DIDENT

Dataset description We build on (Johnson et al., 2017b) and use the Blender rendering engine (Blender Online Com-

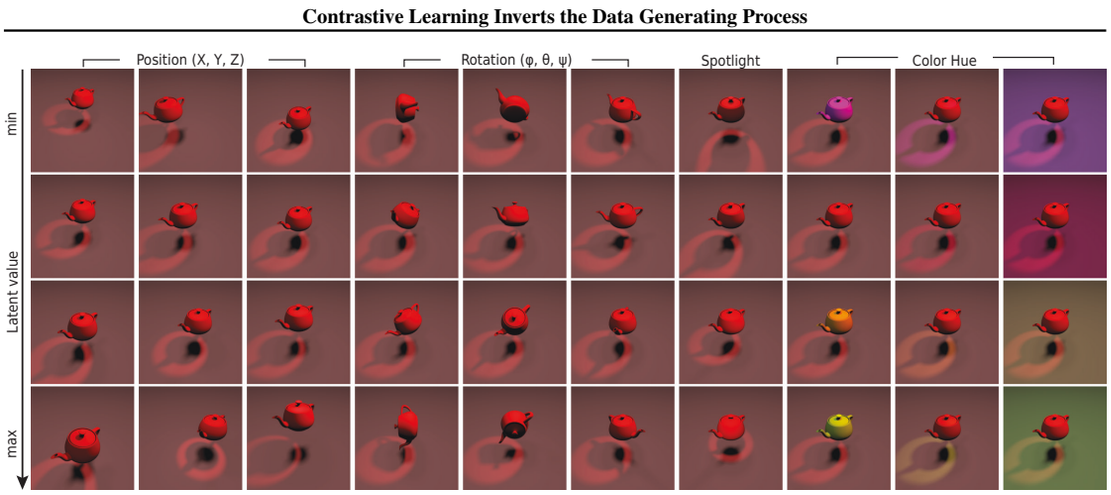


Figure 3. **3DIdent**. Influence of the latent factors $\mathbf{z}$ on the renderings $\mathbf{x}$. Each column corresponds to a traversal in one of the ten latent dimensions while the other dimensions are kept fixed.

munity, 2021) to create visually complex 3D images (see Fig. 3). Each image in the dataset shows a colored 3D object which is located and rotated above a colored ground in a 3D space. Additionally, each scene contains a colored spotlight focused on the object and located on a half-circle around the scene. The observations are encoded with an RGB color space, and the spatial resolution is 224×224 pixels.

The images are rendered based on a 10-dimensional latent, where: (1) three dimensions describe the XYZ position, (2) three dimensions describe the rotation of the object in Euler angles, (3) two dimensions describe the color of the object and the ground of the scene, respectively, and (4) two dimensions describe the position and color of the spotlight. We use the HSV color space to describe the color of the object and the ground with only one latent each by having the latent factor control the hue value. For more details on the dataset see Sec. A.4.

The dataset contains 250 000 observation-latent pairs where the latents are uniformly sampled from the hyperrectangle $\mathcal{Z}$. To sample positive pairs $(\mathbf{z}, \tilde{\mathbf{z}})$ we first sample a value $\tilde{\mathbf{z}}'$ from the data conditional $p(\tilde{\mathbf{z}}'|\mathbf{z})$, and then use nearest-neighbor matching[3] implemented by FAISS (Johnson et al., 2017a) to find the latent $\tilde{\mathbf{z}}$ closest to $\tilde{\mathbf{z}}'$ (in L^2 distance) for which there exists an image rendering. In addition, unlike previous work (Locatello et al., 2019), we create a hold-out test set with 25 000 distinct observation-latent pairs.

Experiments and Results We train a convolutional feature encoder f composed of a ResNet18 architecture (He et al., 2016) and an additional fully-connected layer, with a LeakyReLU nonlinearity as the hidden activation. For more details, see Appx. A.3. Following the same methodology as in Sec. 4.1, i) depending on the assumed space, the output of the feature encoder is normalized accordingly and ii) in addition to the CL models, we also train a supervised model to serve as an upper bound on performance. We consider normal and Laplace distributions for positive pairs. Note, that due to the finite dataset size we only sample from an approximation of these distributions.

As in Tables 1 and 2, the results in Table 4 demonstrate that CL reaches scores close to the topline (supervised) performance, and mismatches between the assumed and ground-truth conditional distribution do not harm the performance significantly. However, if the hypothesis class of the encoder is too restrictive to model the ground-truth conditional distribution, we observe a clear drop in performance, i.e., mapping a box onto a sphere. Note, that this corresponds to the InfoNCE objective for L^2-normalized representations, commonly used for self-supervised representation learning (Wu et al., 2018; He et al., 2020b; Tian et al., 2019; Bachman et al., 2019; Chen et al., 2020a). Finally, the last result shows that leveraging image augmentations (Chen et al., 2020a) as opposed to sampling from a specified conditional distribution of positive pairs $p(\cdot|\cdot)$ results in a performance drop. For details on the experiment, see Appx. Sec. A.3. We explain this with the greater mismatch between the conditional distribution assumed by the model and the conditional distribution induced by the augmentations. In all, we demonstrate validation of our theoretical claims even for generative processes with higher visual complexity than those considered in Sec. 4.1.

[3] We used an Inverted File Index (IVF) with Hierarchical Navigable Small World (HNSW) graph exploration for fast indexing.

Table 4. Identifiability up to affine transformations on the test set of 3DIdent. Mean ± standard deviation over 3 random seeds. As earlier, only the first row corresponds to a setting that matches the theoretical assumptions for linear identifiability; the others show distinct violations. Supervised training with unbounded space achieves scores of $R^2 = (98.67 \pm 0.03)\%$ and MCC $= (99.33 \pm 0.01)\%$. The last row refers to using the image augmentations suggested by Chen et al. (2020a) to generate positive image pairs. For performance on the training set, see Appx. Table 5.

Dataset	Model f			Identity [%]	Unsupervised [%]	
$p(\cdot\|\cdot)$	Space	$q_{\mathrm{h}}(\cdot\|\cdot)$	M.	R^2	R^2	MCC
Normal	Box	Normal	✓	5.25 ± 1.20	96.73 ± 0.10	98.31 ± 0.04
Normal	Unbounded	Normal	✗	—"—	96.43 ± 0.03	54.94 ± 0.02
Laplace	Box	Normal	✗	—"—	96.87 ± 0.08	98.38 ± 0.03
Normal	Sphere	vMF	✗	—"—	65.74 ± 0.01	42.44 ± 3.27
Augm.	Sphere	vMF	✗	—"—	45.51 ± 1.43	46.34 ± 1.59

5. Conclusion

We showed that objectives belonging to the InfoNCE family, the basis for a number of state-of-the-art techniques in self-supervised representation learning, can uncover the true generative factors of variation underlying the observational data. To succeed, these objectives implicitly encode a few weak assumptions about the statistical nature of the underlying generative factors. While these assumptions will likely not be exactly matched in practice, we showed empirically that the underlying factors of variation are identified even if theoretical assumptions are severely violated.

Our theoretical and empirical results suggest that the representations found with contrastive learning implicitly (and approximately) invert the generative process of the data. This could explain why the learned representations are so useful in many downstream tasks. It is known that a decisive aspect of contrastive learning is the right choice of augmentations that form a positive pair. We hope that our framework might prove useful for clarifying the ways in which certain augmentations affect the learned representations, and for finding improved augmentation schemes.

Furthermore, our work opens avenues for constructing more effective contrastive losses. As we demonstrate, imposing a contrastive loss informed by characteristics of the latent space can considerably facilitate inferring the correct semantic descriptors, and thus boost performance in downstream tasks. While our framework already allows for a variety of *conditional* distributions, it is an interesting open question how to adapt it to *marginal* distributions beyond the uniform implicitly encoded in InfoNCE. Also, future work may extend our theoretical framework by incorporating additional assumptions about our visual world, such as compositionality, hierarchy or objectness. Accounting for such inductive biases holds enormous promise in forming the basis for the next generation of self-supervised learning algorithms.

Taken together, we lay a strong theoretical foundation for not only understanding but extending the success of state-of-the-art self-supervised learning techniques.

Author contributions

The project was initiated by WB. RSZ, StS and WB jointly derived the theory. RSZ and YS implemented and executed the experiments. The 3DIdent dataset was created by RSZ with feedback from StS, YS, WB and MB. RSZ, YS, StS and WB contributed to the final version of the manuscript.

Acknowledgements

We thank Muhammad Waleed Gondal, Ivan Ustyuzhaninov, David Klindt, Lukas Schott, Luisa Eck, and Kartik Ahuja for helpful discussions. We thank Bozidar Antic, Shubham Krishna and Jugoslav Stojcheski for ideas regarding the design of 3DIdent. We thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting RSZ, YS and StS. StS acknowledges his membership in the European Laboratory for Learning and Intelligent Systems (ELLIS) PhD program. We acknowledge support from the German Federal Ministry of Education and Research (BMBF) through the Competence Center for Machine Learning (TUE.AI, FKZ 01IS18039A) and the Bernstein Computational Neuroscience Program Tübingen (FKZ: 01GQ1002). WB acknowledges support via his Emmy Noether Research Group funded by the German Science Foundation (DFG) under grant no. BR 6382/1-1 as well as support by Open Philantropy and the Good Ventures Foundation. MB and WB acknowledge funding from the MICrONS program of the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior/Interior Business Center (DoI/IBC) contract number D16PC00003.

References

Bachman, P., Hjelm, R. D., and Buchwalter, W. Learning representations by maximizing mutual information across views. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 15509–15519, 2019.

Baevski, A., Schneider, S., and Auli, M. vq-wav2vec: Self-supervised learning of discrete speech representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020a.

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b.

Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Blender Institute, Amsterdam, 2021.

Burgess, C. and Kim, H. 3d shapes dataset. https://github.com/deepmind/3dshapes-dataset/, 2018.

Całka, A. Local isometries of compact metric spaces. *Proceedings of the American Mathematical Society*, 85(4): 643–647, 1982.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. E. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 2020a.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. E. Big self-supervised models are strong semi-supervised learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b.

Chuang, C., Robinson, J., Lin, Y., Torralba, A., and Jegelka, S. Debiased contrastive learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Li, F. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pp. 248–255. IEEE Computer Society, 2009. doi: 10.1109/CVPR.2009.5206848.

Dittadi, A., Träuble, F., Locatello, F., Wüthrich, M., Agrawal, V., Winther, O., Bauer, S., and Schölkopf, B. On the transfer of disentangled representations in realistic settings. *International Conference on Learning Representations (ICLR)*, 2021.

Geiger, A., Lenz, P., and Urtasun, R. Are we ready for autonomous driving? the KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pp. 3354–3361. IEEE Computer Society, 2012. doi: 10.1109/CVPR.2012.6248074.

Gondal, M. W., Wuthrich, M., Miladinovic, D., Locatello, F., Breidt, M., Volchkov, V., Akpo, J., Bachem, O., Schölkopf, B., and Bauer, S. On the transfer of inductive bias from simulation to the real world: a new disentanglement dataset. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 15714–15725, 2019.

Gutmann, M. U. and Hyvärinen, A. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *The Journal of Machine Learning Research*, 13:307–361, 2012.

Harmeling, S., Ziehe, A., Kawanabe, M., and Müller, K.-R. Kernel-based nonlinear blind source separation. *Neural Computation*, 15(5):1089–1124, 2003.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778. IEEE Computer Society, 2016. doi: 10.1109/CVPR.2016.90.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 9726–9735. IEEE, 2020a. doi: 10.1109/CVPR42600.2020.00975.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. B. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 9726–9735. IEEE, 2020b. doi: 10.1109/CVPR42600.2020.00975.

Hénaff, O. J. Data-efficient image recognition with contrastive predictive coding. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4182–4192. PMLR, 2020.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Hyvärinen, A. and Morioka, H. Unsupervised feature extraction by time-contrastive learning and nonlinear ICA. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 3765–3773, 2016.

Hyvärinen, A. and Morioka, H. Nonlinear ICA of temporally dependent stationary sources. In Singh, A. and Zhu, X. J. (eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pp. 460–469. PMLR, 2017.

Hyvärinen, A. and Pajunen, P. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3):429–439, 1999.

Hyvärinen, A., Karhunen, J., and Oja, E. *Independent Component Analysis*. Wiley Interscience, 2001.

Hyvärinen, A., Sasaki, H., and Turner, R. E. Nonlinear ICA using auxiliary variables and generalized contrastive learning. In Chaudhuri, K. and Sugiyama, M. (eds.), *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pp. 859–868. PMLR, 2019.

Johnson, J., Douze, M., and Jégou, H. Billion-scale similarity search with gpus. *arXiv preprint arXiv:1702.08734*, 2017a.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. B. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 1988–1997. IEEE Computer Society, 2017b. doi: 10.1109/CVPR.2017.215.

Jutten, C., Babaie-Zadeh, M., and Karhunen, J. Nonlinear mixtures. *Handbook of Blind Source Separation, Independent Component Analysis and Applications*, pp. 549–592, 2010.

Khemakhem, I., Kingma, D. P., Monti, R. P., and Hyvärinen, A. Variational autoencoders and nonlinear ICA: A unifying framework. In Chiappa, S. and Calandra, R. (eds.), *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2207–2217. PMLR, 2020a.

Khemakhem, I., Monti, R. P., Kingma, D. P., and Hyvärinen, A. Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ICA. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Klindt, D., Schott, L., Sharma, Y., Ustyuzhaninov, I., Brendel, W., Bethge, M., and Paiton, D. Towards nonlinear disentanglement in natural data with temporal sparse coding. *International Conference on Learning Representations (ICLR)*, 2021.

Lamperti, J. et al. On the isometries of certain function-spaces. *Pacific J. Math*, 8(3):459–466, 1958.

Lee, J. M. Smooth manifolds. In *Introduction to Smooth Manifolds*, pp. 606–607. Springer, 2013.

Li, C.-K. and So, W. Isometries of ℓ_p-norm. *The American Mathematical Monthly*, 101(5):452–453, 1994.

Linsker, R. Self-organization in a perceptual network. *Computer*, 21(3):105–117, 1988.

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled

representations. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124. PMLR, 2019.

Locatello, F., Poole, B., Rätsch, G., Schölkopf, B., Bachem, O., and Tschannen, M. Weakly-supervised disentanglement without compromises. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 6348–6359. PMLR, 2020.

Logeswaran, L. and Lee, H. An efficient framework for learning sentence representations. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

Mankiewicz, P. Extension of isometries in normed linear spaces. *Bulletin de l'Academie polonaise des sciences: Serie des sciences mathematiques, astronomiques et physiques*, 20(5):367–+, 1972.

Newell, M. E. *The Utilization of Procedure Models in Digital Image Synthesis.* PhD thesis, The University of Utah, 1975. AAI7529894.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Ravanelli, M., Zhong, J., Pascual, S., Swietojanski, P., Monteiro, J., Trmal, J., and Bengio, Y. Multi-task self-supervised learning for robust speech recognition. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2020, Barcelona, Spain, May 4-8, 2020*, pp. 6989–6993. IEEE, 2020. doi: 10.1109/ICASSP40776.2020.9053569.

Robinson, J., Chuang, C.-Y., Sra, S., and Jegelka, S. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*, 2020.

Roeder, G., Metz, L., and Kingma, D. P. On linear identifiability of learned representations. *arXiv preprint arXiv:2007.00810*, 2020.

Ruzhansky, M. and Sugimoto, M. On global inversion of homogeneous maps. *Bulletin of Mathematical Sciences*, 5(1):13–18, 2015.

Saunshi, N., Plevrakis, O., Arora, S., Khodak, M., and Khandeparkar, H. A theoretical analysis of contrastive unsupervised representation learning. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5628–5637. PMLR, 2019.

Schneider, S., Baevski, A., Collobert, R., and Auli, M. wav2vec: Unsupervised pre-training for speech recognition. *CoRR*, abs/1904.05862, 2019.

Sprekeler, H., Zito, T., and Wiskott, L. An extension of slow feature analysis for nonlinear blind source separation. *The Journal of Machine Learning Research*, 15(1):921–947, 2014.

Subbotin, M. F. On the law of frequency of error. *Mat. Sb.*, 31(2):296–301, 1923.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning, 2020.

Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. On mutual information maximization for representation learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9929–9939. PMLR, 2020.

Wu, M., Zhuang, C., Yamins, D., and Goodman, N. On the importance of views in unsupervised representation learning. 2020.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 3733–3742. IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00393.

A. Appendix

A.1. Extended Theory for Hyperspheres

A.1.1. Assumptions

Generative Process Let the generator $g : \mathbb{R}^N \to \mathcal{X}$ with $\mathcal{X} \subseteq \mathbb{R}^K$ and $K \geq N$. Further, let the restriction of g to the space $\mathcal{Z} = \mathbb{S}^{N-1} \subset \mathbb{R}^N$ be injective and g be differentiable in the vicinity of $\mathcal{Z}$. We assume that the marginal distribution $p(\mathbf{z})$ over latent variables $\mathbf{z} \in \mathcal{Z}$ is uniform:

$$p(\mathbf{z}) = \frac{1}{|\mathcal{Z}|}. \tag{8}$$

Further, we assume that the conditional distribution over positive pairs $p(\tilde{\mathbf{z}}|\mathbf{z})$ is a von Mises-Fisher (vMF) distribution

$$p(\tilde{\mathbf{z}}|\mathbf{z}) = C_p^{-1} e^{\kappa \mathbf{z}^\top \tilde{\mathbf{z}}} \tag{9}$$

$$\text{with} \quad C_p := \int e^{\kappa \boldsymbol{\eta}^\top \tilde{\mathbf{z}}} \, \mathrm{d}\tilde{\mathbf{z}}, \tag{10}$$

where κ is a parameter controlling the width of the distribution and $\boldsymbol{\eta}$ is any vector on the hypersphere. Finally, we assume that during training one has access to observations $\mathbf{x}$, which are samples from these distributions transformed by the generator function g.

Model Let $f : \mathcal{X} \to \mathbb{S}_r^{N-1}$, where $\mathbb{S}_r^{N-1}$ denotes a hypersphere with radius r. The parameters of this model are optimized using contrastive learning. We associate a conditional distribution $q_\mathrm{h}(\tilde{\mathbf{z}}|\mathbf{z})$ with our model f through $h = f \circ g$ and

$$q_\mathrm{h}(\tilde{\mathbf{z}}|\mathbf{z}) = C_q^{-1}(\mathbf{z}) e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau}$$
$$\text{with} \quad C_q(\mathbf{z}) := \int e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau} \, \mathrm{d}\tilde{\mathbf{z}}, \tag{11}$$

where $C_q(\mathbf{z})$ is the partition function and $\tau > 0$ is a scale parameter.

A.1.2. Proofs for Sec. 3

We begin by recalling a result of Wang & Isola (2020), where the authors show an asymptotic relation between the contrastive loss $\mathcal{L}_\mathsf{contr}$ and two loss functions, the *alignment* loss $\mathcal{L}_\mathsf{align}$ and the *uniformity* loss $\mathcal{L}_\mathsf{uni}$:

Proposition A (Asymptotics of $\mathcal{L}_\mathsf{contr}$, Wang & Isola, 2020)**.** *For fixed $\tau > 0$, as the number of negative samples $M \to \infty$, the (normalized) contrastive loss converges to*

$$\lim_{M\to\infty} \mathcal{L}_\mathsf{contr}(f;\tau,M) - \log M = \mathcal{L}_\mathsf{align}(f;\tau) + \mathcal{L}_\mathsf{uni}(f;\tau), \tag{12}$$

where

$$\mathcal{L}_\mathsf{align}(f;\tau) := -\frac{1}{\tau} \mathop{\mathbb{E}}_{(\tilde{\mathbf{z}},\mathbf{z})\sim p(\tilde{\mathbf{z}},\mathbf{z})} \left[(f\circ g)(\mathbf{z})^\mathsf{T}(f\circ g)(\mathbf{z})\right]$$
$$\mathcal{L}_\mathsf{uni}(f;\tau) := \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[\log \mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})} \left[e^{(f\circ g)(\tilde{\mathbf{z}})^\mathsf{T}(f\circ g)(\mathbf{z})/\tau}\right]\right]. \tag{13}$$

Proof. See Theorem 1 of Wang & Isola (2020). Note that they originally formulated the losses in terms of observations $\mathbf{x}$ and not in terms of the latent variables $\mathbf{z}$. However, this modified version simplifies notation in the following. □

Based on this result, we show that the contrastive loss $\mathcal{L}_\mathsf{contr}$ asymptotically converges to the cross-entropy between the ground-truth conditional p and our assumed model conditional distribution q_h, up to a constant. This is notable, because given the correct model specification for q_h, it is well-known that the cross-entropy is minimized iff $q_h = p$, i.e., the ground-truth conditional distribution and the model distribution will match.

Theorem 1 ($\mathcal{L}_\mathsf{contr}$ converges to the cross-entropy between latent distributions)**.** *If the ground-truth marginal distribution p is uniform, then for fixed $\tau > 0$, as the number of negative samples $M \to \infty$, the (normalized) contrastive loss converges to*

$$\lim_{M\to\infty} \mathcal{L}_\mathsf{contr}(f;\tau,M) - \log M + \log|\mathcal{Z}| = \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[H(p(\cdot|\mathbf{z}), q_h(\cdot|\mathbf{z}))\right] \tag{14}$$

where H is the cross-entropy between the ground-truth conditional distribution p over positive pairs and a conditional distribution q_h parameterized by the model f, and $C_h(\mathbf{z}) \in \mathbb{R}^+$ is the partition function of q_h (see Appendix A.1.1):

$$q_\mathrm{h}(\tilde{\mathbf{z}}|\mathbf{z}) = C_h(\mathbf{z})^{-1} e^{h(\tilde{\mathbf{z}})^\mathsf{T} h(\mathbf{z})/\tau}$$
$$\textit{with} \quad C_h(\mathbf{z}) := \int e^{h(\tilde{\mathbf{z}})^\mathsf{T} h(\mathbf{z})/\tau} \, \mathrm{d}\tilde{\mathbf{z}}. \tag{15}$$

Proof. The cross-entropy between the conditional distributions p and q_h is given by

$$\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[H(p(\cdot|\mathbf{z}), q_h(\cdot|\mathbf{z}))\right] \tag{16}$$
$$= \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})} \left[-\log q_h(\tilde{\mathbf{z}}|\mathbf{z})\right]\right] \tag{17}$$
$$= \mathop{\mathbb{E}}_{\tilde{\mathbf{z}},\mathbf{z}\sim p(\tilde{\mathbf{z}},\mathbf{z})} \left[-\frac{1}{\tau} h(\tilde{\mathbf{z}})^\top h(\mathbf{z}) + \log C_h(\mathbf{z})\right] \tag{18}$$
$$= -\frac{1}{\tau} \mathop{\mathbb{E}}_{\tilde{\mathbf{z}},\mathbf{z}\sim p(\tilde{\mathbf{z}},\mathbf{z})} \left[h(\tilde{\mathbf{z}})^\top h(\mathbf{z})\right] + \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[\log C_h(\mathbf{z})\right]. \tag{19}$$

Using the definition of C_h in Eq. (15) we obtain

$$= -\frac{1}{\tau}\mathop{\mathbb{E}}_{\tilde{\mathbf{z}},\mathbf{z}\sim p(\tilde{\mathbf{z}},\mathbf{z})}\left[h(\tilde{\mathbf{z}})^\top h(\mathbf{z})\right] \tag{20}$$

$$+\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log\int_{\mathcal{Z}} e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau}\,\mathrm{d}\tilde{\mathbf{z}}\right]. \tag{21}$$

By assumption the marginal distribution is uniform, i.e., $p(\mathbf{z}) = |\mathcal{Z}|^{-1}$. We expand by $|\mathcal{Z}||\mathcal{Z}|^{-1}$ and estimate the integral by sampling from $p(\mathbf{z}) = |\mathcal{Z}|^{-1}$, yielding

$$= -\frac{1}{\tau}\mathop{\mathbb{E}}_{\tilde{\mathbf{z}},\mathbf{z}\sim p(\tilde{\mathbf{z}},\mathbf{z})}\left[h(\tilde{\mathbf{z}})^\top h(\mathbf{z})\right] \tag{22}$$

$$+\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log|\mathcal{Z}|\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})}\left[e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau}\right]\right] \tag{23}$$

$$= -\frac{1}{\tau}\mathop{\mathbb{E}}_{\tilde{\mathbf{z}},\mathbf{z}\sim p(\tilde{\mathbf{z}},\mathbf{z})}\left[h(\tilde{\mathbf{z}})^\top h(\mathbf{z})\right] \tag{24}$$

$$+\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})}\left[e^{h(\tilde{\mathbf{z}})^\top h(\mathbf{z})/\tau}\right]\right] + \log|\mathcal{Z}|. \tag{25}$$

By inserting the definition $h = f \circ g$,

$$= -\frac{1}{\tau}\mathop{\mathbb{E}}_{\tilde{\mathbf{z}},\mathbf{z}\sim p(\tilde{\mathbf{z}},\mathbf{z})}\left[(f\circ g)(\tilde{\mathbf{z}})^\top (f\circ g)(\mathbf{z})\right] \tag{26}$$

$$+\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})}\left[e^{(f\circ g)(\tilde{\mathbf{z}})^\top (f\circ g)(\mathbf{z})/\tau}\right]\right] \tag{27}$$

$$+\log|\mathcal{Z}|, \tag{28}$$

we can identify the losses introduced in Proposition A,

$$=\mathcal{L}_{\mathsf{align}}(f;\tau) + \mathcal{L}_{\mathsf{uni}}(f;\tau) + \log|\mathcal{Z}|, \tag{29}$$

which recovers the original alignment term and the uniformity term for maximimizing entropy by means of a von Mises-Fisher KDE up to the constant $\log|\mathcal{Z}|$. According to Proposition A this equals

$$= \lim_{M\to\infty}\mathcal{L}_{\mathsf{contr}}(f;\tau,M) - \log M + \log|Z|, \tag{30}$$

which concludes the proof. □

Proposition 1 (Minimizers of the cross-entropy maintain the dot product)**.** *Let $\mathcal{Z} = \mathbb{S}^{N-1}$, $\tau > 0$ and consider the ground-truth conditional distribution of the form $p(\tilde{\mathbf{z}}|\mathbf{z}) = C_p^{-1}\exp(\kappa\tilde{\mathbf{z}}^\top\mathbf{z})$. Let h map onto a hypersphere with radius $\sqrt{\tau\kappa}$.*[4] *Consider the conditional distribution q_h parameterized by the model, as defined above in Theorem 1, where the hypothesis class for h is assumed to be sufficiently flexible such that $p(\tilde{\mathbf{z}}|\mathbf{z})$ and $q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ can match. If h is a minimizer of the cross-entropy $\mathbb{E}_{p(\tilde{\mathbf{z}}|\mathbf{z})}[-\log q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})]$, then $p(\tilde{\mathbf{z}}|\mathbf{z}) = q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ and $\forall\mathbf{z},\tilde{\mathbf{z}} : \kappa\mathbf{z}^\top\tilde{\mathbf{z}} = h(\mathbf{z})^\top h(\tilde{\mathbf{z}})$.*

[4] Note that in practice this can be implemented as a learnable rescaling operation of the network f.

Proof. By assumption, $q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ is powerful enough to match $p(\tilde{\mathbf{z}}|\mathbf{z})$ for the correct choice of h — in particular, for $h(\mathbf{z}) = \sqrt{\tau\kappa}\mathbf{z}$. The global minimum of the cross-entropy between two distributions is reached if they match by value and have the same support. Thus, this means

$$p(\tilde{\mathbf{z}}|\mathbf{z}) = q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}). \tag{31}$$

This expression also holds true for $\tilde{\mathbf{z}} = \mathbf{z}$; additionally using that h maps from a unit hypersphere to one with radius $\sqrt{\tau\kappa}$ yields

$$p(\mathbf{z}|\mathbf{z}) = q_{\mathrm{h}}(\mathbf{z}|\mathbf{z}) \tag{32}$$

$$\Leftrightarrow \quad C_p^{-1}e^{\kappa\mathbf{z}^\top\mathbf{z}} = C_h(\mathbf{z})^{-1}e^{h(\mathbf{z})^\top h(\mathbf{z})/\tau} \tag{33}$$

$$\Leftrightarrow \quad C_p^{-1}e^{\kappa} = C_h(\mathbf{z})^{-1}e^{\kappa} \tag{34}$$

$$\Leftrightarrow \quad C_p = C_h. \tag{35}$$

As the normalization constants are identical we get for all $\mathbf{z},\tilde{\mathbf{z}}\in\mathcal{Z}$

$$e^{\kappa\mathbf{z}^\top\tilde{\mathbf{z}}} = e^{h(\mathbf{z})^\top h(\tilde{\mathbf{z}})} \Leftrightarrow \kappa\mathbf{z}^\top\tilde{\mathbf{z}} = h(\mathbf{z})^\top h(\tilde{\mathbf{z}}). \tag{36}$$

□

Proposition 2 (Extension of the Mazur-Ulam theorem to hyperspheres and the dot product)**.** *Let $\mathcal{Z} = \mathbb{S}^{N-1}$ and $\mathcal{Z}' = \mathbb{S}_r^{N-1}$ be the hyperspheres with radius 1 and $r > 0$, respectively. If $h : \mathbb{R}^N \to \mathcal{Z}'$ is differentiable in the vicinity of $\mathcal{Z}$ and its restriction to $\mathcal{Z}$ maintains the dot product up to a constant factor, i.e., $\forall\mathbf{z},\tilde{\mathbf{z}}\in\mathcal{Z} : r^2\mathbf{z}^\top\tilde{\mathbf{z}} = h(\mathbf{z})^\top h(\tilde{\mathbf{z}})$, then h is an orthogonal linear transformation scaled by r for all $\mathbf{z}\in\mathcal{Z}$.*

Proof. First, we begin with the case $r = 1$. As h maintains the dot product we have:

$$\forall\mathbf{z},\tilde{\mathbf{z}}\in\mathcal{Z} : \mathbf{z}^\top\tilde{\mathbf{z}} = h(\mathbf{z})^\top h(\tilde{\mathbf{z}}). \tag{37}$$

We consider the partial derivative w.r.t. $\mathbf{z}$ and obtain:

$$\forall\mathbf{z},\tilde{\mathbf{z}}\in\mathcal{Z} : \tilde{\mathbf{z}} = \mathbf{J}_h^\top(\mathbf{z})h(\tilde{\mathbf{z}}). \tag{38}$$

Taking the partial derivative w.r.t. $\tilde{\mathbf{z}}$ yields

$$\forall\mathbf{z},\tilde{\mathbf{z}}\in\mathcal{Z} : \mathbf{I} = \mathbf{J}_h^\top(\mathbf{z})\mathbf{J}_h(\tilde{\mathbf{z}}). \tag{39}$$

We can now conclude

$$\forall\mathbf{z},\tilde{\mathbf{z}}\in\mathcal{Z} : \mathbf{J}_h(\tilde{\mathbf{z}})^{-1} = \mathbf{J}_h^\top(\mathbf{z}). \tag{40}$$

which implies a constant Jacobian matrix $\mathbf{J}_h(\mathbf{z}) = \mathbf{J}_h$ as the identity holds on all points in $\mathcal{Z}$, and further that the Jacobian $\mathbf{J}_h$ is orthogonal. Hence, $\forall\mathbf{z}\in\mathcal{Z} : h(\mathbf{z}) = \mathbf{J}_h\mathbf{z}$ is an orthogonal linear transformation.

Finally, for $r \neq 1$ we can leverage the previous result by introducing $h'(\mathbf{z}) := h(\mathbf{z})/r$. For h' the previous argument holds, implying that h' is an orthogonal transformation. Therefore, the restriction of h to $\mathcal{Z}$ is an orthogonal linear transformation scaled by r^2. □

Taking all of this together, we can now prove Theorem 2:

Theorem 2. *Let $\mathcal{Z} = \mathbb{S}^{N-1}$, the ground-truth marginal be uniform, and the conditional a vMF distribution (cf. Eq. 2). Let the restriction of the mixing function g to $\mathcal{Z}$ be injective and h be differentiable in a vicinity of $\mathcal{Z}$. If the assumed form of q_{h}, as defined above, matches that of p, and if f is differentiable and minimizes the CL loss as defined in Eq. (1), then for fixed $\tau > 0$ and $M \to \infty$, $h = f \circ g$ is linear, i.e., f recovers the latent sources up to an orthogonal linear transformation and a constant scaling factor.*

Proof. As f minimzes the contrastive loss $\mathcal{L}_{\mathsf{contr}}$ we can apply Theorem 1 to see that f also minimizes the cross-entropy between $p(\tilde{\mathbf{z}}|\mathbf{z})$ and $q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ for any point $\mathbf{z}$ on $\mathcal{Z}$. This means, we can apply Proposition 1 to show that the concatenation $h = f \circ g$ is an isometry with respect to the dot product. Finally, according to Proposition 2, h must then be a composition of an orthogonal linear transformation and a constant scaling factor. Thus, f recovers the latent sources up to orthogonal linear transformations, concluding the proof. □

A.2. Extension of theory to subspaces of $\mathbb{R}^N$

Here, we show how one can generalize the theory above from $\mathcal{Z} = \mathbb{S}^{N-1}$ to $\mathcal{Z} \subseteq \mathbb{R}^N$. Under mild assumptions regarding the ground-truth conditional distribution p and the model distribution q_{h}, we prove that all minimizers of the cross-entropy between p and q_{h} are linear functions, if $\mathcal{Z}$ is a convex body. Note that the hyperrectangle $[a_1, b_1] \times \ldots \times [a_N, b_N]$ is an example of such a convex body.

A.2.1. Assumptions

First, we restate the core assumptions for this proof. The main difference to the assumptions for the hyperspherical case above is that we assume different conditional distributions: instead of rotation-invariant von Mises-Fisher distributions, we use translation-invariant distributions (up to restrictions determined by the finite size of the space) of the exponential family.

Generative process Let $g : \mathcal{Z} \to \mathcal{X}$ be an injective function between the two spaces $\mathcal{Z} \subseteq \mathbb{R}^N$ and $\mathcal{X} \subseteq \mathbb{R}^K$ with $K \geq N$ and where $\mathcal{Z}$ is a convex body (e.g., a hyperrectangle). Further, let the marginal distribution be uniform, i.e., $p(\mathbf{z}) = |\mathcal{Z}|^{-1}$. We assume that the conditional distribution over positive pairs $p(\tilde{\mathbf{z}}|\mathbf{z})$ is an exponential distribution

$$\begin{aligned} p(\tilde{\mathbf{z}}|\mathbf{z}) &= C_p^{-1}(\mathbf{z}) e^{-\lambda \delta(\tilde{\mathbf{z}}, \mathbf{z})} \\ \text{with} \quad C_p(\mathbf{z}) &:= \int e^{-\lambda \delta(\mathbf{z}, \tilde{\mathbf{z}})} \, \mathrm{d}\tilde{\mathbf{z}}, \end{aligned} \tag{41}$$

where $\lambda > 0$ a parameter controlling the width of the distribution and δ is a (semi-)metric. If δ is a semi-metric, i.e., it does not fulfill the triangle inequality, there must exist a metric δ' such that δ can be written as the composition of a continuously invertible map $j : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ with $j(0) = 0$ and the metric, i.e., $\delta = j \circ \delta'$. Finally, we assume that during training one has access to samples from both of these distributions.

Note that unlike for the hypersphere, when sampling positive pairs $\mathbf{z}, \tilde{\mathbf{z}} \sim p(\mathbf{z})p(\tilde{\mathbf{z}}|\mathbf{z})$, it is no longer guaranteed that the marginal distributions of $\mathbf{z}$ and $\tilde{\mathbf{z}}$ are the same. When referencing the density functions – or using them in expectation values – $p(\cdot)$ will always denote the same marginal density, no matter if the argument is $\mathbf{z}$ or $\tilde{\mathbf{z}}$. Specifically, $p(\tilde{\mathbf{z}})$ does not refer to $\int p(\mathbf{z})p(\tilde{\mathbf{z}}|\mathbf{z})d\mathbf{z}$.

Model Let $\mathcal{Z}'$ be a subset of $\mathbb{R}^N$ that is a convex body and let $f : \mathcal{X} \to \mathcal{Z}'$ be the model whose parameters are optimized. We associate a conditional distribution $q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ with our model f through

$$\begin{aligned} q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}) &= C_q^{-1}(\mathbf{z}) e^{-\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))/\tau} \\ \text{with} \quad C_q(\mathbf{z}) &:= \int e^{-\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))/\tau} \, \mathrm{d}\tilde{\mathbf{z}}, \end{aligned} \tag{42}$$

where $C_q(\mathbf{z})$ is the partition function and δ is defined above.

A.2.2. Minimizing the cross-entropy

In a first step, we show the analogue of Proposition A for $\mathcal{Z}$ being a convex body:

Proposition 3. *For fixed $\tau > 0$, as the number of negative samples $M \to \infty$, the $\mathcal{L}_{\delta\text{-contr}}$ loss converges to*

$$\begin{aligned} \lim_{M \to \infty} \mathcal{L}_{\delta\text{-contr}}(f; \tau, M) - \log M = \\ \mathcal{L}_{\delta\text{-align}}(f; \tau) + \mathcal{L}_{\delta\text{-uni}}(f; \tau), \end{aligned} \tag{43}$$

where

$$\begin{aligned} \mathcal{L}_{\delta\text{-align}}(f; \tau) &:= \frac{1}{\tau} \mathop{\mathbb{E}}_{\substack{\mathbf{z} \sim p(\mathbf{z}) \\ \tilde{\mathbf{z}} \sim p(\tilde{\mathbf{z}}|\mathbf{z})}} \left[\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z})) \right] \\ \mathcal{L}_{\delta\text{-uni}}(f; \tau) &:= \mathop{\mathbb{E}}_{\mathbf{z} \sim p(\mathbf{z})} \left[\log \left(\mathop{\mathbb{E}}_{\tilde{\mathbf{z}} \sim p(\tilde{\mathbf{z}})} \left[e^{-\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))/\tau} \right] \right) \right], \end{aligned} \tag{44}$$

and $\mathcal{L}_{\delta\text{-contr}}(f; \tau, M)$ is as defined in Eq. (6).

Proof. This proof is adapted from Wang & Isola (2020). By the Continuous Mapping Theorem and the law of large numbers, for any $\mathbf{x}, \tilde{\mathbf{x}}$ and $\{ \mathbf{x}_i^- \}_{i=1}^M$ it follows almost surely

$$\begin{aligned}
&\lim_{M\to\infty} \log\left(\frac{1}{M}e^{-\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))/\tau} + \right.\\
&\qquad \left. \frac{1}{M}\sum_{i=1}^{M} e^{-\delta(f(\mathbf{x}),f(\mathbf{x}_i^-))/\tau}\right)\\
&= \log\left(\mathop{\mathbb{E}}_{\mathbf{x}^-\sim p_{\mathsf{data}}}\left[e^{-\delta(f(\mathbf{x}),f(\mathbf{x}^-))/\tau}\right]\right)\\
&= \log\left(\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})}\left[e^{-\delta(h(\mathbf{z}),h(\tilde{\mathbf{z}}))/\tau}\right]\right),
\end{aligned} \tag{45}$$

where in the last step we expressed the sample $\mathbf{x}$ and negative examples $\mathbf{x}^-$ in terms of their latent factors.

We can now express the limit of the entire loss function as

$$\begin{aligned}
&\lim_{M\to\infty} \mathcal{L}_{\delta\text{-contr}}(f;\tau,M) - \log M\\
&= \frac{1}{\tau}\mathop{\mathbb{E}}_{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathsf{pos}}}[\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))]\\
&\quad + \lim_{M\to\infty} \mathop{\mathbb{E}}_{\substack{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathsf{pos}}\\ \{\mathbf{x}_i^-\}_{i=1}^M \overset{\text{i.i.d.}}{\sim} p_{\mathsf{data}}}}\left[\log\left(\frac{1}{M}e^{-\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))/\tau}\right.\right.\\
&\qquad\qquad \left.\left. + \frac{1}{M}\sum_{i=1}^{M} e^{-\delta(f(\mathbf{x}),f(\mathbf{x}^-))/\tau}\right)\right]\\
&= \frac{1}{\tau}\mathop{\mathbb{E}}_{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathsf{pos}}}[\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))]\\
&\quad + \mathop{\mathbb{E}}_{\substack{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathsf{pos}}\\ \{\mathbf{x}_i^-\}_{i=1}^M \overset{\text{i.i.d.}}{\sim} p_{\mathsf{data}}}}\left[\lim_{M\to\infty}\log\left(\frac{1}{M}e^{-\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))/\tau}\right.\right.\\
&\qquad\qquad \left.\left. + \frac{1}{M}\sum_{i=1}^{M} e^{-\delta(f(\mathbf{x}),f(\mathbf{x}_i^-))/\tau}\right)\right].
\end{aligned} \tag{46}$$

Note that as δ is a (semi-)metric, the expression $e^{-\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))}$ is upper-bounded by 1. Hence, according to the Dominated Convergence Theorem one can switch the limit with the expectation value in the second step. Inserting the previous results yields

$$\begin{aligned}
&= \frac{1}{\tau}\mathop{\mathbb{E}}_{(\mathbf{x},\tilde{\mathbf{x}})\sim p_{\mathsf{pos}}}[\delta(f(\mathbf{x}),f(\tilde{\mathbf{x}}))]\\
&\quad + \mathop{\mathbb{E}}_{\mathbf{x}\sim p_{\mathsf{data}}}\left[\log\left(\mathop{\mathbb{E}}_{\mathbf{x}^-\sim p_{\mathsf{data}}}\left[e^{-\delta(f(\mathbf{x}),f(\mathbf{x}^-))/\tau}\right]\right)\right]\\
&= \frac{1}{\tau}\mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p(\mathbf{z})\\ \tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}}[\delta(h(\mathbf{z}),h(\tilde{\mathbf{z}}))]\\
&\quad + \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\log\left(\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})}\left[e^{-\delta(h(\mathbf{z}),h(\tilde{\mathbf{z}}))/\tau}\right]\right)\right]\\
&= \mathcal{L}_{\delta\text{-align}}(f;\tau) + \mathcal{L}_{\delta\text{-uni}}(f;\tau).
\end{aligned} \tag{47}$$

□

Next, we derive a property similar to Theorem 1, which suggests a practical method to find minimizers of the cross-entropy between the ground-truth p and model conditional q_{h}. This property is based on our previously introduced objective function in Eq. (6), which is a modified version of the InfoNCE objective in Eq. (1).

Theorem 3. *Let δ be a semi-metric and $\tau, \lambda > 0$ and let the ground-truth marginal distribution p be uniform. Consider a ground-truth conditional distribution $p(\tilde{\mathbf{z}}|\mathbf{z}) = C_p^{-1}(\mathbf{z})\exp(-\lambda\delta(\tilde{\mathbf{z}},\mathbf{z}))$ and the model conditional distribution*

$$\begin{aligned}
&q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}) = C_h^{-1}(\mathbf{z})e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau}\\
&\textit{with}\quad C_h(\mathbf{z}) := \int_{\mathcal{Z}} e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau}\mathrm{d}\tilde{\mathbf{z}}.
\end{aligned} \tag{48}$$

Then the cross-entropy between p and q_{h} is given by

$$\begin{aligned}
&\lim_{M\to\infty} \mathcal{L}_{\delta\text{-contr}}(f;\tau,M) - \log M + \log|\mathcal{Z}| =\\
&\qquad \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[H(p(\cdot|\mathbf{z}),q_{\mathrm{h}}(\cdot|\mathbf{z})\right],
\end{aligned} \tag{49}$$

which can be implemented by sampling data from the accessible distributions.

Proof. We use the definition of the cross-entropy to write

$$\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}[H(p(\cdot|\mathbf{z}),q_{\mathrm{h}}(\cdot|\mathbf{z})] \tag{50}$$

$$= -\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}[\log(q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}))]\right]. \tag{51}$$

We insert the definition of q_{h} and get

$$= -\mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}\left[\log(C_h^{-1}(\mathbf{z})) - \frac{1}{\tau}\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))\right]\right] \tag{52}$$

$$= \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}\left[\log(C_h(\mathbf{z})) + \frac{1}{\tau}\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))\right]\right]. \tag{53}$$

As $C_h(\mathbf{z})$ does not depend on $\tilde{\mathbf{z}}$ it can be moved out of the inner expectation value, yielding

$$= \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}\left[\frac{1}{\tau}\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}[\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))] + \log(C_h(\mathbf{z}))\right], \tag{54}$$

which can be written as

$$= \frac{1}{\tau}\mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p(\mathbf{z})\\ \tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}}[\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))] + \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})}[\log(C_h(\mathbf{z}))]. \tag{55}$$

Inserting the definition of C_h gives

$$= \frac{1}{\tau} \mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p(\mathbf{z}) \\ \tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}} [\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))] \tag{56}$$

$$+ \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[\log \left(\int e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau} d\tilde{\mathbf{z}} \right) \right]. \tag{57}$$

Next, the second term can be expanded by $1 = |\mathcal{Z}||\mathcal{Z}|^{-1}$, yielding

$$= \frac{1}{\tau} \mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p(\mathbf{z}) \\ \tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}} [\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))] \tag{58}$$

$$+ \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[\log \left(\int \frac{|\mathcal{Z}|}{|\mathcal{Z}|} e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau} d\tilde{\mathbf{z}} \right) \right]. \tag{59}$$

Finally, by using that the marginal is uniform, i.e., $p(\mathbf{z}) = |\mathcal{Z}|^{-1}$, this can be simplified as

$$= \frac{1}{\tau} \mathop{\mathbb{E}}_{\substack{\mathbf{z}\sim p(\mathbf{z}) \\ \tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}}|\mathbf{z})}} [\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))] \tag{60}$$

$$+ \mathop{\mathbb{E}}_{\mathbf{z}\sim p(\mathbf{z})} \left[\log \left(\mathop{\mathbb{E}}_{\tilde{\mathbf{z}}\sim p(\tilde{\mathbf{z}})} \left[e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau} \right] \right) \right] \tag{61}$$

$$+ \log |\mathcal{Z}| \tag{62}$$

$$= \lim_{M\to\infty} \mathcal{L}_{\delta\text{-contr}}(f; \tau, M) - \log M + \log p|\mathcal{Z}|. \tag{63}$$

□

A.2.3. Cross-entropy minimizers are isometries

Now we show a version of Proposition 1, that is generalized from hyperspherical spaces to (subsets of) $\mathbb{R}^N$.

Proposition 4 (Minimizers of the cross-entropy are isometries)**.** *Let δ be a semi-metric. Consider the conditional distributions of the form $p(\tilde{\mathbf{z}}|\mathbf{z}) = C_p^{-1}(\mathbf{z}) \exp(-\delta(\tilde{\mathbf{z}}, \mathbf{z})/\lambda)$ and*

$$q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}) = C_h^{-1}(\mathbf{z}) e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau}$$
$$\text{with} \quad C_h(\mathbf{z}) := \int_{\mathcal{Z}} e^{-\delta(h(\tilde{\mathbf{z}}),h(\mathbf{z}))/\tau} \mathrm{d}\tilde{\mathbf{z}}, \tag{64}$$

where the hypothesis class for h is assumed to be sufficiently flexible such that $p(\tilde{\mathbf{z}}|\mathbf{z})$ and $q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ can match for any point $\mathbf{z}$. If h is a minimizer of the cross-entropy $\mathcal{L}_{\mathsf{CE}} = \mathbb{E}_{p(\tilde{\mathbf{z}}|\mathbf{z})}[-\log q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})]$, then h is an isometry, i.e., $\forall \mathbf{z}, \tilde{\mathbf{z}} \in \mathcal{Z} : \lambda\tau\delta(\mathbf{z}, \tilde{\mathbf{z}}) = \delta(h(\mathbf{z}), h(\tilde{\mathbf{z}}))$.

Proof. Note that $q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z})$ is powerful enough to match $p(\tilde{\mathbf{z}}|\mathbf{z})$ for the correct choice of h, e.g. the identity. The global minimum of cross-entropy between two distributions is reached if they match by value and have the same support. Hence, if p is a regular density, q_{h} will be a regular density, i.e., q_{h} is continuous and has only finite values $0 \leq q_{\mathrm{h}} < \infty$. As the two distributions match, this means

$$p(\tilde{\mathbf{z}}|\mathbf{z}) = q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}). \tag{65}$$

This expression also holds true for $\tilde{\mathbf{z}} = \mathbf{z}$; additionally using the property $\delta(\mathbf{z}, \mathbf{z}) = 0$ yields

$$p(\mathbf{z}|\mathbf{z}) = q_{\mathrm{h}}(\mathbf{z}|\mathbf{z}) \tag{66}$$

$$\Leftrightarrow \quad C_p^{-1}(\mathbf{z}) e^{-\delta(\mathbf{z},\mathbf{z})/\lambda} = C_h^{-1}(\mathbf{z}) e^{-\delta(h(\mathbf{z}),h(\mathbf{z}))/\tau} \tag{67}$$

$$\Leftrightarrow \quad C_p(\mathbf{z}) = C_h(\mathbf{z}). \tag{68}$$

As the normalization constants are identical, we obtain for all $\mathbf{z}, \tilde{\mathbf{z}} \in \mathcal{Z}$

$$e^{-\delta(\tilde{\mathbf{z}},\mathbf{z})/\lambda} = e^{-\delta(h^*(\tilde{\mathbf{z}}),h^*(\mathbf{z}))/\tau} \tag{69}$$

$$\Leftrightarrow \quad \delta(\tilde{\mathbf{z}}, \mathbf{z}) = \frac{\lambda}{\tau} \delta(h^*(\tilde{\mathbf{z}}), h^*(\mathbf{z})). \tag{70}$$

By introducing a new semi-metric $\delta' := \lambda\tau^{-1}\delta$, we can write this as $\delta(\tilde{\mathbf{z}}, \mathbf{z}) = \delta'(h(\tilde{\mathbf{z}}), h(\mathbf{z}))$, which shows that h is an isometry. If there is no model mismatch, i.e., $\lambda = \tau$, this means $\delta(\mathbf{z}, \tilde{\mathbf{z}}) = \delta(h(\mathbf{z}), h(\tilde{\mathbf{z}}))$. □

Note, that this result does not depend on the choice of $\mathcal{Z}$ but just on the class of conditional distributions allowed.

A.2.4. Cross-entropy minimization identifies the ground-truth factors

Before we continue, let us recall a Theorem by Mankiewicz (1972):

Theorem C (Mankiewicz, 1972)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be normed linear spaces and let $\mathcal{V}$ be a convex body in $\mathcal{X}$ and $\mathcal{W}$ a convex body in $\mathcal{Y}$. Then every surjective isometry between $\mathcal{V}$ and $\mathcal{W}$ can be uniquely extended to an affine isometry between $\mathcal{X}$ and $\mathcal{Y}$.*

Proof. See Mankiewicz (1972). □

In addition, it is known that isometries on closed spaces are bijective:

Lemma A. *Assume h is an isometry of the closed space $\mathcal{Z}$ into itself, i.e., $\forall \mathbf{z}, \tilde{\mathbf{z}} : \delta(\mathbf{z}, \tilde{\mathbf{z}}) = \delta(h(\mathbf{z}), h(\tilde{\mathbf{z}}))$. Then h is bijective.*

Proof. See Lemma (2.6) in Całka (1982) for surjectivity. We show the injectivity by contradiction. Assume h is not injective. Then we can find a point $\tilde{\mathbf{z}} \neq \mathbf{z}$ where $h(\mathbf{z}) = h(\tilde{\mathbf{z}})$. But then $\delta(\mathbf{z}, \tilde{\mathbf{z}}) > \delta(\mathbf{z}, \mathbf{z})$ and $\delta(h(\mathbf{z}), h(\tilde{\mathbf{z}})) = \delta(h(\mathbf{z}), h(\mathbf{z})) = 0$ by the properties of δ. Hence, h is injective. □

Before continuing, we need to generalize the class of functions we consider as distance measures:

Lemma 1. *Let δ' be a the composition of a continuously invertible function $j : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ with $j(0) = 0$ and a metric δ, i.e., $\delta' := j \circ \delta$. Then, (i) δ' is a semi-metric and (ii) if a function $h : \mathbb{R}^n \to \mathbb{R}^n$ is an isometry of a space*

with the semi-metric δ', it is also an isometry of the space with the metric δ.

Proof. (i) Let $\mathbf{z}, \tilde{\mathbf{z}} \in \mathcal{Z}$. Per assumption j must be strictly monotonically increasing on $\mathbb{R}_{\geq 0}$. Since δ is a metric it follows $\delta(\mathbf{z}, \tilde{\mathbf{z}}) \geq 0 \Rightarrow \delta'(\mathbf{z}, \tilde{\mathbf{z}}) = j(\delta(\mathbf{z}, \tilde{\mathbf{z}})) \geq 0$, with equality iff $\mathbf{z} = \tilde{\mathbf{z}}$. Furthermore, since δ is a metric it is symmetric in its arguments and, hence, δ' is symmetric in its arguments. Thus, δ' is a semi-metric.

(ii) h is an isometry of a space with the semi-metric δ', allowing to derive that for all $\mathbf{z}, \tilde{\mathbf{z}} \in \mathcal{Z}$,

$$\delta'(h(\mathbf{z}), h(\tilde{\mathbf{z}})) = \delta'(\mathbf{z}, \tilde{\mathbf{z}}) \tag{71}$$

$$j(\delta(h(\mathbf{z}), h(\tilde{\mathbf{z}}))) = j(\delta(\mathbf{z}, \tilde{\mathbf{z}})) \tag{72}$$

and, applying the inverse j^{-1} which exists by assumption, yields

$$\delta(h(\mathbf{z}), h(\tilde{\mathbf{z}})) = \delta(\mathbf{z}, \tilde{\mathbf{z}}), \tag{73}$$

concluding the proof. □

By combining the properties derived before we can show that h is an affine function:

Theorem 4. *Let $\mathcal{Z} = \mathcal{Z}'$ be a convex body in $\mathbb{R}^N$. Let the mixing function g be differentiable and invertible. If the assumed form of q_{h} as defined in Eq. (42) matches that of p, and if f is differentiable and minimizes the cross-entropy between p and q_{h}, then we find that $h = f \circ g$ is affine, i.e., we recover the latent sources up to affine transformations.*

Proof. According to Proposition 4 h is an isometry and q_{h} is a regular probability density function. If the distance δ used in the conditional distributions p and q_{h} is a semi-metric as in Lemma 1, it follows that h is also an isometry for a proper metric. This also means that h is bijective according to Lemma A. Finally, Theorem C says that h is an affine transformation. □

We use the assumption that the marginal $p(\mathbf{z})$ is uniform, to show

Theorem 5. *Let $\mathcal{Z}$ be a convex body in $\mathbb{R}^N$, $h = f \circ g : \mathcal{Z} \to \mathcal{Z}$, and δ be a metric or a semi-metric as defined in Lemma 1. Further, let the ground-truth marginal distribution be uniform and the conditional distribution be as (5). Let the mixing function g be differentiable and injective. If the assumed form of q_{h} matches that of p, i.e.,*

$$\begin{aligned} q_{\mathrm{h}}(\tilde{\mathbf{z}}|\mathbf{z}) &= C_q^{-1}(\mathbf{z}) e^{-\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))/\tau} \\ \text{with} \quad C_q(\mathbf{z}) &:= \int e^{-\delta(h(\tilde{\mathbf{z}}), h(\mathbf{z}))/\tau} \, \mathrm{d}\tilde{\mathbf{z}}, \end{aligned} \tag{74}$$

and if f is differentiable and minimizes the $\mathcal{L}_{\delta\text{-contr}}$ objective in (6) for $M \to \infty$, we find that $h = f \circ g$ is invertible and affine, i.e., we recover the latent sources up to affine transformations.

Proof. According to Theorem 3 h minimizes the cross-entropy between p and q_{h} as defined in Eq. (4). Then according to Theorem 4, h is an affine transformation. □

This result can be seen as a generalized version of Theorem 2, as it is valid for any convex body $\mathcal{Z} \subseteq \mathbb{R}^N$ and allows a larger variety of conditional distributions. A missing step is to extend this theory beyond uniform marginal distributions. This will be addressed in future work.

Under some assumptions we can further narrow down possible forms of h, thus, showing that h in fact solves the nonlinear ICA problem only up to permutations and elementwise transformations.

For this, let us first repeat a result from Li & So (1994), that shows an important property of isometric matrices:

Theorem D. *Suppose $1 \leq \alpha \leq \infty$ and $\alpha \neq 2$. An $n \times n$ matrix $\mathbf{A}$ is an isometry of L^α-norm if and only if $\mathbf{A}$ is a generalized permutation matrix, i.e., $\forall \mathbf{z} : (\mathbf{A}\mathbf{z})_{\mathbf{i}} = \alpha_{\mathbf{i}} \mathbf{z}_{\sigma(\mathbf{i})}$, with $\alpha_i = \pm 2$ and σ being a permutation.*

Proof. See Li & So (1994). Note that this can also be concluded from the Banach-Lamperti Theorem (Lamperti et al., 1958). □

Leveraging this insight, we can finally show:

Theorem 6. *Let $\mathcal{Z}$ be a convex body in $\mathbb{R}^N$, $h : \mathcal{Z} \to \mathcal{Z}$, and δ be an L^α metric for $\alpha \geq 1, \alpha \neq 2$ or the α-th power of such an L^α metric. Further, let the ground-truth marginal distribution be uniform and the conditional distribution be as in Eq. (5), and let the mixing function g be differentiable and invertible. If the assumed form of $q_{\mathrm{h}}(\cdot|\mathbf{z})$ matches that of $p(\cdot|\mathbf{z})$, i.e., both use the same metric δ up to a constant scaling factor, and if f is differentiable and minimizes the $\mathcal{L}_{\delta\text{-contr}}$ objective in Eq. (6) for $M \to \infty$ we find that $h = f \circ g$ is a composition of input independent permutations, sign flips and rescalings.*

Proof. First, we prove the case where both conditional distributions use exactly the same metric. By Theorem 5 h is an affine transformation. Moreover, according to Proposition 4 is an isometry. Thus, by Theorem D, h is a generalized permutation matrix, i.e., a composition of permutations and sign flips.

Finally, for the case that δ matches the similarity measure in the ground-truth conditional distribution defined in Eq. (5) (denoted as δ^*) only up to a constant rescaling factor r, we know

$$\begin{aligned} &\forall \mathbf{z}, \tilde{\mathbf{z}} : \delta^*(\mathbf{z}, \tilde{\mathbf{z}}) = \delta(h(\mathbf{z}), h(\tilde{\mathbf{z}})) \\ \Leftrightarrow\ &\delta^*(\mathbf{z}, \tilde{\mathbf{z}}) = \delta^*\left(\frac{1}{r} h(\mathbf{z}), \frac{1}{r} h(\tilde{\mathbf{z}})\right). \end{aligned} \tag{75}$$

Thus, $\frac{1}{r}h$ is a δ^* isometry and the same argument as above holds, concluding the proof. □

Table 5. Identifiability up to affine transformations on the training set of 3DIdent. Mean ± standard deviation over 3 random seeds. As earlier, only the first row corresponds to a setting that matches the theoretical assumptions for linear identifiability; the others show distinct violations. Supervised training with unbounded space achieves scores of $R^2 = (99.98 \pm 0.01)\%$ and MCC $= (99.99 \pm 0.01)\%$. The last row refers to using the SimCLR (Chen et al., 2020a) augmentations to generate positive pairs. The last row refers to using the image augmentations suggested by Chen et al. (2020a) to generate positive image pairs; for details see Sec. A.3. In contrast to Table 4, the scores here are reported on the same data the models were trained on.

Dataset	Model f			Identity [%]	Unsupervised [%]	
$p(\cdot\|\cdot)$	Space	$q_{\mathrm{h}}(\cdot\|\cdot)$	M.	R^2	R^2	MCC
Normal	Box	Normal	✓	5.35 ± 0.72	97.83 ± 0.13	98.85 ± 0.07
Normal	Unbounded	Normal	✗	—"—	97.72 ± 0.02	55.90 ± 2.22
Laplace	Box	Normal	✗	—"—	97.95 ± 0.05	98.94 ± 0.03
Normal	Sphere	vMF	✗	—"—	66.73 ± 0.03	42.72 ± 3.20
Augm.	Sphere	vMF	✗	—"—	45.94 ± 1.80	47.6 ± 1.45

A.3. Experimental details

For the experiments presented in Sec. 4.1 we train our feature encoder for 300 000 iterations with a batch size of 6144 utilizing Adam (Kingma & Ba, 2015) with a learning rate of 10^{-4}. Like Hyvärinen & Morioka (2016; 2017), for the mixing network, we i) use 0.2 for the angle of the negative slope[5], ii) use L^2 normalized weight matrices with minimum condition number of 25 000 uniformly distributed samples. For the encoder, we i) use the default (0.01) negative slope ii) use 6 hidden layers with dimensionality $[N \cdot 10, N \cdot 50, N \cdot 50, N \cdot 50, N \cdot 50, N \cdot 10]$ and iii) initialize the normalization magnitude as 1. We sample 4096 latents from the marginal for evaluation. For MCC (Hyvärinen & Morioka, 2016; 2017) we use the Pearson correlation coefficient[6]; we found there to be no difference with Spearman[7].

For the experiments presented in Sec. 4.2.1, we use the same architecture as the encoder in (Klindt et al., 2021). As in (Klindt et al., 2021), we train for 300 000 iterations with a batch size of 64 utilizing Adam (Kingma & Ba, 2015) with a learning rate of 10^{-4}. For evaluation, as in (Klindt et al., 2021), we use 10 000 samples and the Spearman correlation coefficient.

For the experiments presented in Sec. 4.2.2, we train the feature encoder for 200 000 iterations using Adam with a learning rate of 10^{-4}. For the encoder we use a ResNet18 (He et al., 2016) architecture followed by a single hidden layer with dimensionality $N \cdot 10$ and LeakyReLU activation function using the default (0.01) negative slope. The scores on the training set are evaluated on 10% of the whole training set, 25 000 random samples. The test set consists of 25 000 samples not included in the training set. For the last row of Tab. 4 and Tab. 5 we used the best-working combination of image augmentations found by Chen et al. (2020a) to sample positive pairs. To be precise, we used a random crop and resize operation followed by a color distortion augmentation. The random crops had a uniformly distributed size (between 8% and 100% of the original image area) and a random aspect ration (between 3/4 and 4/3); subsequently, they were resized to the original image dimension (224×224) again. The color distortion operation itself combined color jittering (i.e., random changes of the brightness, contrast, saturation and hue) with color dropping (i.e., random grayscale conversations). We used the same parameters for these augmentations as recommended by Chen et al. (2020a).

The experiments in Sec. 4.1 took on the order of 5-10 hours on a GeForce RTX 2080 Ti GPU, the experiments on KITTI Masks took 1.5 hours on a GeForce RTX 2080 Ti GPU and those on 3DIdent took 28 hours on four GeForce RTX 2080 Ti GPUs. The creation of the 3DIdent dataset additionally required approximately 150 hours of compute time on a GeForce RTX 2080 Ti.

A.4. Details on 3DIdent

We build on the rendering pipeline of Johnson et al. (2017b) and use the Blender engine (Blender Online Community, 2021), as of version 2.91.0, for image rendering. The scenes depicted in the dataset show a rotated and translated object onto which a spotlight is directed. The spotlight is located on a half-circle above the scene and shines down. The scenes can be described by 10 parameters: the position of the object along the X-, Y- and Z-axis, the rotation of the object described by Euler angles (3), the position of the spotlight described by a polar angle, and the hue of the object, the ground and the spotlight. The value range is $[-3, 3]$ for all position parameters, and is $[-\pi/2, \pi/2]$ for the remaining parameters. The parameters are sampled from a 10-dimensional unit hyperrectangle, then rescaled to their corresponding value range. This ensures that the variance

[5] See e.g. https://pytorch.org/docs/stable/generated/torch.nn.LeakyReLU.html

[6] See e.g. https://numpy.org/doc/stable/reference/generated/numpy.corrcoef.html

[7] See e.g. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.spearmanr.html

of the latent factors is the same for all latent dimensions.

To ensure that the generative process is injective, we take two measures: First, we use a non-rotationally symmetric object (Utah tea pot, Newell, 1975), thus the rotation information is unambiguous. Second, we use different levels of color saturation for the object, the spotlight and the ground (1.0, 0.8 and 0.6, respectively), thus the object is always distinguishable from the ground.

A.4.1. Comparison to existing datasets

The proposed dataset contains high-resolution renderings of an object in a 3D scene. It features some aspects of natural scenes, e.g. complex 3D objects, different lighting conditions and continuous variables. Existing benchmarks (Klindt et al., 2021; Burgess & Kim, 2018; Gondal et al., 2019; Dittadi et al., 2021) for disentanglement in 3D scenes differ in important aspects to 3DIdent.

KITTI Masks (Klindt et al., 2021) only enables evaluating identification of the two-dimensional position and scale of the object instance. In addition, the observed segmentation masks are significantly lower resolution than examples in our dataset. 3D Shapes (Burgess & Kim, 2018) and MPI3D (Gondal et al., 2019) are rendered at the same resolution (64×64) as KITTI Masks. Whereas the dataset contributed by (Dittadi et al., 2021) is rendered at $2\times$ that resolution (128×128), our dataset is rendered at $3.5\times$ that resolution (224×224), the resolution at which natural image classification is typically evaluated (Deng et al., 2009). With that being said, we do note that KITTI Masks is unique in containing frames of natural video, and we thus consider it complementary to 3DIdent.

Burgess & Kim (2018), Dittadi et al. (2021), and Gondal et al. (2019) contribute datasets which contain variable object rotations around one, one, and two rotation axes, respectively, while 3DIdent contains variable object rotation around all three rotation axes as well as variable lighting conditions. Furthermore, each of these datasets were generated by sampling latent factors from an equidistant grid, thus only covering a limited number values along each axis of variation, effectively resulting in a highly coarse discretization of naturally continuous variables. As 3DIdent instead samples the latent factors uniformly in the latent space, this better reflects the continuous nature of the latent dimensions.

A.5. Effects of the Uniformity Loss

In previous work, Wang & Isola (2020) showed that a part of the contrastive (InfoNCE) loss — the uniformity loss — effectively ensures that the encoded features are uniformly distributed over a hypersphere. We now show that this part is crucial to ensure that the mapping is bijective. More precisely, we demonstrate that if the distribution of the encoded/reconstructed latents $h(\mathbf{z})$ has the same support as the distribution of $\mathbf{z}$, and both distributions are regular, i.e., their densities are non-zero and finite, then the transformation h is bijective.

First, we focus on the more general case of a map between manifolds:

Proposition 5. *Let $\mathcal{M}, \mathcal{N}$ be simply connected and oriented $\mathcal{C}^1$ manifolds without boundaries and $h : \mathcal{M} \to \mathcal{N}$ be a differentiable map. Further, let the random variable $\mathbf{z} \in \mathcal{M}$ be distributed according to $\mathbf{z} \sim p(\mathbf{z})$ for a regular density function p, i.e., $0 < p < \infty$. If the pushforward $p_{\#h}(\mathbf{z})$ of p through h is also a regular density, i.e., $0 < p_{\#h} < \infty$, then h is a bijection.*

Proof. We begin by showing by contradiction that the Jacobian determinant of h does not vanish, i.e., $|\det J_h| > 0$:

Suppose that the Jacobian determinant $|\det J_h|$ vanishes for some $\mathbf{z} \in \mathcal{M}$. Then the inverse of the Jacobian determinant goes to infinity at this point and so does the density of $h(\mathbf{z})$ according to the well-known transformation of probability densities. By assumption, both p and $p_{\#h}$ must be regular density functions and, thus, be finite. This contradicts the initial assumption and so the Jacobian determinant $|\det J_h|$ cannot vanish.

Next, we show that the mapping h is proper. Note that a map is called proper if pre-images of compact sets are compact (Ruzhansky & Sugimoto, 2015). Firstly, a continuous mapping between $\mathcal{M}$ and $\mathcal{N}$ is also closed, i.e., pre-images of closed subsets are also closed (Lee, 2013). In addition, it is well-known that continuous functions on compact sets are bounded. Lastly, according to the Heine–Borel theorem, compact subsets of $\mathbb{R}^D$ are closed and bounded. Taken together, this shows that h is proper.

Finally, according to Theorem 2.1 in (Ruzhansky & Sugimoto, 2015) a proper h with non-vanishing Jacobian determinant is bijective, concluding the proof. □

This theorem directly applies to the case of hyperspheres, which are simply connected and oriented manifolds without boundary. This yields:

Corollary 1. *Let $\mathcal{Z}$ be a hypersphere and $h : \mathcal{Z} \to \mathcal{Z}$ be a differentiable map. Further, let the marginal distribution $p(\mathbf{z})$ of the variable $\mathbf{z} \in \mathcal{Z}$ be a regular density function, i.e., $0 < p < \infty$. If the pushforward $p_{\#h}$ of p through h is also a regular density, i.e., $0 < p_{\#h} < \infty$, then h is a bijection.*

Therefore, we can conclude that a loss term ensuring that the encoded features are distributed according to a regular density function, such as the uniformity term, makes the map h bijective and prevents an information loss. Note that this does not assume that the marginal distribution of

the ground-truth latents $p(\mathbf{z})$ is uniform but only that it is regular and non-vanishing.

Note that while the proposition shows that the uniformity loss is sufficient to ensure bijectivity, we can construct counterexamples if its assumptions (like differentiability) are violated even in just a single point. For instance, the requirement of h being fully differentiable is most likely violated in large unregularized neural networks with ReLU nonlinearities. Here, one might need the full contrastive loss to ensure bijectivity of h.

ArXiv Changelog

- Current Version: Thanks to feedback from readers, we fixed a few inconsistencies in our notation. We also added a considerably simplified proof for Proposition 2.
- June 21, 2021: We studied violations of the uniformity assumption in greater details, and added Figure 2. We thank the anonymous reviewers at ICML for their suggestions. This is also the version available in the proceedings of ICML 2021.
- May 25, 2021: Extensions of the theory: We added additional propositions for the effects of the uniformity loss.
- February 17, 2021: First pre-print.

A.7 Provably Learning Object-Centric Representations

The following 25 pages were published as:

Jack Brady*, **Roland S. Zimmermann***, Yash Sharma, Bernhard Schölkopf, Julius von Kügelken, and Wieland Brendel. "Provably Learning Object-Centric Representations." *ICML (2023)*

A summary is given in Section 3.2 on page 49. * Equal contribution.

Abstract

Learning structured representations of the visual world in terms of objects promises to significantly improve the generalization abilities of current machine learning models. While recent efforts to this end have shown promising empirical progress, a theoretical account of when unsupervised object-centric representation learning is possible is still lacking. Consequently, understanding the reasons for the success of existing object-centric methods as well as designing new theoretically grounded methods remains challenging. In the present work, we analyze when object-centric representations can provably be learned without supervision. To this end, we first introduce two assumptions on the generative process for scenes comprised of several objects, which we call compositionality and irreducibility. Under this generative process, we prove that the ground-truth object representations can be identified by an invertible and compositional inference model, even in the presence of dependencies between objects. We empirically validate our results through experiments on synthetic data. Finally, we provide evidence that our theory holds predictive power for existing object-centric models by showing a close correspondence between models' compositionality and invertibility and their empirical identifiability.

Provably Learning Object-Centric Representations

Jack Brady [*1 2] **Roland S. Zimmermann** [*1 2] **Yash Sharma** [2 3] **Bernhard Schölkopf** [1]
Julius von Kügelgen [†1 4] **Wieland Brendel** [†1 2]

Abstract

Learning structured representations of the visual world in terms of objects promises to significantly improve the generalization abilities of current machine learning models. While recent efforts to this end have shown promising empirical progress, a theoretical account of when unsupervised object-centric representation learning is possible is still lacking. Consequently, understanding the reasons for the success of existing object-centric methods as well as designing new theoretically grounded methods remains challenging. In the present work, we analyze when object-centric representations can provably be learned without supervision. To this end, we first introduce two assumptions on the generative process for scenes comprised of several objects, which we call *compositionality* and *irreducibility*. Under this generative process, we prove that the ground-truth object representations can be identified by an invertible and compositional inference model, even in the presence of dependencies between objects. We empirically validate our results through experiments on synthetic data. Finally, we provide evidence that our theory holds predictive power for existing object-centric models by showing a close correspondence between models' compositionality and invertibility and their empirical identifiability.[1]

1 Introduction

Human intelligence exhibits an unparalleled ability to generalize from a limited amount of experience to a wide range of novel situations (Tenenbaum et al., 2011). To build machines with similar capabilities, a fundamental question is what types of abstract representations of sensory inputs enable such generalization (Goyal & Bengio, 2022). Research in cognitive psychology suggests that one key abstraction is the ability to represent visual scenes in terms of individual objects (Spelke, 2003; Spelke & Kinzler, 2007; Dehaene, 2020; Peters & Kriegeskorte, 2021). Such *object-centric representations* are thought to facilitate core cognitive abilities such as compositional generalization (Fodor & Pylyshyn, 1988; Lake et al., 2017; Battaglia et al., 2018; Greff et al., 2020) and causal reasoning over discrete concepts (Marcus, 2001; Gopnik et al., 2004; Gerstenberg & Tenenbaum, 2017; Gerstenberg et al., 2021).

Significant effort has thus gone into endowing machine learning models with the capacity to learn object-centric representations from raw visual input. While initial approaches were mostly supervised (Ronneberger et al., 2015; He et al., 2017; Chen et al., 2017), a recent wave of new methods explore learning object-centric representations without direct supervision (Greff et al., 2019; Burgess et al., 2019; Lin et al., 2020; Kipf et al., 2020; Locatello et al., 2020; Weis et al., 2021; Biza et al., 2023). These methods have begun exhibiting impressive results, showing potential to scale to complex visual scenes (Singh et al., 2022a; Sajjadi et al., 2022; Seitzer et al., 2023) and real-world video datasets (Kipf et al., 2022; Singh et al., 2022b; Elsayed et al., 2022).

Yet, despite this empirical progress, we still lack a *theoretical* understanding of when unsupervised object-centric representation learning is possible. This makes it challenging to isolate the reasons underlying the success and failure of existing object-centric models and to develop principled ways to improve them. Furthermore, it is currently not possible to design novel object-centric methods that are theoretically grounded and not solely based on heuristics, many of which break down in more realistic settings (Karazija et al., 2021; Papa et al., 2022; Yang & Yang, 2022).

In the present work, we aim to address this deficiency by investigating when object-centric representations can *provably* be learned without any supervision. To this end, we first specify a data-generating process for multi-object scenes as

[*]Equal contribution [†]Shared last author [1]MPI for Intelligent Systems, Tübingen [2]Tübingen AI Center, Tübingen [3]University of Tübingen, Tübingen, Germany [4]Department of Engineering, University of Cambridge, Cambridge, United Kingdom. Correspondence to: Jack Brady, Wieland Brendel <first.last@tue.mpg.de>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[1]Code/Website: brendel-group.github.io/objects-identifiability

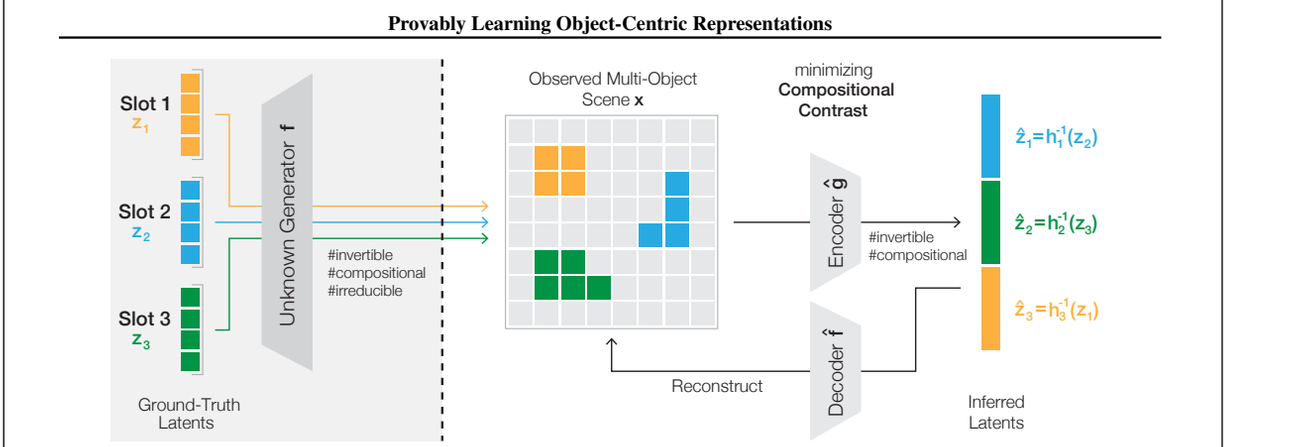


Figure 1. **When can unsupervised object-centric representations provably be learned?** We assume that observed scenes $\mathbf{x}$ comprising K objects are rendered by an unknown generator $\mathbf{f}$ from multiple ground-truth latent slots $\mathbf{z}_1, ..., \mathbf{z}_K$ (here, $K = 3$). We assume that this generative model has two key properties, which we call *compositionality* (Defn. 1) and *irreducibility* (Defn. 5). Under this model, we prove (Thm. 1): An invertible inference model with a compositional inverse yields latent slots $\hat{\mathbf{z}}_i$ which identify the ground-truth slots up to permutation and slot-wise invertible functions $\mathbf{h}_i$ (*slot identifiability*, Defn. 6). To measure violations of compositionality in practice, we introduce a contrast function (Defn. 7) which is zero if and only if a function is compositional, while to measure invertibility, we rely on the reconstruction loss in an auto-encoder framework.

a structured latent variable model in which each object is described by a subset of latents, or a latent *slot*. We then study the *identifiability* of object-centric representations under this model, i.e., we investigate under which conditions an inference model will be guaranteed to recover the subset of ground-truth latents for each object.

Because identifying the ground-truth latent variables is impossible without further assumptions on the generative process (Hyvärinen & Pajunen, 1999; Locatello et al., 2019), previous identifiability results primarily rely on distributional assumptions on the latents (Hyvärinen & Morioka, 2016; 2017; Hyvärinen et al., 2019; Khemakhem et al., 2020a;b; Klindt et al., 2021; Zimmermann et al., 2021). In contrast, we make no such assumptions, thus allowing for arbitrary statistical and causal dependencies between objects.

Structure and Main Contributions. In the present work, we instead take the position that the object-centric nature of the problem imposes a very specific *structure* on the *generator function* that renders scenes from latent slots (§ 2). Specifically, we define two key properties that this function should satisfy: *compositionality* (Defn. 1) and *irreducibility* (Defn. 5). Informally, these properties imply that every pixel can only correspond to one object and that information is shared across different parts of the same object but not between parts of different objects—inspired by the principle of independent causal mechanisms (Peters et al., 2017). Under this generative model, we then prove in § 3 our *main theoretical result*: the ground-truth latent slots can be identified without supervision by an invertible inference model with a compositional inverse (Thm. 1). To quantify compositionality, we introduce a *contrast function* (Defn. 7) that is zero if and only if a function is compositional; to quantify invertibility, we rely on reconstruction error. We validate on synthetic data that inference models which maximize invertibility and compositionality indeed identify the ground-truth latent slots, even with dependencies between latents (§ 5.1). Finally, we examine existing object-centric learning models on image data and find a close correspondence between models' compositionality and invertibility and their success in identifying the ground-truth latent slots (§ 5.2).

To the best of our knowledge, the present work provides the first identifiability result for object-centric representations. We hope that this lays the groundwork for a better understanding of success and failure in unsupervised object-centric learning, and that future work can build on these insights to develop more effective learning methods.

Notation. Bold lowercase $\mathbf{z}$ denotes vectors, bold uppercase $\mathbf{J}$ denotes matrices. For $n \in \mathbb{N}$, let $[n]$ denote the set $\{1, \ldots, n\}$. Additionally, if $\mathbf{f}$ is a function with n component functions, let $\mathbf{f}_S$ denote the restriction of $\mathbf{f}$ to the component functions indexed by $S \subseteq [n]$, i.e., $\mathbf{f}_S := (f_s)_{s \in S}$.

2 Generative Model

While humans have a clear intuition for what constitutes an object, formalizing this notion mathematically is not straightforward. Indeed, there is no universally agreed-upon definition of an object; various formalizations based upon distinct criteria co-exist (Green, 2019; Spelke, 1990; Koffka, 1936; Greff et al., 2020). We approach the problem by

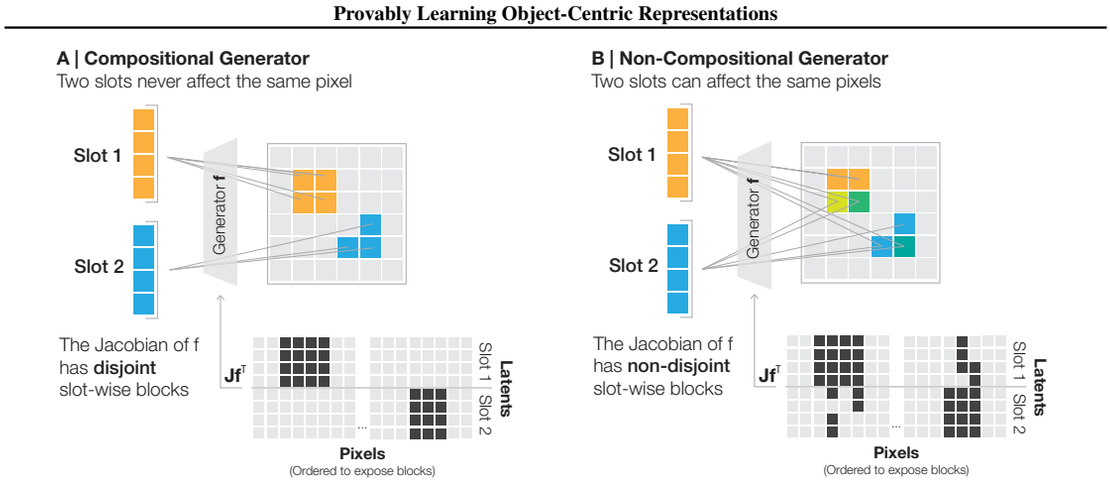


Figure 2. **Difference between a compositional and a non-compositional generator. (A)** For a compositional generator $\mathbf{f}$, every pixel is affected by at most one latent slot. As a result, there always exists an ordering of the pixels such that the generator's Jacobian $\mathbf{Jf}$ consists of disjoint blocks, one for each latent slot *(bottom)*. Note that both the pixel ordering and the specific structure of the Jacobian are not fixed across scenes and might depend on the latent input $\mathbf{z}$. **(B)** For a non-compositional generator, there exists no pixel ordering that exposes such a structure in the Jacobian, since the same pixel can be affected by more than one latent slot.

defining multi-object scenes in terms of a latent variable model (see Fig. 1 for an overview) and argue that the object-centric nature of the problem necessitates a very specific structure on the generator, which we leverage in § 3 to prove our identifiability result.

As a starting point, we assume that observed data samples $\mathbf{x}$ of multi-object scenes are generated from a set of latent random vectors $\mathbf{z}$ through a diffeomorphism[2] $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$, mapping from a *latent* space $\mathcal{Z}$ to an *observation* space $\mathcal{X}$,

$$\mathbf{z} \sim p_{\mathbf{z}}, \qquad \mathbf{x} = \mathbf{f}(\mathbf{z}). \tag{1}$$

The only assumption we place on $p_{\mathbf{z}}$ is that it is fully supported on $\mathcal{Z}$. In particular, we do not require independence and allow for arbitrary dependencies between components of $\mathbf{z}$, motivated by the fact that the presence or properties of certain objects may be correlated with those of other objects.

2.1 Slots and Compositionality

We think of an object in a scene as being encoded not by a single latent component z_i but instead by a group of latents $\mathbf{z}_k$ which specify its properties. For a scene comprised of K objects, we thus assume that the latent space $\mathcal{Z}$ factorizes into K subspaces $\mathcal{Z}_k$, which we refer to as *slots*. Each slot is assumed to have dimension M, representing, e.g., M distinct object properties. More precisely, $\forall k \in [K] : \mathcal{Z}_k = \mathbb{R}^M$, and $\mathcal{Z} = \mathcal{Z}_1 \times \cdots \times \mathcal{Z}_K = \mathbb{R}^{KM}$.

Let $\mathbf{z}_k$ be the latent vector of the k^{th} slot. The full KM-dimensional latent *scene representation* vector $\mathbf{z}$ is then given by the concatenation of the latents from all slots,

$$\mathbf{z} = (\mathbf{z}_1, \ldots, \mathbf{z}_K)\,. \tag{2}$$

We would like to ensure that each latent slot $\mathbf{z}_k$ is responsible for encoding a distinct object in a scene. To this end, the latent scene representation $\mathbf{z}$ should be rendered by $\mathbf{f}$ such that each slot generates *exactly* one object (see Fig. 1). If $\mathbf{f}$ is an arbitrary function with no additional constraints, however, this will generally not be the case.

First, $\mathbf{f}$ lacks any structure which ensures that an object is not generated by more than one latent slot. To see this, let $I_k(\mathbf{z}) \subseteq [N]$ denote the subset of pixels in an image generated from scene representation $\mathbf{z}$ that functionally depend on slot k,

$$I_k(\mathbf{z}) := \left\{ n \in [N] : \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) \neq \mathbf{0} \right\}. \tag{3}$$

Note the dependence on $\mathbf{z}$, which encodes that an object may appear in different places across different scenes.

Without further constraints on $\mathbf{f}$, the pixel subsets $I_k(\mathbf{z})$ and $I_j(\mathbf{z})$ can overlap for any $k \neq j$ such that latent slots k, j can affect the same pixels and thus contribute to generating the same object (see Fig. 2B, top). To avoid this, we impose the following structure on $\mathbf{f}$, which we call *compositionality*.

Definition 1 (Compositionality)**.** *Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be differentiable.* $\mathbf{f}$ *is said to be* compositional *if*

$$\forall \mathbf{z} \in \mathcal{Z} : \qquad k \neq j \implies I_k(\mathbf{z}) \cap I_j(\mathbf{z}) = \varnothing. \tag{4}$$

Compositionality implies that each pixel is a function of at most one latent slot and thus imposes a local sparsity

[2] a differentiable bijection with differentiable inverse

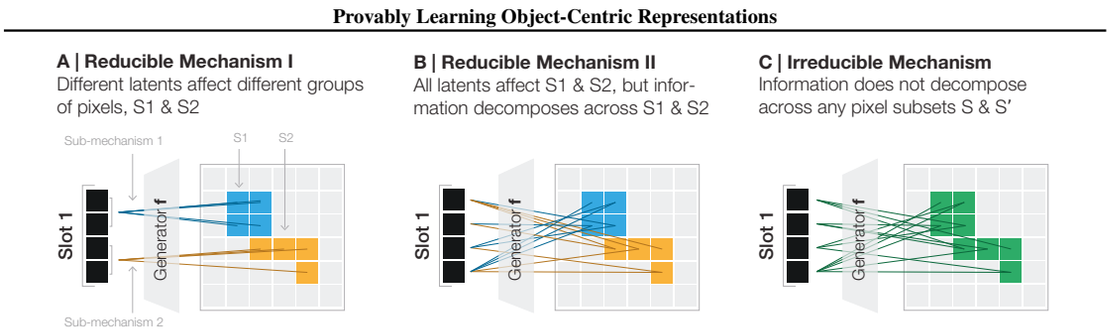


Figure 3. **(Ir)reducible mechanisms. (A)** A simple example of a *reducible mechanism* is one for which disjoint subsets of latents from the same slot render pixel groups S_1 and S_2 separately such that they form *independent sub-mechanisms* according to Defn. 4. This independence between sub-mechanisms is indicated by the difference in colors. **(B)** Not all reducible mechanisms look as simple as panel A: here, S_1 and S_2 depend on every latent component in the slot, but the information in $S_1 \cup S_2$ still decomposes across S_1 and S_2 as sub-mechanisms 1 and 2 are independent. **(C)** In contrast, for an irreducible mechanism, the information does not decompose across any pixel partition S, S', and so it is impossible to separate it into independent sub-mechanisms.

structure on the *Jacobian* matrix $\mathbf{Jf} = \left(\frac{\partial f_i}{\partial z_j}\right)_{ij}$ of $\mathbf{f}$, which is visualized in Fig. 2, bottom. Intuitively, the Jacobian of a compositional generator can always be brought into block structure through an appropriate permutation of the pixels. However, this block structure is local in that the required permutation may differ across scene representations $\mathbf{z}$.

2.2 Mechanisms and Irreducibility

While compositionality ensures that different latent slots do not generate the same object, we need an additional constraint on $\mathbf{f}$ to ensure that each slot generates only one object, rather than something humans would regard as multiple objects. To see this, consider the example depicted in Fig. 3A, where $\mathbf{f}$ maps the first half of the latent slot to the pixels denoted S_1 and the second half to S_2. It is clear that for humans, these groups of pixels would likely be considered as distinct objects. On the other hand, it is not immediately clear what formal criteria would give rise to such a distinction.

Intuitively, the issue with the two "sub-objects" S_1 and S_2 in Fig. 3A appears to be that they are *independent* of each other in some sense. To avoid such splitting of objects within slots, we would thus like to enforce that pixels belonging to the same object are *dependent* on one another. But what is a meaningful notion of such *instance-level* independence of objects? Since we are dealing with a single scene sampled according to Eq. (1), it cannot be statistical in nature. Instead, our intuition is more aligned with the notion of *algorithmic independence* of objects (Janzing & Schölkopf, 2010), a formalization[3] of the principle of independent causal mechanisms (ICM) which posits that physical generative processes consist of "autonomous modules that do not inform or influence each other" (Peters et al., 2017). The two subsets of pixels S_1 and S_2 in Fig. 3A are independent of each other in precisely this sense: they arise from autonomous processes that do not share information.

In the following, we therefore draw inspiration from prior implementations of the ICM principle (Daniusis et al., 2010; Janzing et al., 2012; Gresele et al., 2021, see § 4 for more details) to formalize our intuitions about independence of objects. First, we define the mapping which locally renders information from the k^{th} latent slot to the affected pixels $I_k(\mathbf{z})$ which we refer to as a *mechanism*.

Definition 2 (Mechanism)**.** *$\forall \mathbf{z} \in \mathcal{Z}, k \in [K]$, we define the k^{th}* mechanism *of $\mathbf{f}$ at $\mathbf{z}$ as the Jacobian matrix $\mathbf{Jf}_{I_k}(\mathbf{z})$.*

The k^{th} mechanism can be understood as the sub-matrix of the Jacobian of $\mathbf{f}$ whose rows correspond to the pixels $I_k(\mathbf{z})$ affected by slot k. Further, we define a *sub-mechanism* as the restriction to a *subset* of the affected pixels.

Definition 3 (Sub-Mechanism)**.** *$\mathbf{Jf}_S(\mathbf{z})$ is said to be a* sub-mechanism *of $\mathbf{Jf}_{I_k}(\mathbf{z})$, if $S \subseteq I_k(\mathbf{z})$ and S is nonempty.*

In light of these definitions, Fig. 3A consists of two sub-mechanism, $\mathbf{Jf}_{S_1}(\mathbf{z})$ and $\mathbf{Jf}_{S_2}(\mathbf{z})$, which generate pixels S_1 and S_2. To characterize the level of dependence between sets of pixels and their associated sub-mechanisms, we propose to use the matrix *rank*, which can be seen as a non-statistical measure of information as it locally characterizes the latent capacity used to generate the corresponding pixels.

Definition 4 (Independent/Dependent Sub-Mechanisms)**.** *Let $S_1, S_2 \subseteq [N]$ and $\mathbf{z} \in \mathcal{Z}$. The sub-mechanisms $\mathbf{Jf}_{S_1}(\mathbf{z})$ and $\mathbf{Jf}_{S_2}(\mathbf{z})$ are said to be* independent *if:*

$$\operatorname{rank}\left(\mathbf{Jf}_{S_1 \cup S_2}(\mathbf{z})\right) = \operatorname{rank}\left(\mathbf{Jf}_{S_1}(\mathbf{z})\right) + \operatorname{rank}\left(\mathbf{Jf}_{S_2}(\mathbf{z})\right). \tag{5}$$

Conversely, they are said to be dependent *if:*

$$\operatorname{rank}\left(\mathbf{Jf}_{S_1 \cup S_2}(\mathbf{z})\right) < \operatorname{rank}\left(\mathbf{Jf}_{S_1}(\mathbf{z})\right) + \operatorname{rank}\left(\mathbf{Jf}_{S_2}(\mathbf{z})\right).$$

[3] albeit an impractical one formulated in terms of Kolmogorov complexity (algorithmic information), which is not computable

Intuitively, two sub-mechanisms $\mathbf{Jf}_{S_1}(\mathbf{z})$ and $\mathbf{Jf}_{S_2}(\mathbf{z})$ are independent according to Defn. 4 if the information content of pixels $S_1 \cup S_2$ decomposes across S_1 and S_2 in the sense that the latent capacity required to *jointly* generate $S_1 \cup S_2$ (LHS of Eq. (5)) is the same as that required to generate S_1 and S_2 *separately* (RHS of Eq. (5)). Such a decomposition will occur when the rows of the sub-mechanism $\mathbf{Jf}_{S_1}(\mathbf{z})$ do not lie in the row-space of the sub-mechanism $\mathbf{Jf}_{S_2}(\mathbf{z})$ and vice-versa. This will be the case in Fig. 3A where $\mathbf{Jf}_{S_1}(\mathbf{z})$ and $\mathbf{Jf}_{S_2}(\mathbf{z})$ affect different pixels since the rows of the Jacobian for pixels S_1 and S_2 will never have non-zero entries for the same column. As shown in Fig. 3B, however, it could also be the case that all latents within a slot affect pixels in both S_1 and S_2, yet the information content of $S_1 \cup S_2$ still decomposes across S_1 and S_2 since the rows of $\mathbf{Jf}_{S_1}(\mathbf{z})$ and $\mathbf{Jf}_{S_2}(\mathbf{z})$ could span linearly independent subspaces.

To enforce that each slot generates only one object, we now finally place the condition on the mechanisms of $\mathbf{f}$ that they cannot be partitioned into independent sub-mechanisms (see Fig. 3C). We refer to this property as *irreducibility*.

Definition 5 (Irreducibility)**.** $\mathbf{f}$ *is said to have* irreducible mechanisms*, or is* irreducible*, if for all* $\mathbf{z} \in \mathcal{Z}$*,* $k \in [K]$ *and any partition of* $I_k(\mathbf{z})$ *into* S_1 *and* S_2*, the sub-mechanisms* $\mathbf{Jf}_{S_1}(\mathbf{z})$ *and* $\mathbf{Jf}_{S_2}(\mathbf{z})$ *are dependent in the sense of Defn. 4.*

3 Theory: Slot Identifiability

Given multi-object scenes sampled from the generative model outlined in § 2, we now seek to understand under what conditions an *inference model* $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ will provably identify the ground-truth object representations. Ideally, we would like $\hat{\mathbf{g}}$ to recover the true inverse $\mathbf{g} := \mathbf{f}^{-1}$, but that is generally only possible up to certain irresolvable ambiguities. In our multi-object setting, the objective is to separate the object representations such that each inferred slot captures *one and only one* ground-truth slot. We refer to this notion as *slot identifiability* and define it as follows.

Definition 6 (Slot Identifiability)**.** *Let* $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ *be a diffeomorphism. An inference model* $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ *is said to* slot-identify $\mathbf{z} = \mathbf{g}(\mathbf{x})$ *via* $\hat{\mathbf{z}} = \hat{\mathbf{g}}(\mathbf{x}) = \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z}))$ *if for all* $k \in [K]$ *there exist a unique* $j \in [K]$ *and a diffeomorphism* $\mathbf{h}_k : \mathcal{Z}_k \to \mathcal{Z}_j$ *such that* $\hat{\mathbf{z}}_j = \mathbf{h}_k(\mathbf{z}_k)$ *for all* $\mathbf{z} \in \mathcal{Z}$*.*

We are now in a position to state our main theoretical result (all complete proofs are provided in Appx. A).

Theorem 1. *Let* $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ *be a diffeomorphism that is compositional (Defn. 1) with irreducible mechanisms (Defn. 5). If an inference model* $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ *is* (i) *a diffeomorphism with* (ii) *compositional inverse* $\hat{\mathbf{f}} = \hat{\mathbf{g}}^{-1}$*, then* $\hat{\mathbf{g}}$ *slot-identifies* $\mathbf{z} = \mathbf{g}(\mathbf{x})$ *in the sense of Defn. 6.*

Proof Sketch. Irreducibility of $\mathbf{f}$ ensures that information is shared across different parts of an object, and compositionality of $\mathbf{f}$ that this information is not shared with other objects. This creates an asymmetry in the latent capacity required to encode the entirety of one object compared to parts of different objects. When $\hat{\mathbf{g}}$ satisfies (i) and (ii), this asymmetry can be leveraged to show that each inferred slot $\hat{\mathbf{z}}_j$ maps to *one* and *only* ground-truth slot $\mathbf{z}_k$ by a *proof by contradiction*. Namely, suppose that $\hat{\mathbf{g}}$ maps pixels of two distinct objects to the same slot j. If $\hat{\mathbf{g}}$ were to encode all latent information required to generate these pixels in slot j, there would not be sufficient total latent capacity to recover the entire scene, leading to a violation of (i) invertibility. Hence, information for at least one of the pixels needs to be distributed across multiple slots, violating (ii) compositionality of $\hat{\mathbf{f}} = \hat{\mathbf{g}}^{-1}$.

Implications for Object-Centric Learning. Thm. 1 highlights important conceptual points for object-centric representation learning. First, it shows that distributional assumptions on the latents $\mathbf{z}$ are not necessary for slot identifiability; instead, it suffices to enforce structure on the generator $\mathbf{f}$. This falls in line with state-of-the-art (SOTA) object-centric learning methods (Locatello et al., 2020; Singh et al., 2022b; Seitzer et al., 2023; Elsayed et al., 2022), which are based on an auto-encoding framework, thus imposing no additional structure on $p_{\mathbf{z}}$. However, while these models directly enforce invertibility through the reconstruction objective, it is less clear whether and to what extent they also enforce compositionality. Specifically, compositionality is not explicitly optimized in any object-centric methods. Yet, the success of SOTA models in practice suggests that it may be implicitly enforced to some extent through additional inductive biases in the model. We explore this point empirically (see Fig. 6) and leave a more theoretical exploration for future work.

Thm. 1 also emphasizes that using a restricted latent bottleneck plays an important role in achieving slot identifiability. Specifically, Thm. 1 is predicated on $\dim(\mathbf{z})$=$\dim(\hat{\mathbf{z}})$ and would no longer hold in its current form if $\dim(\mathbf{z})$<$\dim(\hat{\mathbf{z}})$. The importance of restricting the latent capacity of object-centric models was emphasized empirically by Engelcke et al. (2020a). Yet, the most successful object-centric models in practice often use $\dim(\mathbf{z})$<$\dim(\hat{\mathbf{z}})$ (Dittadi et al., 2022; Locatello et al., 2020; Sajjadi et al., 2022). A potential explanation for this discrepancy is that SOTA object-centric models do encode information from multiple objects in each latent slot, but this additional information is ignored by the decoder during reconstruction such that image-level segmentations remain accurate. We provide some evidence for this hypothesis through experiments with existing object-centric models in § 5.2.

Measuring Compositionality. While Thm. 1 reveals properties an inference function should satisfy to achieve slot identifiability, it presents these properties in an abstract mathematical form. If we seek to leverage Thm. 1 to assess the performance of existing object-centric models or inform

new training objectives for object-centric learning, we require a way to quantify whether an inference model is (i) a diffeomorphism and (ii) compositional. Regarding (i), one clear choice is to train an auto-encoder with differentiable encoder $\hat{\mathbf{g}}$ and decoder $\hat{\mathbf{f}}$ and minimize reconstruction loss to enforce invertibility. Regarding (ii), on the other hand, it is much less obvious how to quantify compositionality. To this end, we introduce the following contrast function, which we prove to be zero if and only if a function is compositional:

Definition 7 (Compositional Contrast)**.** *Let* $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ *be differentiable. The* compositional contrast *of* $\mathbf{f}$ *at* $\mathbf{z}$ *is*

$$C_{\text{comp}}(\mathbf{f}, \mathbf{z}) = \sum_{n=1}^{N} \sum_{k=1}^{K} \sum_{j=k+1}^{K} \left\| \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) \right\| \left\| \frac{\partial f_n}{\partial \mathbf{z}_j}(\mathbf{z}) \right\| . \quad (6)$$

For a given scene representation $\mathbf{z}$ and generator $\mathbf{f}$, the contrast function in Eq. (6) computes the sum over all pixels n of all pairwise products of the (L2) norms of those pixels' gradients with respect to any two distinct slots $k \neq j$. As such, it is a non-negative quantity that can only be zero if every pixel is affected by at most one slot (i.e., $\mathbf{f}$ is *compositional*), for otherwise there would be a pair of slots $k \neq j$ for which the gradient norms are both non-zero resulting in their product being non-zero.

We leverage this characterization of compositionality to provide our second result, which can be viewed as an optimization-based perspective on Thm. 1.

Theorem 2. *Let* $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ *be a diffeomorphism that is compositional (Defn. 1) with irreducible mechanisms (Defn. 5). If an encoder* $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ *and decoder* $\hat{\mathbf{f}} : \mathcal{Z} \to \mathcal{X}$ *are both differentiable and solve the following functional equation*

$$\mathbb{E}_{\mathbf{x} \sim p_{\mathbf{x}}} \left[\left\| \hat{\mathbf{f}}(\hat{\mathbf{g}}(\mathbf{x})) - \mathbf{x} \right\|_2^2 + \lambda C_{\text{comp}} \left(\hat{\mathbf{f}}, \hat{\mathbf{g}}(\mathbf{x}) \right) \right] = 0, \quad (7)$$

for $\lambda > 0$*, then* $\hat{\mathbf{g}}$ *slot-identifies* $\mathbf{z}$ *in the sense of Defn. 6.*

4 Related Work

Object-Centric Generative Models. Prior works have also formulated generative models for multi-object scenes based on latent slots (Roux et al., 2011; Heess, 2012; Greff et al., 2015; 2017; 2019; van Steenkiste et al., 2018; von Kügelgen et al.; Engelcke et al., 2020b; 2021), though without studying identifiability. Our assumptions on the generative model (§ 2) bear intuitive similarity to some of these prior works, but they also differ in several fundamental ways. First, compositionality (Defn. 1) is stated as a desideratum for nearly all object-centric generative models. Yet, this constraint is not actually enforced by most existing approaches, particularly those based on spatial mixture models in which every slot may affect every pixel (Greff et al., 2015; 2017; 2019; van Steenkiste et al., 2018; Engelcke et al., 2020b; 2021). More closely related is a dead-leaves model approach, in which a scene is sequentially generated by layering objects such that each pixel is affected by at most one slot (Roux et al., 2011; von Kügelgen et al.; Tangemann et al., 2023). In contrast, we define compositionality directly through assumptions on the structure of the (Jacobian of the) generator. Second, our irreducibility criterion (Defns. 4 and 5) bears conceptual similarity to prior works, which assume that different objects do not share information whereas parts of the same object do (Hyvärinen & Perkiö, 2006; Greff et al., 2015; 2017; van Steenkiste et al., 2018). Importantly, however, these works formalize this intuition using statistical criteria such as *statistical independence* between pixels from different objects and dependence between pixels from the same object. However, this leads to an incorrect characterization of objects: e.g., the presence of a coffee cup should increase the likelihood that a table is also present, despite these being separate objects (Träuble et al., 2021; Schölkopf et al., 2021). Here, we instead formulate independence/dependence between objects in a *non-statistical* sense, inspired by algorithmic independence of mechanisms.

Objects and Causal Mechanisms. In causal modelling (Spirtes et al., 2001; Pearl, 2009), a *mechanism* typically refers to a function that determines the value of an effect variable from its direct causes and possibly a noise term, leading to a conditional distribution of effect given causes. Thus, we could view objects as the effects of the latent variables that cause them. While the causal variables are generally not independent, it has been argued that the mechanisms producing them should be (Schölkopf et al., 2012; Peters et al., 2017). Since this is an independence between functions or conditionals rather than between random variables, it is non-trivial to formalize it statistically (Janzing & Schölkopf, 2010; Guo et al., 2022). Hence, various implementations of the principle have been proposed (Daniusis et al., 2010; Janzing et al., 2010; 2012; Shajarisales et al., 2015; Locatello et al., 2018; Besserve et al., 2018; 2021; Janzing, 2021), typically for settings in which both cause and effect are observed. Our notion of independent sub-mechanisms is most closely related to work by Gresele et al. (2021), who also study representation learning and define mechanisms more broadly in terms of the Jacobian $\mathbf{Jf}$: they assume independent latents and formalize mechanism independence as column-orthogonality of the Jacobian. In contrast, our rank condition (Eq. (5)) is inspired by object-centric representation learning with dependent latents.

Identifiable Representation Learning. As this is the first identifiability study of unsupervised object-centric representations, our problem setting differs from existing work both in terms of the assumptions we make on the generative process and the type of identifiability that we aim to achieve.

First, prior work on identifiable representation learning commonly places assumptions on the latent distribution, such as conditional independence given an auxiliary variable (Hyvärinen & Morioka, 2016; 2017; Hyvärinen et al., 2019; Khemakhem et al., 2020a; Hälvä & Hyvärinen, 2020; Hälvä et al., 2021) or access to views arising from pairs of similar latents (Gresele et al., 2019; Klindt et al., 2021; Zimmermann et al., 2021; von Kügelgen et al., 2021), while leaving the generator $\mathbf{f}$ completely unconstrained. In contrast, we place no assumptions on $p_{\mathbf{z}}$ and instead impose structure on (the Jacobian of) the generator $\mathbf{f}$. Recent works have also leveraged assumptions on $\mathbf{Jf}$ such as orthogonality (Gresele et al., 2021; Zheng et al., 2022; Reizinger et al., 2022; Buchholz et al., 2022), unit determinant (Yang et al., 2022), or a *fixed* sparsity structure (Moran et al., 2021; Lachapelle et al., 2021; Lachapelle & Lacoste-Julien, 2022). While the latter relates to our definition of compositionality (Defn. 1), we crucially allow the sparsity pattern on $\mathbf{Jf}$ to vary with $\mathbf{z}$ (in line with the basic notion that objects are not fixed in space), and impose sparsity with respect to slots rather than individual latents. Secondly, existing work typically aims to identify individual latent components z_i up to permutations (or linear transformations). However, this is inappropriate for object-centric representation learning, where we aim to capture and isolate the subsets of latents corresponding to each object in well-defined slots. Identifying such groups of latents is similar to efforts in independent subspace analysis (ISA; Hyvärinen & Hoyer, 2000). However, results for ISA are generally restricted to linear models and independent groups, whereas we allow for nonlinear models and dependence. Our notion of slot identifiability is most closely related to that of block-identifiability introduced by (von Kügelgen et al., 2021) and can be seen as an extension or generalization thereof to a setting with multiple blocks.

5 Experiments

Thm. 2 states that inference models which minimize reconstruction loss $\mathcal{L}_{\text{rec}}$ and compositional contrast C_{comp} achieve *slot identifiability* (Defn. 6). This provides a concrete way to empirically test our main theoretical result. To do so, we perform two main sets of experiments. First, in § 5.1 we generate controlled synthetic data according to the process specified in § 2 and train an inference model on this data which directly optimizes $\mathcal{L}_{\text{rec}}$ and C_{comp} jointly. Second, in § 5.2 we seek to better understand the relationship between $\mathcal{L}_{\text{rec}}$, C_{comp}, and slot identifiability in existing object-centric models. To this end, we analyze a set of models trained on a multi-object sprites dataset.

Quantifying Slot Identifiability. To assess whether a model is slot identifiable in practice, we first establish a metric to measure slot identifiability. Specifically, we want to measure if there exists an invertible function between each ground-truth and exactly one inferred latent slot. To this end, we first fit nonlinear models between inferred and ground-truth slots and measure their quality by the R^2 coefficient of determination. To properly measure this R^2 score, we must first match each ground-truth slot to its corresponding inferred slot as permutations could exist between slots. For our experiments in § 5.1, this permutation will be global i.e. the same for all inferred latents, thus we use the Hungarian Algorithm (Kuhn, 1955) to find the optimal matching based on the R^2 scores for models fit between every pair of slots. For our experiments on image data in § 5.2, however, such a permutation will be local due to the permutation invariance of the generator. To resolve this, we follow a procedure similar to that of Locatello et al. (2020) and Dittadi et al. (2022) using online matching when fitting models between slots. Specifically, at every training iteration, we compute a matching loss for each sample for all possible pairings of ground-truth and inferred slots and use the Hungarian algorithm to find the optimal assignment for minimizing this loss. After resolving permutations, the R^2 scores for the matched slots tell us how much information about each ground-truth slot is contained in one inferred slot. We also need to ensure, however, that inferred slots only contain information about one ground-truth slot and not multiple. To this end, we correct this score by subtracting the maximum R^2 score from models fit between each inferred latent slot and the ground-truth slots that it was not previously matched with. Taking the mean of this score across all slots yields the final score, which we refer to as the *slot identifiability score* (SIS). Further details on the metric are given in Appx. B.4.

5.1 Synthetic Data

Experimental Setup. To generate synthetic data according to § 2, we first sample a KM-dimensional latent vector from a normal distribution $p_{\mathbf{z}} = \mathcal{N}(0, \Sigma)$, where we consider scenarios with both statistically independent latents ($\Sigma = \mathbf{I}$) and dependent latents ($\Sigma \sim \text{Wishart}_{KM}(\mathbf{I}, KM)$). We then partition the latent vector into K slots, each with dimension M, and apply the same multi-layer perceptron (MLP) to each of the K slots separately. The MLP has 2 layers, uses LeakyReLU non-linearities, and is chosen to lead to invertibility almost surely by following the settings used in previous work (Hyvärinen & Morioka, 2016; 2017; Zimmermann et al., 2021). Observations $\mathbf{x}$ are obtained by concatenating the slot-wise MLP outputs such that the generator is compositional according to Defn. 1 as well as invertible.[4] We train models with a number of slots $K \in \{2, 3, 5\}$ and $\lambda \in \{10^{-7}, 10^{-5}, 10^{-2}, 0, 1, 10\}$ (see Thm. 2) each across 10 random seeds (180 models in total). In all cases, we use slot-dimension $M = 3$ and slot-output dimension of 20 such that $\dim(\mathbf{x}) = K \cdot 20$. Further details on this setup may be found in Appx. B.1.

[4] Regarding enforcing irreducibility, see Appx. B.1.

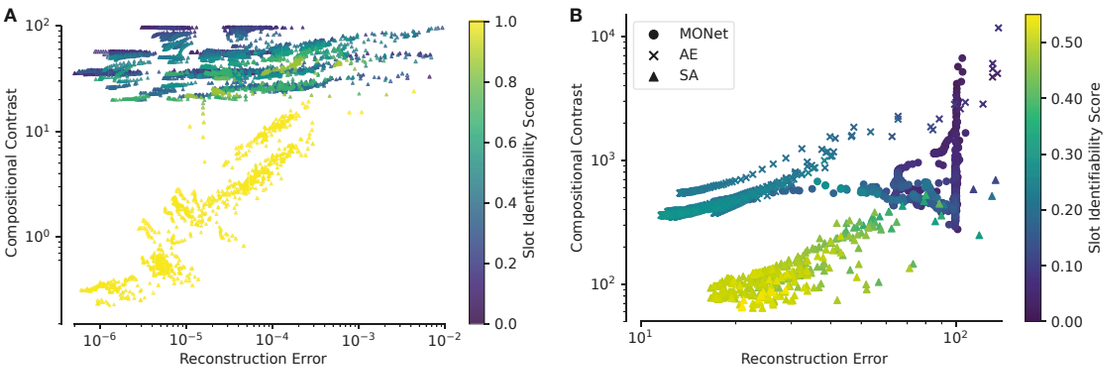


Figure 4. **(A) Experimental validation of Thm. 2.** We trained models on synthetic data generated according to § 2 with 2, 3, 5 independent latent slots (see § 5.1). The color coding indicates the level of identifiability achieved by the model, measured by the Slot Identifiability Score (SIS), where higher values correspond to more identifiable models. As predicted by our theory, if a model sufficiently minimizes both reconstruction error and compositional contrast, then it identifies the ground-truth latent slots. **(B) Application of Thm. 2 to existing object-centric models.** We train 3 existing object-centric architectures—MONet, Slot Attention (SA), and an additive auto-encoder (AE)—on image data and visualize their SIS as a function of both reconstruction error and compositional contrast. We see across models that, in general, SIS increases as reconstruction error and compositional contrast are minimized.

Results. In Fig. 4A, we visualize the SIS as a function of the reconstruction error and compositional contrast for independent latents for all $K \in \{2, 3, 5\}$. We normalize $\mathcal{L}_{\text{rec}}$ and C_{comp} to ensure that their scores are comparable across different K, which we discuss in further detail in Appx. B.3. As predicted by Thm. 2, we can see that all models that minimize both objectives jointly yield high SIS, whereas models that fail to minimize, e.g., the compositional contrast achieve subpar identifiability. Results for dependent latents yield a similar trend which can be seen in Fig. 5.

5.2 Existing Object-Centric Models

Experimental Setup. We now aim to understand the predictions made by our theory in the context of existing object-centric models trained on image data. To this end, we consider image data generated by the Spriteworld renderer (Watters et al., 2019). Specifically, we generate images with 2 to 4 objects, each described by 4 continuous (size, color, x/y position) and 1 discrete (shape) independent latent factors. Samples of this dataset are shown in Fig. 8. We investigate three object-centric approaches on this data: Slot Attention (Locatello et al., 2020), MONet (Burgess et al., 2019), and an additive auto-encoder. We train all models with 4 latent slots, each with dimension 16, leading to an inferred latent dimension larger than the ground-truth. This discrepancy between inferred and ground-truth latent dimensionality is ubiquitous in existing object-centric models. However, it violates our theoretical assumptions which require equal dimensions. See Appx. B.2 for further experimental details.

Results. SIS as a function of reconstruction error and compositional contrast is shown in Fig. 4B. Similar to Fig. 4A, SIS tends to increase as $\mathcal{L}_{\text{rec}}$ and C_{comp} are minimized, highlighting that our theory holds predictive power for slot identifiability in existing object-centric models. Notably, this is in spite of our theoretical assumptions not being exactly met due to the inferred latent dimension exceeding the ground-truth. This mismatch in dimension does seem to have an effect on SIS, however, which can be seen in Fig. 7. Here, we can see that the subtracted R^2 score in the SIS computation is non-zero across models suggesting that these models are using their additional latent capacity to encode information from multiple objects, despite the decoder presumably not using this information during reconstruction.

6 Discussion

Limitations of Experiments. We emphasize that the main goal of this work is to create a theoretical foundation for object-centric learning. Hence, we focus our experiments on validating Thm. 2 (§ 5.1) and exploring our theoretical predictions in existing object-centric models (§ 5.2). While our experiments in § 5.2 provide evidence that existing models which minimize $\mathcal{L}_{\text{rec}}$ and C_{comp} achieve higher SIS, scaling up these experiments to more models and datasets would lead to a more comprehensive understanding of the exact extent to which the performance of existing models can be understood from our theory. We leave such a larger empirical study for future work.

Limitations of Theory. While we believe that our theoretical assumptions capture the essence of important concepts in object-centric learning, they will be violated to various degrees in practical scenarios. For example, the assumption of compositionality (Defn. 1) on the generator $\mathbf{f}$ is broken

by translucency/reflection, as a single pixel can then be affected by multiple latent slots. Additionally, occlusions are not yet fully covered by our theory, as pixels at the border of occluding objects would be affected by multiple latent slots. Additionally, it is common to assume in practice that the generator $\mathbf{f}$ is invariant to permutations of the latent slots it acts on. This permutation invariance leads to a lack of invertibility of $\mathbf{f}$, however, as permuted latents will give rise to the same observation. We anticipate that our theoretical results can be adapted to incorporate such a permutation invariant generator but leave this for future work.

Relationship to Existing Definitions of Objects. Under our framework, groups of pixels corresponding to an object have the property that the latent capacity needed to encode partitions of these pixels separately exceeds the latent capacity needed to encode the pixels as a whole (Defn. 5). Intuitively, this implies that there is latent information shared across different parts of an object. By considering the location of objects as one such latent information, our definition relates to the Gestalt law of common fate (Koffka, 1936; Tangemann et al., 2023) and the concept of a Spelke Object (Spelke, 1990; Chen et al., 2022) which posit that pixels belonging to the same object move together. Furthermore, by considering color or texture as shared latent information, our definition relates to the Gestalt law of similarity (Koffka, 1936) that posits that items sharing visual features tend to be grouped together as a single object.

Extensions of Theory. While our theoretical results provide relatively general conditions under which object-centric representations can be identified, there are several potential ways our results could be extended. First, we hypothesize that the reverse implication of our main result may hold as well, i.e., given the generative model in § 3, compositionality and invertibility are not only sufficient but also necessary conditions for slot identifiability. A formal proof of this conjecture would further highlight the importance of these properties. Additionally, it would be interesting to aim to extend our theoretical approach to identifying not just objects but also abstractions such as part-whole hierarchies (Hinton, 2021) or individual object attributes. In this case, our notion of compositionality would need to be adjusted to account for abstractions that interact during generation. Lastly, it would be interesting to extend our results to leverage weakly-supervised information, such as motion, which has been shown empirically to be helpful for object-centric learning (Tangemann et al., 2023; Kipf et al., 2022; Elsayed et al., 2022; Chen et al., 2022).

Optimizing C_{comp} in Object-Centric Models. While creating a new method for object-centric learning is not the focus of this work, one question based on Thm. 2 is whether C_{comp} can be optimized directly in object-centric models on image data to improve slot identifiability. In this setting, explicitly optimizing C_{comp}, as was done in § 5.1, is challenging as the contrast in its current form is based on Jacobians. Thus, naively optimizing it through gradient descent corresponds to second-order optimization, which creates computational challenges for larger models and data dimensionalities. As previously noted, it could also be the case that there exist implicit ways to enforce that C_{comp} is minimized, which could be occurring to some extent through inductive biases in existing object-centric models. We leave finding computationally efficient ways to minimize C_{comp}, whether explicit or implicit, for future work.

Concluding Remarks. Representing scenes in terms of objects is a key aspect of visual intelligence and an important component of generalization in humans. While empirical object-centric learning methods are increasingly successful, we have thus far been lacking a precise theoretical understanding of what properties of the data and model are sufficient to provably learn object-centric representations. To the best of our knowledge, this work is the first to provide such a theoretical understanding. Along with invertibility, two intuitive assumptions on the generator—compositionality and irreducibility–are sufficient to identify the ground-truth object representations. By extending identifiability theory towards object-centric learning, we hope to facilitate a deeper understanding of existing object-centric models as well as provide a solid foundation for the next generation of models to build upon.

Author Contributions. JB developed the theory with technical help from RSZ, insight from YS, and advising from WB and JvK. JB implemented and executed the experiments with help from RSZ and YS, while RSZ implemented the C_{comp} and SIS metrics on image data. JB and JvK led the writing of the manuscript with help from WB, BS, and RSZ. WB and RSZ created all figures in the manuscript.

Acknowledgments. We thank: the anonymous reviewers for helpful suggestions which led to improvements in the manuscript, Andrea Dittadi for helpful discussions regarding experiments, Attila Juhos for pointing out an issue with Thm. 1, Amin Charusaie, Michel Besserve, and Simon Buchholz for helpful technical discussions, and Zac Cranko for theoretical efforts in the early stages of the project.

This work was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A, 01IS18039B. WB acknowledges financial support via an Emmy Noether Grant funded by the German Research Foundation (DFG) under grant no. BR 6382/1-1 and via the Open Philantropy Foundation funded by the Good Ventures Foundation. WB is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 – Project number 390727645. This research utilized compute resources at the Tübingen Machine Learning Cloud, DFG FKZ INST 37/1057-1 FUGG. The authors

thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for supporting RSZ and YS.

References

Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V. F., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Çaglar Gülçehre, Song, H. F., Ballard, A. J., Gilmer, J., Dahl, G. E., Vaswani, A., Allen, K. R., Nash, C., Langston, V., Dyer, C., Heess, N. M. O., Wierstra, D., Kohli, P., Botvinick, M. M., Vinyals, O., Li, Y., and Pascanu, R. Relational inductive biases, deep learning, and graph networks. *ArXiv*, abs/1806.01261, 2018. [Cited on page 1.]

Besserve, M., Shajarisales, N., Schölkopf, B., and Janzing, D. Group invariance principles for causal generative models. In *AISTATS*, volume 84 of *Proceedings of Machine Learning Research*, pp. 557–565, 2018. [Cited on page 6.]

Besserve, M., Sun, R., Janzing, D., and Schölkopf, B. A theory of independent mechanisms for extrapolation in generative models. In *AAAI*, pp. 6741–6749, 2021. [Cited on page 6.]

Biza, O., van Steenkiste, S., Sajjadi, M. S., Elsayed, G. F., Mahendran, A., and Kipf, T. Invariant slot attention: Object discovery with slot-centric reference frames. *ArXiv preprint*, abs/2302.04973, 2023. [Cited on page 1.]

Buchholz, S., Besserve, M., and Schölkopf, B. Function classes for identifiable nonlinear independent component analysis. In *NeurIPS*, 2022. [Cited on page 7.]

Burgess, C. P., Higgins, I., Pal, A., Matthey, L., Watters, N., Desjardins, G., and Lerchner, A. Understanding disentangling in β-vae. *ArXiv preprint*, abs/1804.03599, 2018. [Cited on page 22.]

Burgess, C. P., Matthey, L., Watters, N., Kabra, R., Higgins, I., Botvinick, M., and Lerchner, A. Monet: Unsupervised scene decomposition and representation. *ArXiv preprint*, abs/1901.11390, 2019. [Cited on pages 1 and 8.]

Chen, H., Venkatesh, R. M., Friedman, Y., Wu, J., Tenenbaum, J. B., Yamins, D. L. K., and Bear, D. Unsupervised segmentation in real-world images via spelke object inference. In *European Conference on Computer Vision*, 2022. [Cited on page 9.]

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. [Cited on page 1.]

Daniusis, P., Janzing, D., Mooij, J. M., Zscheischler, J., Steudel, B., Zhang, K., and Schölkopf, B. Inferring deterministic causal relations. In *UAI 2010, Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence, Catalina Island, CA, USA, July 8-11, 2010*, pp. 143–150, 2010. [Cited on pages 4 and 6.]

Dehaene, S. *How We Learn: Why Brains Learn Better Than Any Machine... for Now*. 2020. [Cited on page 1.]

Dittadi, A., Papa, S. S., Vita, M. D., Schölkopf, B., Winther, O., and Locatello, F. Generalization and robustness implications in object-centric learning. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5221–5285, 2022. [Cited on pages 5, 7, 22, and 23.]

Elsayed, G. F., Mahendran, A., van Steenkiste, S., Greff, K., Mozer, M. C., and Kipf, T. Savi++: Towards end-to-end object-centric learning from real-world videos. In *NeurIPS*, 2022. [Cited on pages 1, 5, and 9.]

Engelcke, M., Jones, O. P., and Posner, I. Reconstruction bottlenecks in object-centric generative models. *ArXiv preprint*, abs/2007.06245, 2020a. [Cited on page 5.]

Engelcke, M., Kosiorek, A. R., Jones, O. P., and Posner, I. GENESIS: generative scene inference and sampling with object-centric latent representations. In *ICLR*, 2020b. [Cited on page 6.]

Engelcke, M., Jones, O. P., and Posner, I. GENESIS-V2: inferring unordered object representations without iterative refinement. In *NeurIPS*, pp. 8085–8094, 2021. [Cited on page 6.]

Fodor, J. A. and Pylyshyn, Z. W. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2): 3–71, 1988. [Cited on page 1.]

Gerstenberg, T. and Tenenbaum, J. B. Intuitive theories. *Oxford handbook of causal reasoning*, pp. 515–548, 2017. [Cited on page 1.]

Gerstenberg, T., Goodman, N. D., Lagnado, D. A., and Tenenbaum, J. B. A counterfactual simulation model of causal judgments for physical events. *Psychological Review*, 128(5):936, 2021. [Cited on page 1.]

Gopnik, A., Glymour, C., Sobel, D. M., Schulz, L. E., Kushnir, T., and Danks, D. A theory of causal learning in children: causal maps and bayes nets. *Psychological review*, 111(1):3, 2004. [Cited on page 1.]

Goyal, A. and Bengio, Y. Inductive biases for deep learning of higher-level cognition. *Proceedings of the Royal Society A*, 478(2266):20210068, 2022. [Cited on page 1.]

Green, E. J. A theory of perceptual objects. *Philosophy and Phenomenological Research*, 99(3):663–693, 2019. [Cited on page 2.]

Greff, K., Srivastava, R. K., and Schmidhuber, J. Binding via reconstruction clustering. *ArXiv preprint*, abs/1511.06418, 2015. [Cited on page 6.]

Greff, K., van Steenkiste, S., and Schmidhuber, J. Neural expectation maximization. In *NIPS*, pp. 6691–6701, 2017. [Cited on page 6.]

Greff, K., Kaufman, R. L., Kabra, R., Watters, N., Burgess, C., Zoran, D., Matthey, L., Botvinick, M. M., and Lerchner, A. Multi-object representation learning with iterative variational inference. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2424–2433, 2019. [Cited on pages 1 and 6.]

Greff, K., Van Steenkiste, S., and Schmidhuber, J. On the binding problem in artificial neural networks. *ArXiv preprint*, abs/2012.05208, 2020. [Cited on pages 1 and 2.]

Gresele, L., Rubenstein, P. K., Mehrjou, A., Locatello, F., and Schölkopf, B. The incomplete rosetta stone problem: Identifiability results for multi-view nonlinear ICA. In *Proceedings of the Thirty-Fifth Conference on Uncertainty in Artificial Intelligence, UAI 2019, Tel Aviv, Israel, July 22-25, 2019*, volume 115 of *Proceedings of Machine Learning Research*, pp. 217–227, 2019. [Cited on page 7.]

Gresele, L., von Kügelgen, J., Stimper, V., Schölkopf, B., and Besserve, M. Independent mechanism analysis, a new concept? *Advances in Neural Information Processing Systems*, 34:28233–28248, 2021. [Cited on pages 4, 6, and 7.]

Guo, S., Tóth, V., Schölkopf, B., and Huszár, F. Causal de Finetti: On the identification of invariant causal structure in exchangeable data. *ArXiv preprint*, abs/2203.15756, 2022. [Cited on page 6.]

Hälvä, H. and Hyvärinen, A. Hidden markov nonlinear ICA: unsupervised learning from nonstationary time series. In *Proceedings of the Thirty-Sixth Conference on Uncertainty in Artificial Intelligence, UAI 2020, virtual online, August 3-6, 2020*, volume 124 of *Proceedings of Machine Learning Research*, pp. 939–948, 2020. [Cited on page 7.]

Hälvä, H., Corff, S. L., Lehéricy, L., So, J., Zhu, Y., Gassiat, E., and Hyvärinen, A. Disentangling identifiable features from noisy data with structured nonlinear ICA. In *NeurIPS*, pp. 1624–1633, 2021. [Cited on page 7.]

He, K., Gkioxari, G., Dollár, P., and Girshick, R. B. Mask R-CNN. In *ICCV*, pp. 2980–2988, 2017. [Cited on page 1.]

Heess, N. M. O. *Learning generative models of mid-level structure in natural images*. PhD thesis, The University of Edinburgh, 2012. [Cited on page 6.]

Hinton, G. E. How to represent part-whole hierarchies in a neural network. *Neural computation*, pp. 1–40, 2021. [Cited on page 9.]

Horn, R. A. and Johnson, C. R. *Matrix analysis*. 2012. [Cited on page 16.]

Hyvärinen, A. and Hoyer, P. O. Emergence of phase- and shift-invariant features by decomposition of natural images into independent feature subspaces. *Neural Comput.*, 12(7):1705–1720, 2000. [Cited on page 7.]

Hyvärinen, A. and Morioka, H. Unsupervised feature extraction by time-contrastive learning and nonlinear ICA. In *NIPS*, pp. 3765–3773, 2016. [Cited on pages 2 and 7.]

Hyvärinen, A. and Morioka, H. Nonlinear ICA of temporally dependent stationary sources. In *AISTATS*, volume 54 of *Proceedings of Machine Learning Research*, pp. 460–469, 2017. [Cited on pages 2 and 7.]

Hyvärinen, A. and Perkiö, J. Learning to segment any random vector. *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pp. 4167–4172, 2006. [Cited on page 6.]

Hyvärinen, A., Sasaki, H., and Turner, R. E. Nonlinear ICA using auxiliary variables and generalized contrastive learning. In *AISTATS*, volume 89 of *Proceedings of Machine Learning Research*, pp. 859–868, 2019. [Cited on pages 2 and 7.]

Hyvärinen, A. and Pajunen, P. Nonlinear independent component analysis: Existence and uniqueness results. *Neural Networks*, 12(3):429–439, 1999. ISSN 0893-6080. [Cited on page 2.]

Janzing, D. Causal versions of maximum entropy and principle of insufficient reason. *Journal of Causal Inference*, 9(1):285–301, 2021. [Cited on page 6.]

Janzing, D. and Schölkopf, B. Causal inference using the algorithmic markov condition. *IEEE Transactions on Information Theory*, 56(10):5168–5194, 2010. [Cited on pages 4 and 6.]

Janzing, D., Hoyer, P. O., and Schölkopf, B. Telling cause from effect based on high-dimensional observations. In *ICML*, pp. 479–486, 2010. [Cited on page 6.]

Janzing, D., Mooij, J., Zhang, K., Lemeire, J., Zscheischler, J., Daniušis, P., Steudel, B., and Schölkopf, B. Information-geometric approach to inferring causal directions. *Artificial Intelligence*, 182:1–31, 2012. [Cited on pages 4 and 6.]

Kabra, R., Burgess, C., Matthey, L., Kaufman, R. L., Greff, K., Reynolds, M., and Lerchner, A. Multi-object datasets. https://github.com/deepmind/multi-object-datasets/, 2019. [Cited on page 22.]

Karazija, L., Laina, I., and Rupprecht, C. Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation. *ArXiv preprint*, abs/2111.10265, 2021. [Cited on page 1.]

Khemakhem, I., Kingma, D. P., Monti, R. P., and Hyvärinen, A. Variational autoencoders and nonlinear ICA: A unifying framework. In *AISTATS*, volume 108 of *Proceedings of Machine Learning Research*, pp. 2207–2217, 2020a. [Cited on pages 2 and 7.]

Khemakhem, I., Monti, R. P., Kingma, D. P., and Hyvärinen, A. Ice-beem: Identifiable conditional energy-based deep models based on nonlinear ICA. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b. [Cited on page 2.]

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015. [Cited on page 22.]

Kipf, T., Elsayed, G. F., Mahendran, A., Stone, A., Sabour, S., Heigold, G., Jonschkowski, R., Dosovitskiy, A., and Greff, K. Conditional object-centric learning from video. In *ICLR*, 2022. [Cited on pages 1 and 9.]

Kipf, T. N., van der Pol, E., and Welling, M. Contrastive learning of structured world models. In *ICLR*, 2020. [Cited on page 1.]

Klindt, D. A., Schott, L., Sharma, Y., Ustyuzhaninov, I., Brendel, W., Bethge, M., and Paiton, D. M. Towards nonlinear disentanglement in natural data with temporal sparse coding. In *ICLR*, 2021. [Cited on pages 2 and 7.]

Koffka, K. *Principles Of Gestalt Psychology*. 1936. [Cited on pages 2 and 9.]

Kuhn, H. W. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. [Cited on pages 7 and 23.]

Lachapelle, S. and Lacoste-Julien, S. Partial disentanglement via mechanism sparsity. In *UAI 2022 Workshop on Causal Representation Learning*, 2022. [Cited on page 7.]

Lachapelle, S., Rodriguez, P., Sharma, Y., Everett, K. E., Le Priol, R., Lacoste, A., and Lacoste-Julien, S. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ica. In *First Conference on Causal Learning and Reasoning*, 2021. [Cited on page 7.]

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and brain sciences*, 40, 2017. [Cited on page 1.]

Lin, Z., Wu, Y., Peri, S. V., Sun, W., Singh, G., Deng, F., Jiang, J., and Ahn, S. SPACE: unsupervised object-oriented scene representation via spatial attention and decomposition. In *ICLR*, 2020. [Cited on page 1.]

Locatello, F., Vincent, D., Tolstikhin, I., Rätsch, G., Gelly, S., and Schölkopf, B. Competitive training of mixtures of independent deep generative models. *ArXiv preprint*, abs/1804.11130, 2018. [Cited on page 6.]

Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. Challenging common assumptions in the unsupervised learning of disentangled representations. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124, 2019. [Cited on page 2.]

Locatello, F., Weissenborn, D., Unterthiner, T., Mahendran, A., Heigold, G., Uszkoreit, J., Dosovitskiy, A., and Kipf, T. Object-centric learning with slot attention. In *NeurIPS*, 2020. [Cited on pages 1, 5, 7, 8, and 22.]

Marcus, G. F. *The Algebraic Mind: Integrating Connectionism and Cognitive Science*. 2001. [Cited on page 1.]

Moran, G. E., Sridhar, D., Wang, Y., and Blei, D. M. Identifiable variational autoencoders via sparse decoding. *ArXiv preprint*, abs/2110.10804, 2021. [Cited on page 7.]

Papa, S., Winther, O., and Dittadi, A. Inductive biases for object-centric representations in the presence of complex textures. In *UAI 2022 Workshop on Causal Representation Learning*, 2022. [Cited on page 1.]

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E. Z., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, pp. 8024–8035, 2019. [Cited on page 22.]

Pearl, J. *Causality*. 2 edition, 2009. [Cited on page 6.]

Peters, B. and Kriegeskorte, N. Capturing the objects of vision with neural networks. *Nature Human Behaviour*, 5(9):1127–1144, 2021. [Cited on page 1.]

Peters, J., Janzing, D., and Schölkopf, B. *Elements of causal inference: foundations and learning algorithms*. 2017. [Cited on pages 2, 4, and 6.]

Reizinger, P., Gresele, L., Brady, J., von Kügelgen, J., Zietlow, D., Schölkopf, B., Martius, G., Brendel, W., and Besserve, M. Embrace the gap: VAEs perform independent mechanism analysis. In *Advances in Neural Information Processing Systems*, 2022. [Cited on page 7.]

Ronneberger, O., Fischer, P., and Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234–241, 2015. [Cited on page 1.]

Roux, N. L., Heess, N. M. O., Shotton, J., and Winn, J. M. Learning a generative model of images by factoring appearance and shape. *Neural Computation*, 23:593–650, 2011. [Cited on page 6.]

Sajjadi, M. S. M., Duckworth, D., Mahendran, A., van Steenkiste, S., Pavetic, F., Lucic, M., Guibas, L. J., Greff, K., and Kipf, T. Object scene representation transformer. In *NeurIPS*, 2022. [Cited on pages 1 and 5.]

Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., and Mooij, J. M. On causal and anticausal learning. In *ICML*, 2012. [Cited on page 6.]

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Toward causal representation learning. *Proceedings of the IEEE*, 109(5): 612–634, 2021. [Cited on page 6.]

Seitzer, M., Horn, M., Zadaianchuk, A., Zietlow, D., Xiao, T., Simon-Gabriel, C.-J., He, T., Zhang, Z., Schölkopf, B., Brox, T., and Locatello, F. Bridging the gap to real-world object-centric learning. In *The Eleventh International Conference on Learning Representations*, 2023. [Cited on pages 1 and 5.]

Shajarisales, N., Janzing, D., Schölkopf, B., and Besserve, M. Telling cause from effect in deterministic linear dynamical systems. In *ICML*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 285–294, 2015. [Cited on page 6.]

Singh, G., Deng, F., and Ahn, S. Illiterate DALL-E learns to compose. In *ICLR*, 2022a. [Cited on page 1.]

Singh, G., Wu, Y., and Ahn, S. Simple unsupervised object-centric learning for complex and naturalistic videos. In *NeurIPS*, 2022b. [Cited on pages 1 and 5.]

Spelke, E. S. Principles of object perception. *Cogn. Sci.*, 14: 29–56, 1990. [Cited on pages 2 and 9.]

Spelke, E. S. What makes us smart? core knowledge and natural language. *Language in mind: Advances in the study of language and thought*, pp. 277–311, 2003. [Cited on page 1.]

Spelke, E. S. and Kinzler, K. D. Core knowledge. *Developmental science*, 10(1):89–96, 2007. [Cited on page 1.]

Spirtes, P., Glymour, C., and Scheines, R. *Causation, Prediction, and Search*, volume 1. 2001. [Cited on page 6.]

Tangemann, M., Schneider, S., von Kügelgen, J., Locatello, F., Gehler, P. V., Brox, T., Kuemmerer, M., Bethge, M., and Schölkopf, B. Unsupervised object learning via common fate. In *2nd Conference on Causal Learning and Reasoning*, 2023. [Cited on pages 6 and 9.]

Tenenbaum, J. B., Kemp, C., Griffiths, T. L., and Goodman, N. D. How to grow a mind: Statistics, structure, and abstraction. *Science*, 331:1279 – 1285, 2011. [Cited on page 1.]

Träuble, F., Creager, E., Kilbertus, N., Locatello, F., Dittadi, A., Goyal, A., Schölkopf, B., and Bauer, S. On disentangled representations learned from correlated data. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 10401–10412, 2021. [Cited on page 6.]

van Steenkiste, S., Chang, M., Greff, K., and Schmidhuber, J. Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. In *ICLR (Poster)*, 2018. [Cited on page 6.]

von Kügelgen, J., Ustyuzhaninov, I., Gehler, P., Bethge, M., and Schölkopf, B. Towards causal generative scene models via competition of experts. [Cited on page 6.]

von Kügelgen, J., Sharma, Y., Gresele, L., Brendel, W., Schölkopf, B., Besserve, M., and Locatello, F. Self-supervised learning with data augmentations provably isolates content from style. volume 34, pp. 16451–16467, 2021. [Cited on page 7.]

Watters, N., Matthey, L., Borgeaud, S., Kabra, R., and Lerchner, A. Spriteworld: A flexible, configurable reinforcement learning environment. https://github.com/deepmind/spriteworld/, 2019. [Cited on pages 8 and 22.]

Weis, M. A., Chitta, K., Sharma, Y., Brendel, W., Bethge, M., Geiger, A., and Ecker, A. S. Benchmarking unsupervised object representations for video sequences. *J. Mach. Learn. Res.*, 22:183:1–183:61, 2021. [Cited on page 1.]

Yang, X., Wang, Y., Sun, J., Zhang, X., Zhang, S., Li, Z., and Yan, J. Nonlinear ICA using volume-preserving transformations. In *ICLR*, 2022. [Cited on page 7.]

Yang, Y. and Yang, B. Promising or elusive? unsupervised object segmentation from real-world single images. In *NeurIPS*, 2022. [Cited on page 1.]

Zheng, Y., Ng, I., and Zhang, K. On the identifiability of nonlinear ICA: Sparsity and beyond. In *Advances in Neural Information Processing Systems*, 2022. [Cited on page 7.]

Zimmermann, R. S., Sharma, Y., Schneider, S., Bethge, M., and Brendel, W. Contrastive learning inverts the data generating process. In *ICML*, volume 139 of *Proceedings of Machine Learning Research*, pp. 12979–12990, 2021. [Cited on pages 2 and 7.]

A Proofs

In this section, we present the proofs for the results presented in the main text. First, we recall our notation:

Notation. N will denote the dimensionality of observations $\mathbf{x}$, K the number of latent slots, and M the dimensionality of each latent slot $\mathbf{z_k}$. For $n \in \mathbb{N}$, $[n]$ will denote the set of natural numbers from 1 to n, i.e., $[n] := \{1, \dots, n\}$. If $\mathbf{f}$ is a function with n component functions, then $\mathbf{f}_S$ will denote the restriction of $\mathbf{f}$ to the component functions indexed by $S \subseteq [n]$, i.e. $\mathbf{f}_S := (f_s)_{s \in S}$ where $\mathbf{f}_S$ is ordered according to the natural ordering of the elements of S. Additionally, when restricting $\mathbf{f}$ to the component functions indexed by $I_k(\mathbf{z})$, defined according to Eq. (3), we will drop the dependence on $\mathbf{z}$ for notional convenience i.e. $\mathbf{f}_{I_k}(\mathbf{z}) := \mathbf{f}_{I_k(\mathbf{z})}(\mathbf{z})$. For functions $\mathbf{f}, \hat{\mathbf{f}}$, we will use $I_k(\mathbf{z}), \hat{I}_k(\hat{\mathbf{z}})$, respectively, to distinguish between the indices defined for each function according to Eq. (3). Lastly, we will slightly abuse notation and use $\mathbf{0}$ to denote both the zero vector and a matrix whose entries are all 0.

We begin by proving several lemmata which will be leveraged for our main theoretical result. We start with the intuitive result that sub-mechanisms from different latent slots are independent in the sense of Defn. 4.

Lemma 1 (Sub-Mechanisms of Distinct Mechanisms are Independent)**.** *Let $\mathbf{f}$ be a diffeomorphism that is compositional (Defn. 1), and let $S_1, S_2 \subseteq [N]$ be nonempty. $\forall \mathbf{z} \in \mathcal{Z}, k \in [K]$, if $S_1 \subseteq I_k(\mathbf{z})$, $S_2 \cap I_k(\mathbf{z}) = \varnothing$, then sub-mechanisms $\mathbf{Jf}_{S_1}(\mathbf{z})$, $\mathbf{Jf}_{S_2}(\mathbf{z})$ are independent in the sense of Defn. 4.*

Proof. From the definition of $I_k(\mathbf{z})$ in Eq. (3) it follows that:

$$\forall n \in [N]: \quad \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) \neq \mathbf{0} \implies n \in I_k(\mathbf{z}).$$

Since $S_1 \subseteq I_k(\mathbf{z})$, we know that $\forall n \in S_1 : \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) \neq \mathbf{0}$. Further, since $S_2 \cap I_k(\mathbf{z}) = \varnothing$ it means that $\forall n \in S_2 : \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) = \mathbf{0}$. Put differently, this means that rows of $\mathbf{Jf}_{S_1}(\mathbf{z})$ are non-zero for those rows where $\mathbf{Jf}_{S_2}(\mathbf{z})$ vanishes and vice versa. Therefore, one cannot represent any column of $\mathbf{Jf}_{S_1}(\mathbf{z})$ as a linear combination of those of $\mathbf{Jf}_{S_2}(\mathbf{z})$. Hence,

$$\operatorname{rank}\left(\mathbf{Jf}_{S_1}(\mathbf{z})\right) + \operatorname{rank}\left(\mathbf{Jf}_{S_2}(\mathbf{z})\right) = \operatorname{rank}\left(\left[\mathbf{Jf}_{S_1}(\mathbf{z}); \mathbf{Jf}_{S_2}(\mathbf{z})\right]\right),$$

where $[\cdot\,;\cdot]$ denotes vertical concatenation. Note that the RHS is equal to $\mathbf{Jf}_{S_1 \cup S_2}(\mathbf{z})$ up to permutations of rows (which do not change the rank). Thus, Eq. (5) holds for S_1, S_2 showing that $\mathbf{Jf}_{S_1}(\mathbf{z})$, $\mathbf{Jf}_{S_2}(\mathbf{z})$ are independent in the sense of Defn. 4. □

We next show that the rank of each sub-mechanism is less than or equal to the latent slot-dimension dimension, M.

Lemma 2. *Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a diffeomorphism that is compositional (Defn. 1). $\forall \mathbf{z} \in \mathcal{Z}, k \in [K]$, if $S \subseteq I_k(\mathbf{z})$ is non-empty:*

$$\operatorname{rank}\left(\mathbf{Jf}_S(\mathbf{z})\right) \leq M. \tag{8}$$

Proof. Since $S \subseteq I_k(\mathbf{z})$, then by compositionality of $\mathbf{f}$

$$\forall \mathbf{z} \in \mathcal{Z}, s \in S, j \in [K] \setminus \{k\}: \quad \frac{\partial f_s}{\partial \mathbf{z}_j}(\mathbf{z}) = \mathbf{0}. \tag{9}$$

Thus, $\mathbf{Jf}_S(\mathbf{z})$ has at most M non-zero columns (those corresponding to the non-zero partials w.r.t. $\mathbf{z}_k$) which implies $\operatorname{rank}(\mathbf{Jf}_S(\mathbf{z})) \leq M$. □

We now show that the rank of each mechanism is equal to the latent slot-dimension M.

Lemma 3. *Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a diffeomorphism that is compositional (Defn. 1). Then $\forall \mathbf{z} \in \mathcal{Z}, k \in [K]$:*

$$\operatorname{rank}(\mathbf{Jf}_{I_k}(\mathbf{z})) = M.$$

Proof. First note $\mathbf{f}$ is a diffeomorphism and is thus invertible. Therefore, $\mathbf{Jf}$ must be invertible and thus have full column-rank, i.e., $\forall \mathbf{z} \in \mathcal{Z} : \operatorname{rank}(\mathbf{Jf}(\mathbf{z})) = MK$.

Next, $\forall \mathbf{z} \in \mathcal{Z}, k \in [K]$, let $I_k^C := [N] \setminus I_k$ denote the complement of I_k in $[N]$ such that $I_k^C \cap I_k = \varnothing$. Thus, by Lemma 1, the corresponding sub-mechanisms are independent:

$$\forall \mathbf{z} \in \mathcal{Z}, k \in [K]: \quad \operatorname{rank}(\mathbf{Jf}(\mathbf{z})) = \operatorname{rank}(\mathbf{Jf}_{I_k}(\mathbf{z})) + \operatorname{rank}(\mathbf{Jf}_{I_k^C}(\mathbf{z})) = MK. \tag{10}$$

By compositionality of $\mathbf{f}$,

$$\forall \mathbf{z} \in \mathcal{Z}, j \in [K] \setminus \{k\} : \quad \frac{\partial \mathbf{f}_{I_k}}{\partial \mathbf{z}_j}(\mathbf{z}) = \mathbf{0}. \tag{11}$$

Thus, $\mathbf{J}\mathbf{f}_{I_k}(\mathbf{z})$ has at most M non-zero columns implying that $\operatorname{rank}(\mathbf{J}\mathbf{f}_{I_k}(\mathbf{z})) \leq M$. Furthermore, by definition,

$$\forall \mathbf{z} \in \mathcal{Z} : \quad \frac{\partial \mathbf{f}_{I_k^C}}{\partial \mathbf{z}_k}(\mathbf{z}) = \mathbf{0}, \tag{12}$$

which means that $\mathbf{J}\mathbf{f}_{I_k^C}(\mathbf{z})$ has at most $(K-1)M$ non-zero columns implying $\operatorname{rank}(\mathbf{J}\mathbf{f}_{I_k^C}(\mathbf{z})) \leq (K-1)M$. Inserting this result in Eq. (10) yields

$$\forall \mathbf{z} \in \mathcal{Z}, k \in [K] : \quad M \leq MK - \operatorname{rank}(\mathbf{J}\mathbf{f}_{I_k^C}(\mathbf{z})) = \operatorname{rank}(\mathbf{J}\mathbf{f}_{I_k}(\mathbf{z})) \leq M, \tag{13}$$

which can only be true if $\operatorname{rank}(\mathbf{J}\mathbf{f}_{I_k}(\mathbf{z})) = M$. □

Next, we show that for ground-truth generator $\mathbf{f}$ and inferred generator $\hat{\mathbf{f}}$, the sub-mechanisms at a given point with respect to the same pixel subset S will be have the same rank.

Lemma 4. *Let $\mathbf{f}, \hat{\mathbf{f}} : \mathcal{Z} \to \mathcal{X}$ be diffeomorphisms with inverses $\mathbf{g}, \hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$, respectively. Then $\forall \mathbf{z} \in \mathcal{Z}, S \subseteq [N]$ s.t. $S \neq \varnothing$, $\operatorname{rank}(\mathbf{J}\mathbf{f}_S(\mathbf{z})) = \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_S(\hat{\mathbf{z}}))$, where $\hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z}))$.*

Proof. First, we introduce the function

$$\mathbf{h} := \hat{\mathbf{g}} \circ \mathbf{f} : \mathcal{Z} \to \mathcal{Z} \quad \text{s.t.} \quad \hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z})) = \mathbf{h}(\mathbf{z}),$$

We can express $\mathbf{f}$ as $\mathbf{f} = \hat{\mathbf{f}} \circ \hat{\mathbf{g}} \circ \mathbf{f} = \hat{\mathbf{f}} \circ \mathbf{h}$. Thus, if $S \subseteq [N], S \neq \varnothing$, $\mathbf{f}_S = \hat{\mathbf{f}}_S \circ \mathbf{h}$. Therefore,

$$\forall \mathbf{z} \in \mathcal{Z}, \ \operatorname{rank}(\mathbf{J}\mathbf{f}_S(\mathbf{z})) = \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_S(\hat{\mathbf{z}})\mathbf{J}\mathbf{h}(\mathbf{z})). \tag{14}$$

Because $\mathbf{h}$ is a diffeomorphism, $\mathbf{J}\mathbf{h}(\mathbf{z})$ is invertible. Thus $\operatorname{rank}(\mathbf{A}\mathbf{J}\mathbf{h}(\mathbf{z})) = \operatorname{rank}(\mathbf{A})$ for any matrix $\mathbf{A}$ s.t. $\mathbf{A}\mathbf{J}\mathbf{h}(\mathbf{z})$ is defined (Horn & Johnson, 2012, Section 0.4.6). Therefore, by Eq. (14):

$$\forall \mathbf{z} \in \mathcal{Z}, \ \operatorname{rank}(\mathbf{J}\mathbf{f}_S(\mathbf{z})) = \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_S(\hat{\mathbf{z}})). \tag{15}$$

□

We now prove several propositions which will be used to build our main result (Thm. 1). Firstly, we show that each inferred latent slot depends on at least one ground-truth slot.

Proposition 1. *Let $\mathcal{Z}$ be a latent space, $\mathcal{X}$ an observation space, and $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ a diffeomorphism that is compositional (Defn. 1). Let $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ be a diffeomorphism and $\hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z})), \forall \mathbf{z} \in \mathcal{Z}$. Then, $\forall \mathbf{z} \in \mathcal{Z}, i \in [K], \exists j \in [K] : \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_i}(\mathbf{z}) \neq \mathbf{0}$.*

Proof. We first define the function

$$\mathbf{h} := \hat{\mathbf{g}} \circ \mathbf{f} : \mathcal{Z} \to \mathcal{Z} \quad \text{s.t.} \quad \hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z})) = \mathbf{h}(\mathbf{z}).$$

As $\hat{\mathbf{g}}$ and $\mathbf{f}$ are both diffeomorphisms, $\mathbf{h}$ is also a diffeomorphism.

Note that $\forall \mathbf{z} \in \mathcal{Z}, \mathbf{J}\mathbf{h}(\mathbf{z})$ is a square matrix. Furthermore, because $\mathbf{h}$ is a diffeomorphism, it follows that $\forall \mathbf{z} \in \mathcal{Z}$, $\mathbf{J}\mathbf{h}(\mathbf{z})$ is full rank. This implies $\mathbf{J}\mathbf{h}(\mathbf{z})$ must have all non-zero columns, which implies

$$\forall \mathbf{z} \in \mathcal{Z}, i \in [K], \exists j \in [K] : \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_i}(\mathbf{z}) \neq \mathbf{0}.$$

□

Next, we show that each inferred latent slot generates the same pixels as at most one ground-truth slot.

Proposition 2. *Let $\mathcal{Z}$ be a latent space and $\mathcal{X}$ an observation space defined as in § 2. Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a diffeomorphism that is compositional (Defn. 1) with irreducible mechanisms (Defn. 5). Let $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ be a diffeomorphism with inverse $\hat{\mathbf{f}} : \mathcal{Z} \to \mathcal{X}$ that is compositional (Defn. 1). Then $\forall \mathbf{z} \in \mathcal{Z}, j \in [K]$, there exists exactly one $i \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing$, where $\hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z}))$*

Proof. Our goal is to show that $\hat{\mathbf{f}}$ maps each inferred latent slot $\hat{\mathbf{z}}_j$ to pixels generated by exactly one ground-truth latent slot $\mathbf{z}_i$.

Step 1 We will first show that $\hat{\mathbf{f}}$ maps each inferred latent slot $\hat{\mathbf{z}}_j$ to pixels generated by at least one ground-truth latent slot $\mathbf{z}_i$. More precisely, we aim to show:

$$\forall \mathbf{z} \in \mathcal{Z}, j \in [K], \exists i \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing. \tag{16}$$

Suppose for a contradiction to Eq. (16) that:

$$\exists \mathbf{z}^* \in \mathcal{Z}, j \in [K], \nexists i \in [K] : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) \neq \varnothing. \tag{17}$$

We will show that this assumption leads to a contradiction and, hence, is false. Let, $\mathbf{z}^*$ denote the value for which Eq. (17) holds. Eq. (17) coupled with the definition of $I_i(\mathbf{z}^*)$ in Eq. (3) imply that there exists pixels which depend on $\hat{\mathbf{z}}^*$ under $\hat{\mathbf{f}}$ but not on $\mathbf{z}^*$ under $\mathbf{f}$. More precisely,

$$\exists i \in \hat{I}_j(\hat{\mathbf{z}}^*) : \mathbf{J}\hat{\mathbf{f}}_i(\hat{\mathbf{z}}^*) \neq \mathbf{0}, \quad \nexists i \in \hat{I}_j(\hat{\mathbf{z}}^*) : \mathbf{J}\mathbf{f}_i(\mathbf{z}^*) \neq \mathbf{0} \tag{18}$$

This then implies that:

$$\operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{\hat{I}_j}(\hat{\mathbf{z}}^*)) \neq \mathbf{0}, \quad \operatorname{rank}(\mathbf{J}\mathbf{f}_{\hat{I}_j}(\mathbf{z}^*)) = \mathbf{0} \tag{19}$$

which contradicts the equality of Jacobian ranks between $\mathbf{f}$ and $\hat{\mathbf{f}}$ stated in Lemma 4. Thus, our assumed contradiction in Eq. (17) cannot hold and we conclude that Eq. (16) must hold true.

Step 2 We will now show that $\hat{\mathbf{f}}$ maps each inferred latent slot $\hat{\mathbf{z}}_j$ to pixels generated by at most one ground-truth latent slot $\mathbf{z}_i$. More precisely, for $C := \{ P \subseteq [K] : |P| > 1 \}$ we aim to show:

$$\forall \mathbf{z} \in \mathcal{Z}, j \in [K], \nexists P \in C : \qquad i \in P \implies \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing. \tag{20}$$

Suppose for a contradiction to Eq. (20) that:

$$\exists \mathbf{z}^* \in \mathcal{Z}, j \in [K], P \in C : \qquad i \in P \implies \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) \neq \varnothing. \tag{21}$$

We will let $\mathbf{z}^*$ denote the value for which Eq. (21) holds and without loss of generality let $j = 1$.

Step 2.1 First, $\forall i \in P$ we define the sets:

$$O_{i,1} := \{ j \in I_i(\mathbf{z}^*) \mid j \in \hat{I}_1(\hat{\mathbf{z}}^*) \}, \;\; O_{i,2} := \{ j \in I_i(\mathbf{z}^*) \mid j \notin \hat{I}_1(\hat{\mathbf{z}}^*) \}, \tag{22}$$

Intuitively, the set $O_{i,1}$ represents the pixels which are a function of both ground-truth latent slot $\mathbf{z}_i^*$ and inferred slot $\hat{\mathbf{z}}_1^*$, while $O_{i,2}$ represents the pixels which are a function of $\mathbf{z}_i^*$ but not $\hat{\mathbf{z}}_1^*$. Our aim is now to show that for all $\forall i \in P$, the sets $O_{i,1}, O_{i,2}$ form a partition of $I_i(\mathbf{z}^*)$.

By Eq. (22), $\forall i \in P$, $O_{i,1} \cup O_{i,2} = I_i(\mathbf{z}^*)$, and $O_{i,1} \cap O_{i,2} = \varnothing$. We thus only need to show that $O_{i,1}, O_{i,2} \neq \varnothing$.

We first note that by our assumed contradiction in Eq. (21), there are pixels which are a function of both ground-truth slot $\mathbf{z}_i^*$ and inferred slot $\hat{\mathbf{z}}_1^*$ i.e.:

$$\forall i \in P, \exists j \in I_i(\mathbf{z}^*) : j \in \hat{I}_1(\hat{\mathbf{z}}^*) \implies j \in O_{i,1} \implies O_{i,1} \neq \varnothing. \tag{23}$$

We will now show that $\forall i \in P, \; O_{i,2} \neq \varnothing$. Suppose for a contradiction that

$$\exists i \in P : O_{i,2} = \varnothing, \tag{24}$$

This implies that $I_i(\mathbf{z}^*) = O_{i,1}$ as $I_i(\mathbf{z}^*) = O_{i,1} \cup O_{i,2} = O_{i,1} \cup \varnothing$. Further, Eq. (22) implies that $O_{i,1} \subseteq \hat{I}_1(\hat{\mathbf{z}}^*)$ thus $O_{i,1} = I_i(\mathbf{z}^*) \subseteq \hat{I}_1(\hat{\mathbf{z}}^*)$.

Next, consider another ground-truth slot $\mathbf{z}_k^*$ where $k \neq i \in P$. As previously established, $O_{k,1} \neq \varnothing$. Moreover, by Eq. (22), $O_{k,1} \subseteq \hat{I}_1(\hat{\mathbf{z}}^*)$. Thus, $A := I_i(\mathbf{z}^*) \cup O_{k,1} \subseteq \hat{I}_1(\hat{\mathbf{z}}^*)$. Now, note that because $\hat{\mathbf{f}}$ is compositional, Lemma 2 implies that the rank of the sub-mechanism defined by $A \leq M$. When coupled with the equality of Jacobian ranks between $\mathbf{f}$ and $\hat{\mathbf{f}}$ stated in Lemma 4, we get:

$$\operatorname{rank}(\mathbf{J}\mathbf{f}_A(\mathbf{z}^*)) = \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_A(\hat{\mathbf{z}}^*)) \leq M. \tag{25}$$

Moreover, according to Eq. (22), $O_{k,1} \subseteq I_k(\mathbf{z}^*)$. By compositionality of $\mathbf{f}$, it thus follows that $O_{k,1} \cap I_i(\mathbf{z}^*) = \varnothing$ since $i \neq k$. Therefore, by Lemma 1, we know the sub-mechanisms defined by $I_i(\mathbf{z}^*)$ and $O_{k,1}$ are independent such that

$$\operatorname{rank}(\mathbf{J}\mathbf{f}_A(\mathbf{z}^*)) = \operatorname{rank}(\mathbf{J}\mathbf{f}_{I_i}(\mathbf{z}^*)) + \operatorname{rank}(\mathbf{J}\mathbf{f}_{O_{k,1}}(\mathbf{z}^*)). \tag{26}$$

Leveraging Lemma 3 yields $\operatorname{rank}(\mathbf{J}\mathbf{f}_{I_i}(\mathbf{z}^*)) = M$. Inserting this in the previous equation yields

$$\operatorname{rank}(\mathbf{J}\mathbf{f}_A(\mathbf{z}^*)) = M + \operatorname{rank}(\mathbf{J}\mathbf{f}_{O_{k,1}}(\mathbf{z}^*)), \tag{27}$$

which according to Eq. (25) must be $\leq M$ i.e.

$$M \geq \operatorname{rank}(\mathbf{J}\mathbf{f}_A(\mathbf{z}^*)) = M + \operatorname{rank}(\mathbf{J}\mathbf{f}_{O_{k,1}}(\mathbf{z}^*)). \tag{28}$$

Now, note that by the definition of $I_k(\mathbf{z}^*)$ in Eq. (3), $\forall i \in I_k(\mathbf{z}^*)$, $\mathbf{J}\mathbf{f}_i(\mathbf{z}^*) \neq \mathbf{0}$. Because $O_{k,1} \neq \varnothing$ and $O_{k,1} \subseteq I_k(\mathbf{z}^*)$, it follows that $\mathbf{J}\mathbf{f}_{O_{k,1}}(\mathbf{z}^*) \neq \mathbf{0}$. This implies $\operatorname{rank}(\mathbf{J}\mathbf{f}_{O_{k,1}}(\mathbf{z}^*)) > 0$. However, this contradicts Eq. (28) and, hence, also the initial assumption in Eq. (24). Therefore, we conclude that $\forall i \in P,\ O_{i,2} \neq \varnothing$.

Taken together, we have shown that $\forall i \in P$, the sets $O_{i,1}, O_{i,2}$ are nonempty and form a partition of $I_i(\mathbf{z}^*)$.

Step 2.2 Next, we first note that Lemma 3 implies that the rank of the mechanism $\mathbf{J}\mathbf{f}_{I_i}(\mathbf{z}^*)$ is equal to M. Moreover, by assumption, $\mathbf{J}\mathbf{f}_{I_i}(\mathbf{z}^*)$ is irreducible. Because $O_{i,1}$ and $O_{i,2}$ form a partition of $I_i(\mathbf{z}^*)$, irreducibility then implies:

$$\forall i \in P : \operatorname{rank}(\mathbf{J}\mathbf{f}_{O_{i,1}}(\mathbf{z}^*)) + \operatorname{rank}(\mathbf{J}\mathbf{f}_{O_{i,2}}(\mathbf{z}^*)) > M. \tag{29}$$

Due to the equality of Jacobian ranks between $\mathbf{f}$ and $\hat{\mathbf{f}}$ stated in Lemma 4, Eq. (29) implies

$$\forall i \in P : \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{O_{i,1}}(\hat{\mathbf{z}}^*)) + \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{O_{i,2}}(\hat{\mathbf{z}}^*)) > M. \tag{30}$$

By the definition of $O_{i,1}, O_{i,2}$ in Eq. (22), $\forall i \in P : O_{i,1} \subseteq \hat{I}_1(\hat{\mathbf{z}}^*),\ O_{i,2} \cap \hat{I}_1(\hat{\mathbf{z}}^*) = \varnothing$. It thus follows from Lemma 1 that the sub-mechanisms defined by $O_{i,1}$ and $O_{i,2}$ are independent under $\hat{\mathbf{f}}$ in the sense of Defn. 4. Because $O_{i,1}$ and $O_{i,2}$ form a partition of $I_i(\mathbf{z}^*)$, this independence, when coupled with Eq. (30), implies:

$$\forall i \in P : \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{I_i}(\hat{\mathbf{z}}^*)) = \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{O_{i,1}}(\hat{\mathbf{z}}^*)) + \operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{O_{i,2}}(\hat{\mathbf{z}}^*)) > M. \tag{31}$$

We know from Lemma 3 that the mechanism defined by $I_i(\mathbf{z}^*)$ has rank M under $\mathbf{f}$. The equality of Jacobian ranks between $\mathbf{f}$ and $\hat{\mathbf{f}}$ stated in Lemma 4 then implies:

$$\operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{I_i}(\hat{\mathbf{z}}^*)) = \operatorname{rank}(\mathbf{J}\mathbf{f}_{I_i}(\mathbf{z}^*)) = M, \tag{32}$$

which contradicts Eq. (31), and, hence the initial assumption of this proof by contradiction in Eq. (21) cannot be correct and Eq. (20) must hold true.

We have now shown that $\forall \mathbf{z} \in \mathcal{Z}, j \in [K]$, there exists at least one and at most one $i \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing$ implying there exists exactly one, thus completing the proof. □

We now provide a corollary to Prop. 2 stating that the result also holds when the roles of $\hat{I}_j(\hat{\mathbf{z}}), I_i(\mathbf{z})$ are reversed.

Corollary 1. $\forall \mathbf{z} \in \mathcal{Z}, i \in [K]$, *there exists exactly one* $j \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing$.

Proof. We will first prove that there exists at least one $j \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing$. Assume, for a contradiction that:

$$\exists \mathbf{z}^* \in \mathcal{Z}, i \in [K], \nexists j \in [K] : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) \neq \varnothing. \tag{33}$$

This contradiction can be shown not to hold by exactly repeating the procedure in **Step 1** of Prop. 2.

We thus only need to prove that there exists at most one $j \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing$. Let $C := \{\, P \subseteq [K] : |P| > 1 \,\}$. Suppose for a contradiction that:

$$\exists \mathbf{z}^* \in \mathcal{Z}, i \in [K], P \in C : \qquad j \in P \implies \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) \neq \varnothing. \tag{34}$$

Let $A := [K] \setminus P$. We know by Prop. 2 that $\forall j \in A$, there exists exactly one $i \in [K] : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) \neq \varnothing$. This implies that at least $|[K]| - |A| = |P|$ ground-truth latent slots generate pixels which do not overlap with the pixels generated by any inferred latent slots in A. In other words, there exists a set $B \subset [K]$ with cardinality $\geq |P| > 1$ s.t.

$$\forall i \in B, \forall j \in A : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) = \varnothing \tag{35}$$

Now consider the set P. We know by Eq. (34), that for all $j \in P : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) \neq \varnothing$. By Prop. 2, we know that for all $j \in P$, $\hat{I}_j(\hat{\mathbf{z}}^*)$ can intersect only with $I_i(\mathbf{z}^*)$. Given that $|B| > 1$, this then implies

$$\exists i \in B : \forall j \in P : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) = \varnothing \tag{36}$$

Now, by construction, $[K] = A \cup P$. Thus, Eq. (35) and Eq. (36) together imply:

$$\exists i \in B \subset [K] : \forall j \in [K] : \hat{I}_j(\hat{\mathbf{z}}^*) \cap I_i(\mathbf{z}^*) = \varnothing \tag{37}$$

We have already shown in the first part of this corollary, however, that Eq. (37) cannot be true by repeating the procedure in **Step 1** of Prop. 2. Thus, our assumed contradiction in Eq. (34) cannot be true.

We have now shown that $\forall \mathbf{z} \in \mathcal{Z}, i \in [K]$, there exists at least one and at most one $j \in [K] : \hat{I}_j(\hat{\mathbf{z}}) \cap I_i(\mathbf{z}) \neq \varnothing$ implying there exists exactly one, thus completing the proof. □

We now build upon Prop. 2 and Cor. 1, to show that all inferred latent slots depend on at most one ground-truth slot.

Proposition 3. *Let $\mathcal{Z}$ be a latent space and $\mathcal{X}$ an observation space. Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a diffeomorphism that is compositional (Defn. 1) with irreducible mechanisms (Defn. 5). Let $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ be a diffeomorphism with inverse $\hat{\mathbf{f}} : \mathcal{Z} \to \mathcal{X}$ that is compositional (Defn. 1). Let $\hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z})), \forall \mathbf{z} \in \mathcal{Z}$. Then, $\forall \mathbf{z} \in \mathcal{Z}, i \in [K]$, there exists at most one $j \in [K] : \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_i}(\mathbf{z}) \neq \mathbf{0}$.*

Proof. Our goal is to show that at most one $\hat{\mathbf{z}}_j$ is a function of a given $\mathbf{z}_i$. More precisely, let $C := \{ P \subseteq [K] : |P| > 1 \}$. We aim to show that:

$$\forall \mathbf{z} \in \mathcal{Z}, i \in [K], \nexists P \in C : \qquad j \in P \implies \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_i}(\mathbf{z}) \neq \mathbf{0}. \tag{38}$$

Suppose for a contradiction to Eq. (38) that:

$$\exists \mathbf{z}^* \in \mathcal{Z}, i \in [K], P \in C : \qquad j \in P \implies \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_i}(\mathbf{z}^*) \neq \mathbf{0}. \tag{39}$$

Let $\mathbf{z}^*$ denote the value for which Eq. (39) holds and without loss of generality let $i = 1$.

We first introduce the function

$$\mathbf{h} := \hat{\mathbf{g}} \circ \mathbf{f} : \mathcal{Z} \to \mathcal{Z} \text{ s.t. } \hat{\mathbf{z}} := \hat{\mathbf{g}}(\mathbf{f}(\mathbf{z})) = \mathbf{h}(\mathbf{z}).$$

Note that $\mathbf{f} = \hat{\mathbf{f}} \circ \hat{\mathbf{g}} \circ \mathbf{f} = \hat{\mathbf{f}} \circ \mathbf{h}$. Thus, $\forall S \subseteq [N], \ \mathbf{f}_S = \hat{\mathbf{f}}_S \circ \mathbf{h}$. Therefore,

$$\forall \mathbf{z} \in \mathcal{Z}, j \in [K] : \frac{\partial \mathbf{f}_{\hat{I}_j}}{\partial \mathbf{z}_1}(\mathbf{z}) = \frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}}(\hat{\mathbf{z}}) \frac{\partial \hat{\mathbf{z}}}{\partial \mathbf{z}_1}(\mathbf{z}) \tag{40}$$

Due to the compositionality of $\hat{\mathbf{f}}$, $\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_k}(\hat{\mathbf{z}}) = \mathbf{0}, \forall k \neq j \in [K]$. This implies that these columns can be ignored when taking the product in Eq. (40), s.t.

$$\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}}(\hat{\mathbf{z}}) \frac{\partial \hat{\mathbf{z}}}{\partial \mathbf{z}_1}(\mathbf{z}) = \frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}) \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_1}(\mathbf{z}). \tag{41}$$

Now by Cor. 1, there exists exactly one $j \in P \subseteq [K]$ s.t. $\hat{I}_j(\hat{\mathbf{z}}^*) \cap I_1(\mathbf{z}^*) \neq \varnothing$. By the definition of $I_i(\mathbf{z})$ in Eq. (3), this implies that there exists exactly one $j \in P$ s.t. $\frac{\partial \mathbf{f}_{\hat{I}_j}}{\partial \mathbf{z}_1}(\mathbf{z}^*) \neq \mathbf{0}$. $|P| > 1$, thus there exists a $j \in P$ s.t.

$$\frac{\partial \mathbf{f}_{\hat{I}_j}}{\partial \mathbf{z}_1}(\mathbf{z}^*) = \frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}^*) \frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_1}(\mathbf{z}^*) = \mathbf{0} \tag{42}$$

where we leveraged Eq. (40), Eq. (41) to get the first equality above. Now, we know by Lemma 3, that $\mathbf{J}\hat{\mathbf{f}}_{\hat{I}_j}(\hat{\mathbf{z}}^*)$ is full column-rank. By compositionality of $\hat{\mathbf{f}}$, we also know that $\operatorname{rank}(\mathbf{J}\hat{\mathbf{f}}_{\hat{I}_j}(\hat{\mathbf{z}}^*)) = \operatorname{rank}(\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}^*))$ as these are the only non-zero columns in $\mathbf{J}\hat{\mathbf{f}}_{\hat{I}_j}(\hat{\mathbf{z}}^*)$. Thus, $\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}^*)$ is also full column-rank. Now, Eq. (42) implies that all columns of $\frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_1}(\mathbf{z}^*)$ must be in $\operatorname{null}(\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}^*))$. Because, $\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}^*)$ is full-column rank, $\operatorname{null}(\frac{\partial \hat{\mathbf{f}}_{\hat{I}_j}}{\partial \hat{\mathbf{z}}_j}(\hat{\mathbf{z}}^*)) = \mathbf{0}$. However, by Eq. (39) at least one column of $\frac{\partial \hat{\mathbf{z}}_j}{\partial \mathbf{z}_1}(\mathbf{z}^*)$ is non-zero. Thus, we obtain a contradiction and conclude that Eq. (38) must hold. □

Building on top of the previous propositions, we now prove our main identifiability result:

Theorem 1. *Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a diffeomorphism that is compositional (Defn. 1) with irreducible mechanisms (Defn. 5). If an inference model $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ is* (i) *a diffeomorphism with* (ii) *compositional inverse $\hat{\mathbf{f}} = \hat{\mathbf{g}}^{-1}$, then $\hat{\mathbf{g}}$ slot-identifies $\mathbf{z} = \mathbf{g}(\mathbf{x})$ in the sense of Defn. 6.*

Proof. According to Prop. 1 every inferred latent slot $\hat{\mathbf{z}}_j$ depends on *at least* one ground-truth latent slot $\mathbf{z}_i$. At the same time, Prop. 3 states that every inferred latent slot depends on *at most* one ground-truth slot. Hence, every inferred latent slot depends on *exactly* one ground-truth slot.

This implies that the Jacobian $\mathbf{J}\mathbf{h}(\mathbf{z})$ of $\mathbf{h} = \hat{\mathbf{g}} \circ \mathbf{f} : \mathcal{Z} \to \mathcal{Z}$ must be block diagonal up to permutation everywhere:

$$\forall \mathbf{z} \in \mathcal{Z} : \qquad \mathbf{J}\mathbf{h}(\mathbf{z}) = \mathbf{P}(\mathbf{z})\mathbf{B}(\mathbf{z}) \tag{43}$$

where $\mathbf{P}(\mathbf{z})$ is a permutation matrix and $\mathbf{B}(\mathbf{z})$ a block-diagonal matrix.

Next, note that

$$\det(\mathbf{J}\mathbf{h}(\mathbf{z})) = \det(\mathbf{P}(\mathbf{z}))\det(\mathbf{B}(\mathbf{z})) = \det(\mathbf{B}(\mathbf{z})) \neq 0 \tag{44}$$

since $\mathbf{h}$ is diffeomorphic. Hence, $\mathbf{B}(\mathbf{z})$ is invertible with continuous inverse. We conclude that

$$\mathbf{P}(\mathbf{z}) = \mathbf{J}\mathbf{h}(\mathbf{z})\mathbf{B}^{-1}(\mathbf{z}) \tag{45}$$

is continuous. At the same time, $\mathbf{P}(\mathbf{z})$ can only attain a finite set of values since it is a permutation. Hence, $\mathbf{P}(\mathbf{z})$ must be constant in $\mathbf{z}$, that is, the same global permutation is used everywhere.[5]

Thus, for any $j \in K$, there exists a *unique* $i \in K$ such that the function $\mathbf{h}_j = \hat{\mathbf{g}}_j \circ \mathbf{f} : \mathcal{Z} \to \mathcal{Z}_j$ is, in fact, constant in all slots except $\mathcal{Z}_i$, i.e., it can be written as a mapping $\mathbf{h}_j : \mathcal{Z}_i \to \mathcal{Z}_j$.

Finally, all such $\mathbf{h}_j$ are diffeomorphic, since $\mathbf{h}$ is a diffeomorphism.

This concludes the proof that assumptions *(i)* and *(ii)* imply $\hat{\mathbf{g}}$ slot-identifies $\mathbf{z}$. □

We now show that the compositional contrast C_{comp} introduced in Eq. (6) indicates whether a map is compositional:

Lemma 5. *Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a differentiable function. $\mathbf{f}$ is compositional in the sense of Defn. 1 if and only if for all $\mathbf{z} \in \mathcal{Z}$:*

$$C_{\text{comp}}(\mathbf{f}, \mathbf{z}) = 0\,.$$

Proof. ($\Rightarrow$) We begin by analyzing $C_{\text{comp}}(\mathbf{f}, \mathbf{z})$:

$$\sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{j=k+1}^{K}\left\|\frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z})\right\|_2\left\|\frac{\partial f_n}{\partial \mathbf{z}_j}(\mathbf{z})\right\|_2 \tag{46}$$

Since all summands are non-negative, the sum can only equal zero if every summand is zero $\forall \mathbf{z} \in \mathcal{Z}$. Since $j \neq k$ in the summand, this means:

$$\forall \mathbf{z} \in \mathcal{Z}, \forall n \in [N], k \neq j \in [K] : \left\|\frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z})\right\|_2\left\|\frac{\partial f_n}{\partial \mathbf{z}_j}(\mathbf{z})\right\|_2 = 0 \tag{47}$$

[5] Suppose for a contradiction that $\mathbf{P}(\mathbf{z})$ attains distinct values at some $\mathbf{z}^A \neq \mathbf{z}^B$ in $\mathcal{Z}$. Since $\mathcal{Z}$ is convex, the line connecting $\mathbf{z}^A$ and $\mathbf{z}^B$ is also in $\mathcal{Z}$ and $\mathbf{P}$ must change value somewhere along this line, leading to a discontinuity and thus a contradiction.

This relation can only be satisfied if one (or both) of the partial derivatives in the summand have a norm of zero, i.e. if they are zero. More precisely,

$$\forall \mathbf{z} \in \mathcal{Z}, \forall n \in [N], k \neq j \in [K] : \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) = \mathbf{0} \vee \frac{\partial f_n}{\partial \mathbf{z}_j}(\mathbf{z}) = \mathbf{0}. \tag{48}$$

According to Defn. 1 a map $\mathbf{f}$ is compositional if

$$\forall \mathbf{z} \in \mathcal{Z} : \qquad k \neq j \implies I_k(\mathbf{z}) \cap I_j(\mathbf{z}) = \varnothing. \tag{49}$$

By the definition of $I_i(\mathbf{z})$ in Eq. (3), we can restate Eq. (49) as:

$$\forall \mathbf{z} \in \mathcal{Z}, k \neq j, \nexists n \in [N] : \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) \neq \mathbf{0} \wedge \frac{\partial f_n}{\partial \mathbf{z}_j}(\mathbf{z}) \neq \mathbf{0} \tag{50}$$

which implies:

$$\forall \mathbf{z} \in \mathcal{Z}, n \in [N], k \neq j : \frac{\partial f_n}{\partial \mathbf{z}_k}(\mathbf{z}) = \mathbf{0} \vee \frac{\partial f_n}{\partial \mathbf{z}_j}(\mathbf{z}) = \mathbf{0} \tag{51}$$

which is equivalent to Eq. (48). Hence, $\forall \mathbf{z} \in \mathcal{Z} : C_{\text{comp}}(\mathbf{f}, \mathbf{z}) = 0$ implies that $\mathbf{f}$ is compositional.

($\Leftarrow$) We now prove the reverse direction i.e. that if $\mathbf{f}$ is compositional, then $\forall \mathbf{z} \in \mathcal{Z} : C_{\text{comp}}(\mathbf{f}, \mathbf{z}) = 0$. Note that the form of compositionality given in Eq. (50) implies that $\forall \mathbf{z} \in \mathcal{Z}$, at least one term in the summand of $C_{\text{comp}}(\mathbf{f}, \mathbf{z})$ in Eq. (51) will be zero. Thus, each summand is equal to zero. This then implies that $\forall \mathbf{z} \in \mathcal{Z} : C_{\text{comp}}(\mathbf{f}, \mathbf{z}) = 0$, completing the proof. □

Finally, by leveraging Lemma 5, we can obtain Thm. 1 in a less abstract form.

Theorem 2. *Let $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ be a diffeomorphism that is compositional (Defn. 1) with irreducible mechanisms (Defn. 5). If an encoder $\hat{\mathbf{g}} : \mathcal{X} \to \mathcal{Z}$ and decoder $\hat{\mathbf{f}} : \mathcal{Z} \to \mathcal{X}$ are both differentiable and solve the following functional equation*

$$\mathbb{E}_{\mathbf{x} \sim p_{\mathbf{x}}} \left[\left\| \hat{\mathbf{f}}(\hat{\mathbf{g}}(\mathbf{x})) - \mathbf{x} \right\|_2^2 + \lambda C_{\text{comp}} \left(\hat{\mathbf{f}}, \hat{\mathbf{g}}(\mathbf{x}) \right) \right] = 0, \tag{7}$$

for $\lambda > 0$, then $\hat{\mathbf{g}}$ slot-identifies $\mathbf{z}$ in the sense of Defn. 6.

Proof. As both summands of the functional are non-negative, solving the functional equation means solving for each of the summands to be equal to zero. Thus, we can analyze both of them separately. Solving the first sub-functional equation, i.e.,

$$\mathbb{E}_{\mathbf{x} \sim p_{\mathbf{x}}} \left[\left\| \hat{\mathbf{f}}(\hat{\mathbf{g}}(\mathbf{x})) - \mathbf{x} \right\|_2^2 \right] = 0,$$

implies that $\hat{\mathbf{f}}$ is an inverse of $\hat{\mathbf{g}}$ for every $\mathbf{x} \sim p_{\mathbf{x}}$. Because $p_{\mathbf{z}}$ is assumed to have full support over $\mathcal{Z}$, and $p_{\mathbf{x}}$ is defined by applying a diffeomorphism $\mathbf{f} : \mathcal{Z} \to \mathcal{X}$ on $p_{\mathbf{z}}$, this implies that $p_{\mathbf{x}}$ has full support over $\mathcal{X}$. This means that $\hat{\mathbf{f}}$ is an inverse of $\hat{\mathbf{g}}$ over the entire space $\mathcal{X}$ i.e. $\hat{\mathbf{f}} = \hat{\mathbf{g}}^{-1}$. Since per assumption $\hat{\mathbf{g}}$ and $\hat{\mathbf{f}}$ are differentiable it follows that $\hat{\mathbf{g}}$ is a diffeomorphism.

Moreover, per Lemma 5, solving the second sub-functional equation for $\lambda > 0$, i.e.,

$$\mathbb{E}_{\mathbf{x} \sim p_{\mathbf{x}}} \left[\lambda C_{\text{comp}}(\hat{\mathbf{f}}, \hat{\mathbf{g}}(\mathbf{x})) \right] = 0,$$

means that $\hat{\mathbf{f}}$ is compositional as $p_{\mathbf{x}}$ has full support over $\mathcal{X}$ and $\hat{\mathbf{g}}$ is a diffeomorphism between $\mathcal{X}$ and $\mathcal{Z}$. From Thm. 1 it now follows that $\hat{\mathbf{g}}$ slot-identifies $\mathbf{z}$, concluding the proof. □

B Experimental Details

B.1 Synthetic Data § 5.1

Enforcing Irreducibility We choose slot-output dimension, which we will denote $\dim(\mathbf{x}_s)$, to be greater than slot-dimension M as this is required for irreducibility (Defn. 5). To see this, assume the number of rows in each mechanism (Defn. 2), equal in our case to $\dim(\mathbf{x}_s)$, were equal to M. Because mechanisms have $\text{rank} = M$ (Lemma 3) and we have

M rows, this implies that no row is in the span of any others. Hence, the mechanism would be reducible. Beyond enforcing that the slot-output dimension, equal to 20 in this case, is greater than $M = 3$, we do not do anything further to ensure that our ground-truth generator is irreducible. This is because it is extremely unlikely that the generator, as we have constructed it, could be reducible. Specifically, if the generator were reducible, then as $\dim(\mathbf{x}_s)$ becomes larger than M, each new row in the Jacobian would need to lie in the span of some subset of the previous rows. As $\dim(\mathbf{x}_s)$ continues to increase relative to M, however, this becomes increasingly unlikely since the rows in the weight matrices of our MLP generator are randomly sampled i.e. entries are sampled uniformly from $[-10, 10]$.

Inference Model Training and Evaluation For our inference model, we use a 3 layer MLP with 80 hidden units in each layer and LeakyReLU activation functions. We train on 75,000 samples and use 6,000 and 5,000 for validation and test sets, respectively. We train for 100 epochs with the Adam optimizer (Kingma & Ba, 2015) on batches of 64 with an initial learning rate of 10^{-3}, which we decay by factor of 10 after 50 epochs. We use the validation set to find the optimal permutation for the Hungarian matching and then evaluate the SIS on the test set after applying this permutation to the slots. We compute the SIS for models every 4 epochs during training, all of which are plotted in Fig. 4. We trained all models using PyTorch (Paszke et al., 2019).

B.2 Existing Object-Centric Models § 5.2

Data Generation We generate image data using the Spriteworld renderer (Watters et al., 2019). Images consist of 2 to 4 objects, each described by 4 continuous (size, color, x/y position) and 1 discrete (shape) independent latent factors. We sample all factors uniformly where size is sampled from $[.1, .15]$ and x/y position both from $[.1, .8]$. We represent color using HSV and sample hue from $[0, 1]$ while fixing saturation and value to 3 and 1, respectively. The dataset consists of 100,000 images, 90,000 of which are used for training and 10,000 for evaluation.

Inference Model Training and Evaluation We use the same Slot Attention model proposed by Locatello et al. (2020), with the changes being that we use 16 convolutional filters in the decoder opposed to 32 and do not use a learning rate warm-up. For MONet, we follow the setup used by Dittadi et al. (2022) on Multi-dSprites (Kabra et al., 2019). For our additive autoencoder, we use the convolutional encoder/decoder architecture proposed by Burgess et al. (2018). The model decodes each slot separately to get slot-wise reconstructions and mask, applies the normalized mask to each slot-wise reconstruction, and then adds the results together to get the final reconstructed image. For all models, we use 4 slots with a slot-dimension of 16. We train all models for $500,000$ iterations (356 epochs) on batches of 64 with between 5 to 12 random seeds for each model. We train using the Adam optimizer (Kingma & Ba, 2015) with an initial learning rate of 10^{-4}, which we decay throughout training for all models using the same decay scheduler as Locatello et al. (2020). We trained all models using PyTorch (Paszke et al., 2019).

B.3 Compositional Contrast Normalized Variants

When computing C_{comp} in § 5.1 and § 5.2, we use a few different normalized variants of the contrast to overcome potential issues with the definition given in Defn. 7. Firstly, as the number of latent slots K increases, the contrast in Defn. 7 will scale by a factor $K^2 - K$. Thus, when comparing models across different numbers of slots in § 5.1, we divide the contrast by this factor to ensure that comparisons remain meaningful across different values of K. Another issue with the contrast in Defn. 7, is that it is not scale invariant. Specifically, naively minimizing the norm of the gradients for each pixel across slots will also minimize the contrast, despite all slots having similar gradient norms for a given pixel. This scale invariance did not cause issues when optimizing C_{comp} directly in § 5.1. However, when evaluating the C_{comp} of object-centric models in § 5.2, we account for this invariance. Specifically, we divide the gradient norms for each pixel with respect to each slot by the mean gradient norm for this pixel across slots. This gradient normalization creates an additional problem, however: Pixels with a relatively small gradient norm, such as black background pixels, will be weighted equally to pixels with a larger gradient norm such as pixels corresponding to an object. To account for this, we weight each pixel's contribution to the contrast by the pixel's mean gradient across slots.

B.4 Slot Identifiability Score

We are interested in a metric measuring how much information about the ground-truth latent slots is contained in the inferred latent slots without mixing information about different ground-truth slots into the same inferred slot. Let $S_1, S_2 \in [0, 1]$ denote scores that quantify how much information about each ground-truth slot can be extracted from the most and second-most predictive inferred slot, respectively. The aforementioned metric can be computed by just subtracting the two scores,

i.e.

$$S = S_1 - S_2. \tag{52}$$

Following previous work, we use the R^2 coefficient of determination as a score for continuous factors of variation (which we restrict to be strictly non-negative) and the accuracy for categorical factors (Dittadi et al., 2022). We compute one S value for each type and take the weighted mean which we then average across all slots to get the final slot identifiability score (SIS).

Computing SIS on Synthetic Data § 5.1 To compute the scores S_1 and S_2 defined in our experiments in § 5.1, we must fit two inference models between ground-truth and inferred slots: one between the best-matching slots and one between the second-best-matching slots. In § 5.1, we fit these models by first fitting a kernel ridge regression model between every pair of inferred and ground-truth slots and computing the R^2 scores for the predictions given by each model. We then use the Hungarian algorithm (Kuhn, 1955) to match each ground-truth slot to its most predictive inferred slot based on these R^2 scores, which gives us S_1. To get S_2, we take the highest R^2 score for each inferred slot with respect to the ground-truth slots that it was not already matched with. For our experiments in Fig. 5 with dependent latent slots, S_2 will inevitably be non-zero even if a model is perfectly identifiable. Thus, for these experiments, we only consider S_1 and refer to this metric as the Slot MCC (Mean Correlation Coefficient).

Computing SIS on Image Data § 5.2 When training models to compute S_1 and S_2 in our experiments on image data in § 5.2, one issue that arises is that the permutation between inferred latent slots and ground-truth slots is not necessarily a global permutation but can also be a local permutation. This is due to the ground-truth generator function being permutation invariant. To resolve this, we take a similar approach to work by Dittadi et al. (2022) and perform an online matching during training of inferred latent slots to ground-truth slots using the training loss. Specifically, we compute the loss for every pairing of the ground-truth and inferred slots and use the Hungarian algorithm to pick the permutation that yields the lowest aggregate loss. As every slot can contain both continuous and categorical variables, we compute the mean squared error for continuous factors and cross-entropy for categorical variables and sum them up to obtain the training loss. In our experiments, we notice that the cross-entropy tends to yield unstable matching results. Therefore, we use the minimum probability margin [6] to compute the categorical loss to solve the matching problem. Before fitting the readout models, we standardized both the ground-truth and inferred latents. We parameterized the readout models as 5-layer MLPs with LeakyReLU nonlinearity and a hidden dimensionality of 256, and trained them for up to 100 epochs using the Lion optimizer with a learning rate of 10^{-4}. To prevent the network from locking in too early on a suboptimal solution, we add a small amount of noise (10 % of the maximum matching loss value) to the losses before determining the optimal matching. Finally, we suggest performing cross-validation and early stopping to prevent overfitting.

For training the model to compute S_2, we proceed as for S_1 but ensure that the model is not using the same permutation used for computing S_1, i.e., it is trained on the second-best matching between ground-truth and inferred slots. Lastly, when computing S_2, we aim to avoid scenarios in which the model finds a spurious permutation yielding a non-zero S_2 despite the model being identifiable. To account for this, we compute S_2 on the ground-truth latent slots, denoted S_2^{gt}, using the same procedure for computing S_2, and use this score to adjust our previous scores. Specifically, by adjusting the value range accordingly, we obtain a score of

$$S = \frac{S_1 - S_2^{\text{gt}}}{1 - S_2^{\text{gt}}} - \frac{S_2 - S_2^{\text{gt}}}{1 - S_2^{\text{gt}}}, \tag{53}$$

To ensure that the subtracting term is not increasing the final score, we restrict it to be positive, yielding the final score:

$$S = \frac{S_1 - S_2^{\text{gt}}}{1 - S_2^{\text{gt}}} - \max\left(\frac{S_2 - S_2^{\text{gt}}}{1 - S_2^{\text{gt}}}, 0\right). \tag{54}$$

We may additionally be interested in considering the two terms on the RHS of Eq. (54) separately. Thus, we define them below as:

$$\hat{S}_1 = \frac{S_1 - S_2^{\text{gt}}}{1 - S_2^{\text{gt}}}, \quad \hat{S}_2 = \frac{S_2 - S_2^{\text{gt}}}{1 - S_2^{\text{gt}}}, \quad S = \hat{S}_1 - \max(\hat{S}_2, 0). \tag{55}$$

[6] i.e., $\max_i p_i - p_y$, where p denotes the predicted probability for different values of the categorical distribution and y the ground-truth value

C Additional Figures and Experiments

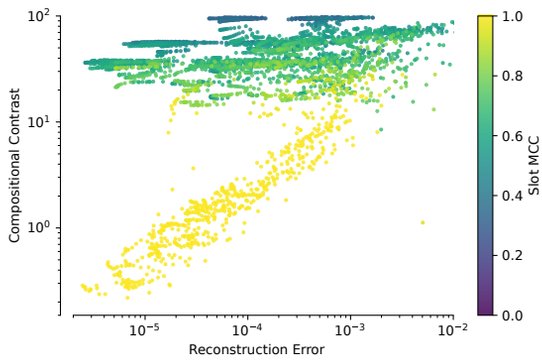


Figure 5. **Experimental validation of Thm. 2 for statistically dependent slots.** We trained models on synthetic data generated according to § 2 with 2, 3, 5 dependent latent slots (see § 5.1). The color coding indicates the level of identifiability achieved by the model, measured by the Slot Mean Correlation Coefficient (MCC), where higher values correspond to more identifiable models. As predicted by our theory, if a model sufficiently minimizes both reconstruction error and compositional contrast, then it identifies the ground-truth latent slots.

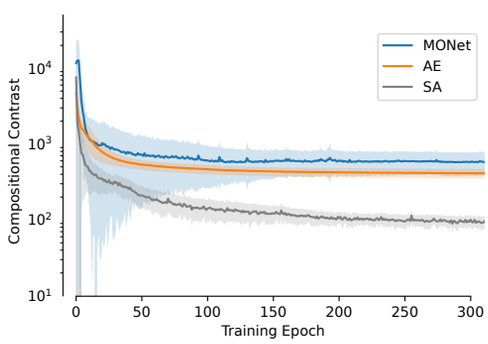


Figure 6. **Compositional Contrast (C_{comp}) throughout training.** Here, we plot the compositional contrast (C_{comp}) over the course of training for MONet, Slot Attention (SA) as well as an additive auto-encoder (AE), on image data. We can see that all models appear to be minimizing C_{comp} to some extent despite it not being explicitly optimized for in any of these models.

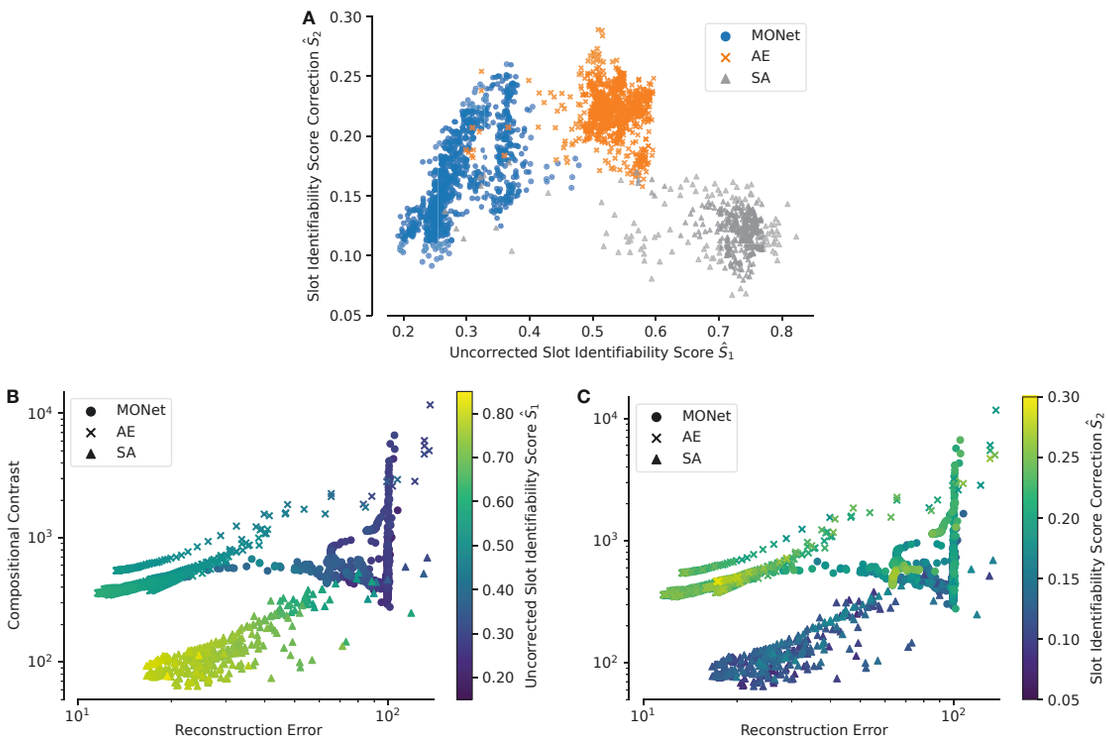


Figure 7. **Analysis of Information Leakage Between Slots from Models Trained in § 5.2. (A) Uncorrected Slot Identifiability Score ($\hat{S}_1$) vs. Correction ($\hat{S}_2$).** We train 3 existing object-centric architectures—MONet, Slot Attention (SA), and an additive auto-encoder (AE)—on image data and investigate whether inferred latent slots encode information from multiple objects when using an inferred latent dimension greater than the ground-truth. To test this, we look at the R^2 score for a model fit between each inferred slot and the second most predictive ground-truth slot for this slot. We refer to this score as the *slot identifiability score correction*, defined as $\hat{S}_2$ in Appx. B.4. We plot this score against the uncorrected slot identifiability score i.e. the most predictive ground-truth slot, defined as $\hat{S}_1$ in Appx. B.4. We can see that for all models, $\hat{S}_2$ is non-zero, even as $\hat{S}_1$ increases, suggesting that models are leveraging their additional latent capacity to encode information about multiple objects in the same latent slot. **(B) and (C) Influence of Reconstruction Error and Compositional Contrast on $\hat{S}_1$ and $\hat{S}_2$.** Here, we further visualize the slot identifiability score correction ($\hat{S}_1$) and the uncorrected score ($\hat{S}_2$) as a function of the reconstruction error and the compositional contrast in panels B and C, respectively. We can see in B that, similar to the SIS in Fig. 4, $\hat{S}_1$ tends to increase as reconstruction loss and compositional contrast decrease. We can additionally see in C that, while $\hat{S}_2$ decreases to some extent with C_{comp}, there is generally less of a correlation between $\hat{S}_2$ and these metrics. This suggests that the latent capacity must also be restricted to minimize $\hat{S}_2$.

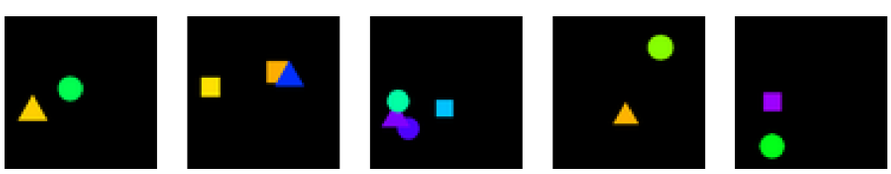


Figure 8. Samples from our multi-sprites dataset used in § 5.2. Objects are described by five latent factors: shape, color, size, and x/y position. Occlusions are present in the dataset, as shown in the samples above (see the second and third images from the left).