Advancing Multi-View Scene Interpretation: Leveraging Deep Learning for Optimized Input Image Analysis

Dissertation

der Mathematisch-Naturwissenschaftlichen Fakultät

der Eberhard Karls Universität Tübingen

zur Erlangung des Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

vorgelegt von

Arijit Mallick

aus Midnapur/Indien

Tübingen

2024

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

Tag der mündlichen Qualifikation:	11.07.2025
Dekan:	Prof. Dr. Thilo Stehle
1. Berichterstatter:	Prof. Dr. Hendrik P. A. Lensch
2. Berichterstatter:	Prof. Dr. Andreas Geiger

To my mother and my late father

Abstract

This research aims to advance multi-view scene interpretation by addressing key challenges in image processing, 3D reconstruction, and burst image denoising. We leverage deep learning techniques to enhance input image quality and develop innovative methodologies for these computer vision tasks. Furthermore, our approach focuses on overcoming limitations in existing classical and learning-based methods. We introduce a novel view interpolation technique that generates intermediate frames accurately without requiring additional geometric input. This method lays the foundation for our subsequent work on multi-view 3D reconstruction. To address the lack of ground truth depth information in 3D reconstruction, we propose a meta-learning and unsupervised approach to tackle the classic problem of multi-view stereo. We also tackle the issue of low-resolution depth maps by introducing a depth-enhancing transformer-CNN hybrid module. Furthermore, we delve into burst image denoising by proposing a model that leverages multiple image alignment and feature volume merging, achieving state-of-the-art performance. Finally, we explore burst image denoising, proposing a model that utilizes multiple image alignment and feature volume merging to achieve state-of-the-art performance. Our research contributes significantly to the field of computer vision and has potential applications in various domains.

Kurzfassung

Diese Forschung zielt darauf ab, die Interpretation von Multi-View-Szenen voranzutreiben, indem zentrale Herausforderungen in der Bildverarbeitung, 3D-Rekonstruktion und Burst Image Denoising angegangen werden. Wir nutzen Deep Learning-Techniken, um die Qualität der Eingabebilder zu verbessern und innovative Methoden für diese Computer Vision-Aufgaben zu entwickeln. Darüber hinaus konzentriert sich unser Ansatz darauf, die Einschränkungen bestehender klassischer und lernbasierter Methoden zu überwinden. Wir führen eine neuartige View Interpolation-Technik ein, die Zwischenbilder genau generiert, ohne zusätzliche geometrische Eingaben zu benötigen. Diese Methode bildet die Grundlage für unsere nachfolgende Arbeit zur Multi-View-3D-Rekonstruktion. Um den Mangel an Ground Truth-Tiefeninformationen in der 3D-Rekonstruktion zu adressieren, schlagen wir einen Meta-Learning- und unüberwachten Ansatz vor, um das klassische Problem des Multi-View-Stereo zu lösen. Wir gehen auch das Problem der niedrigauflösenden Tiefenkarten an, indem wir ein Depth-Enhancing Transformer-CNN Hybridmodul einführen. Schließlich befassen wir uns mit Burst Image Denoising, indem wir ein Modell vorschlagen, das mehrere Bildausrichtungen und Feature Volume Merging nutzt, um eine state-of-the-art Leistung zu erzielen. Unsere Forschung leistet einen bedeutenden Beitrag zum Bereich der Computer Vision und hat potenzielle Anwendungen in verschiedenen Domänen.

Publications

The scholarly contributions made in this thesis has been published in the following three conference proceedings. Each of the works highlighted in this thesis has been showcased during the poster sessions at these conferences.

- Mallick, Arijit and Stückler, Jörg and Lensch, Hendrik P. A. **Learning to Adapt Multi-View Stereo by Self-Supervision**. Proceedings of the British Machine Vision Conference (BMVC), 2020.
- Mallick, Arijit and Engelhardt, Andreas and Braun,Raphael and Lensch, Hendrik P. A. **Local Attention Guided Joint Depth Upsampling**. Proceedings of Vision, Modeling, and Visualization (VMV), 2022.
- Mallick, Arijit and Braun, Raphael and Lensch, Hendrik P. A.**CANDID: Correspondence AligNment for Deep-burst Image Denoising**. Proceedings of 20th Conference on Robots and Vision (CRV), 2023.

Contents

Chapter 1

Introduction

Visual perception modeling, a cornerstone of scene understanding, has witnessed substantial progress in recent years. Central to this endeavor is the acquisition and analysis of meaningful visual data. Robust image-processing models, often augmented by complementary sensor data such as camera position, play a pivotal role in extracting valuable information. While data quantity can certainly enhance computer vision tasks like multi-view stereo, burst-image denoising, and view interpolation, it is the strategic acquisition of high-quality data that truly underpins accurate scene understanding. Moreover, vision-guided scene modeling can be further refined by integrating data from multiple sensory modalities.Hence, in order to achieve superior scene reconstruction performance, it is essential to find an optimal setup that depends on the particular source of incoming sensory data. In order to understand these nuances, it is important that we initialize with a solid introduction to the exact computer vision tasks covered in this work. This will also serve as the motivation behind writing this thesis. Multiple view sensor image input has been utilized in various applications, such as multi-view stereo for 3D reconstruction (Schönberger *et al.*, 2016; Yao *et al.*, 2018a; Ji *et al.*, 2017a; Khot *et al.*, 2019; Mallick *et al.*, 2020a), multi-view classification (Kan *et al.*, 2016; Chen *et al.*, 2022; Han *et al.*, 2021), and image registration for denoising (Tico, 2008a; Mildenhall *et al.*, 2018; Mallick *et al.*, 2023a). All of these applications primarily showcase the superior scenario modelling capability than its single view counterpart's performance (e.g monodepth estimation in comparison to depth estimation from stereo (Smolyanskiy *et al.*, 2018), or guided depth upsampling using additional RGB image in comparison to conventional upsampling, etc.). We briefly introduce some of the important problem statements that we specifically tackled, which contributed to the completion of this thesis work. To summarize it, the overall goal of this report is to highlight the importance of multi-view models in understanding scene representation for all possible computer vision tasks. Once the reader has been properly introduced to the specific problem statements, we discuss each of the problems in detail, summarize our solutions, and subsequently suggest possible directions for future improvement in the following chapters. We will sequentially move from one problem to another, addressing the motivation behind a given task and simultaneously proposing solutions to the overall multiple view input based reconstruction problem on a broader perspective.

1.1 Multiple view tasks in computer vision

We will concentrate on a variety of traditional computer vision tasks that utilize multiple-view input. These include 3D reconstruction based on multiple-view stereo, view interpolation tasks performed between consecutive frames, burst-image registration and feature aggregation for the purpose of denoising, and guided depth super-resolution to enhance scene reconstruction analysis. Each of these tasks plays a crucial role in our research and will be thoroughly explored. Furthermore, we will wrap up our discussion with an ongoing research project. This project is centered around the development of a cost-effective iterative neural correspondence computation model which serves as the fundamental structure for a lightweight depth estimation module. Its design and functionality are critical to the overall success of our research. These problem statements are at the forefront of recent advances in machine learning algorithms.

1.1.1 High quality frame interpolation

Image frame interpolation is a fundamental computer vision problem that involves synthesizing of new frames between existing ones, such as in a video. In our case, rather than utilizing a stream of video with temporal properties, we obtain a camera setup with geometric or positional inconstancy. With the help of multiple incoming images from different camera positions, we propose a learning module which can directly reproduce the view frame from that missing camera position.This has wide usages in applications such as image stitching, panoramic image viewing, tourism photography, mobile photography, terrain mapping and robot navigation to name a few. Existing methods usually involve estimating an optical flow between subsequent frames, followed by warping to obtain an intermediate frame. Such methods suffer from parallax error and can cause noisy artifacts during final image reconstruction. Instead of estimating the homography matrix directly, we propose learning the stitching process to directly predict the missing pixels. The frame interpolation problem involves a warping stage to align the multiple source images based on estimated camera motion. This suggests that we can leverage multiple view image inputs to estimate depth images, which can improve our understanding of the surroundings and eventually solve complex vision problem such as multiple view stereo.

1.1.2 Multi-view stereo with semi-supervision

Multiple view image input along with camera geometry information has been widely used to reconstruct 3D surroundings, using both classical and learning approaches. This is widely known as the multiple-view stereo (MVS) problem. Classical approaches lag behind in reconstructing homogeneous surfaces, while learning methods require expensive ground truth data, such as depth maps. Additionally, learning methods are traditionally expensive, as most successful methods use volumetric cost computation by dis-

cretizing the depth space. We address some basic problems of the MVS task in the light of current learning approaches. These are insufficient ground truth training data (such as depth maps, precise camera geometry information etc.) and the adaptive capability of a trained model in case of environmental changes (such as indoor to outdoor). We propose a meta-learning powered training method by utilizing an established extension of the unsupervised multi-view stereo (MVS) approach. Our method estimates depths with combined photometric projection losses of the neighboring frames, guided externally by another supervised pre-trained network. This network ensures that the supervision signals are propagating the correct values. After certain iterations, our method performs a correction step to check the unsupervised learning. We evaluate our meta-learning powered pre-trained model by fine-tuning it on another well-established dataset.

1.1.3 Guided depth upsampling with transformers

Multi-view stereo learning approaches that use volumetric methods are computationally expensive and often reduce the output depth map resolution due to downsampling. This can lead to decreased accuracy in depth fusion to obtain point clouds. Additionally, photometric losses can further introduce inaccuracies. To address this, we propose a guided depth upsampling mechanism inspired by popular super-resolution approaches. One challenge of single image direct upsampling is that depth maps are single-channel raw information with limited contextual knowledge of surrounding objects or surfaces. This can make it difficult to upsample depth maps without introducing artifacts. We propose to solve this problem by leveraging additional information, such as the high-resolution RGB color map (hence called guided upsampling). Additionally, we employ the combined power of vision transformers and CNNs. Vision transformers compute local and global attention of the given scene, while CNNs learn upsampling weights from these contextual features.

1.1.4 Burst image denoising

Another aspect of estimating 3D motion from multiple view inputs is its use in realigning and combining burst images from digital and mobile phone photography to obtain high-resolution denoised images. Burst imaging or burst image photography involves capturing multiple snapshots of a scene in quick succession. This can introduce motion blur and jittery noise, especially in handheld devices. Therefore, there is a need for a fast and accurate method to re-align and denoise burst images. We propose a method that uses multiple view learning to estimate the re-alignment factor and compensate for noise in burst images by efficiently aggregating corresponding incoming image feature volume. This is achieved by generating pseudo-bursts from a pre-trained single image denoising model to provide additional information to the network.

1.1.5 Further directions on correspondence localisation

Multiple view learning has the potential to revolutionize our understanding of the world around us. Additionally, our ongoing work proposes a lightweight multiple view correspondence localization network that extracts features from multiple images to be used in heavyweight reconstruction tasks (disparity, depth estimation). Our preliminary results are promising and suggest a new approach to multiple view understanding. We delve into a more comprehensive overview of the details in the following sections.

Chapter 2

Literature review

The literature review covers the research background of the fundamental conceptual development of computer vision in the light of machine learning, and gradually progresses to seminal works on their applications that made this thesis possible. We begin our discussion with convolutional neural networks (CNNs) and their role in advancing classical computer vision problems. Next, we review view interpolation with CNNs and gradually progress to the more complex task of multi-view stereo and how comprehensive CNN models with meta-learning have been used to tackle it. Next, we provide a background research on the problem of depth upsampling. This is due to the fact that contemporary MVS models produce low resolution depths which needs enhancement for better reconstruction. For this task, we have used a combined CNN-Transformer model. Finally, we discuss the classical multi-image input problem of burst-image denoising and image enhancement, and review how classical methods and CNNs have performed over time. To establish a foundation for our discussion of current learning-based image processing models, we commence with a review of their building blocks: convolutional neural networks (CNNs).

2.1 Convolutional Neural Networks (CNNs)

The primary component of any computer vision problem is an image or a multiple sources of images with some peripheral camera data, if available. In order to have a robust understanding of the given image, one needs to accurately analyze and interpret the corresponding features in the context of the given task. This stage is known as a feature extraction, which looks into the overall local as well as global contextual meanings and patterns within the image. Prominent among them is SIFT (Lowe, 1999), which estimates points of interest called keypoints in an image. This is followed by keypoint localization, orientation and eventually description. The phenomenal success of this algorithm in object detection (Sirmacek and Unsalan, 2009), reconstruction (Fabbri *et al.*, 2012), action recognition (Niebles *et al.*, 2006) paved the way for tackling higher level computer vision tasks in a new light. This was followed by much faster and robust SURF, (Bay *et al.*, 2008) which used multi-resolution pyramid technique to ensure that the points of interests in the image are scale invariant. This was followed

by GLOH (Mikolajczyk and Schmid, 2005)descriptor which considered more spatial regions for computing histograms and HOG (Dalal and Triggs, 2005) which took into account the number of occurrences of gradient orientation in localized portions of an image. Parallelly, the advent of neural networks (Amari, 1967; Fukushima, 1980; Linnainmaa, 1970) revolutionized the field of computer vision. The initial backpropagation algorithm applied to a CNN was implemented in 1988 (Zhang, 1988) which was a simplified version of Neocognitron with a primitive convolutional layer for image feature extraction for alphabet recognition (Zhang *et al.*, 1990). We et al. in this seminal work coined the term Shift invariant artificial neural network, which became the precursor to modern day CNNs. Furthermore, an advanced version of this model was used to solve higher level medical image processing such as medical image segmentation (Zhang, 1991) and breast cancer detection (Zhang, 1994). This would become a classic template in future learning based computer vision tasks.

On the other hand, (Waibel, 1987) introduced Time Delay Neural Networks, and it was among the first convolutional neural networks which would perform shift invariance. The model showed the weight sharing and backpropagating properties (Waibel *et al.*, 1989) alongside global optimization of the weights. TDNNs were the first of its kind to share weights in the temporal axis (LeCun and Bengio, 1995). These were primarily designed for time invariant signals such as speech. This was further developed into a two-dimensional version by Hampshire and Waibel (Hampshire and Waibel, 1990). These two-dimensional convolutional networks successfully performed phenom recognition, and this model was invariant to both time and frequency. This is considered a landmark achievement, as it paved the way for translational invariance in future computer vision tasks (Waibel *et al.*, 1989). Another important milestone was achieved in 1990 when (Yamaguchi *et al.*, 1990) presented max pooling. In this constant filtering operation, their kernel architecture calculated the maximum value of the particular region and propagated it. A powerful combination of TDNNs and maxpooling resulted in modelling a word recognition system. They implemented a complicated model where several TDNNs were combined in order to map a larger group of syllables. The resultant TDNN responses of the large syllable inputs were combined with the help of maxpooling layers and subsequently propagated to the network to perform word recognition.

Furthermore, true image recognition performing model was developed by (LeCun *et al.*, 1989) which used backpropagation to handwritten digit dataset. Although being the first fully functional visual recognition model, it had a very high training time. This was followed by the successful LeNet-5 in 1998 (LeCun *et al.*, 1998) that classified digit. The robustness of this algorithm was reflected in its applicability where financial institutions used this for handwriting recognition on low-resolution images. Although proven to be a breakthrough, the model training would be extensively time and memory consuming when applied to higher resolution images.

The massive development in machine learning was achieved due to introduction of Graphical Processing Units (GPUs). In early 2000s, breakthroughs were made when it was shown that training in GPUs are at least 20 times faster than training the same neural

network models in traditional CPUs (Oh and Jung, 2004; Steinkraus *et al.*, 2005). Ciresan et al. showed that multi-layered standard neural network can perform with state-of-the-art precision on GPUs on previous MNIST handwritten digits (Cireşan *et al.*, 2010) related image processing tasks. This was further extended successfully to CNNs by completing multiple benchmark challenges (Benchmark, 2023; Schmidhuber, 2017, 2015; Ciresan *et al.*, 2011).This also included now famous CIFAR10 dataset (Cireşan *et al.*, 2012). A heavyweight CNN model by (Krizhevsky *et al.*, 2012) (AlexNet) won the ImageNet Large Scale Visual Recognition Challenge. This was further followed by 100 layers deep CNN model which won the ImageNet 2015 contest (He *et al.*, 2016). These developments eventually set a standard for CNN based image processing task handling. Taking a cue from these standard CNN implementations, we extend our application to one such task, called view interpolation for high-resolution image stitching.

2.2 Frame interpolation with CNNs

Frame or view interpolation is a technique used to improve the quality of an image by interpolating pixels to missing positions. This can be achieved with the help of classical nearest-neighbor or bicubic interpolation techniques. We focus on multiple-view image input consisting of different camera positions. Hence, the first step is to correct the camera jitter with the help of optical flow estimation or camera intrinsic estimation and use that information to obtain a high quality image.

Frame / view interpolation. Previous works on optical flow based method (D. Mahajan and Belhumeur, 2009) and the Eulerian phase-based approach has shown promising results (S. Meyer and SorkineHornung, 2015). Existing methods usually estimate dense motion between consecutive frames and corresponding interpolation is done based on the dense correspondence estimates (S. Baker and Szeliski, 2011), (M. Werlberger and Bischof, 2011), (Z. Yu and Chen, 2013). In addition to this, classical flow-based (Brox *et al.*, 2004) and multiscale phase/interpolation-based (Didyk *et al.*, 2013), (Meyer *et al.*, 2015)methods have already shown promising results and paved the way for implementation in future learning models.

CNNs for disparity estimation and view interpolation. Recent success of deep learning in almost all aspects of computer vision has been the prime inspiration for frame interpolation in our current work. Evaluating optical flow through deep learning-based methods (A. Dosovitskiy and Brox., 2015), (G¨uney and Geiger, 2016), (Teney and Hebert, 2016) is not an exception as well. Rendering unseen images from neighboring frames is also a conventional approach (A. Dosovitskiy and Brox, 2015), (T. D. Kulkarni and Tenenbaum, 2015), (J. Yang and Lee, 2015). Briefly speaking, extensive research has been done on view synthesis with the help of deep learning methods (J. Flynn and

Snavely, 2016) and is still an active research problem due to frame generation constraints which includes generation of high-quality artifact-free frames.

Deep Voxel Flow (DVF) (Liu *et al.*, 2017) has shown a simple self supervised method for video frame synthesis with the help of training only triplets of consecutive video frame data. This has been our inspiration for producing intermediate frames for our pipeline. It employs a simple autoencoder model which estimates voxel flow and provides a map which is then utilized for a trilinear interpolation from two given frames to produce the corresponding output. One of the major problems of both supervised and self-supervised learning approaches is that they typically do not generalize well to novel domains.

Meta-Learning for Self-supervision Recent developments in meta learning (Finn *et al.*, 2017) have demonstrated methods that efficiently adapt to novel tasks for supervised regression and reinforcement learning. The main idea behind model agnostic meta learning (MAML) is to train the model parameters in such a way that the network can better generalize to a new task through fine-tuning. MAML (Finn *et al.*, 2017) learns a feature representation which is suitable for a variety of tasks. It maximizes the sensitivity of the loss functions of the new task, and this inherent property facilitates generalization when continuing training in a different domain. Previous work on adaptive learning of stereo disparity estimation (Tonioni *et al.*, 2019) has utilized this meta-learning and have shown how feature representations can be learned for self-supervised learning and improved generalization on new datasets. From a multi-view stereo standpoint, the self-supervised losses tend to add more error as the number of neighboring frames increases because of unpredictable occlusion, out of bound pixel projections and variable camera baseline. Hence, it is very important to adapt to these variable conditions. We propose (Mallick *et al.*, 2020b) to learn adaptive feature representations for self-supervised multi-view stereo reconstruction through meta-learning. We develop extensions to a network architecture based on MVSNet (Yao *et al.*, 2018b) with which the model learns to mask uncertain predictions due to outliers such as occlusions. This assists the self-supervised fine-tuning on new domain data. One drawback of the volumetric based learning models are that they produce low resolution depth maps. Hence, our next work deals with guided depth map upsampling for high quality reconstruction applications.

2.3 Guided Joint Depth Upsampling

Classical methods The classical joint depth super resolution literature can be divided into filter-based methods and optimization-based approaches. In filter-based approaches, texture and edge features are extracted from the given guide RGB image to inform hand-crafted filters that try to estimate the weights for spatially-varying filter masks that are convolved with the lower resolution target image.

Joint bilateral upsampling (Kopf *et al.*, 2007) extends the single image bilateral filter (Tomasi and Manduchi, 1998) to steer the filter with a guide image. The bilateral weights are obtained by converting the local guide RGB image pixel values to bilateral weights, which are then applied cross-modal to the low resolution input. This concept was also extended to a faster foreground-background task understanding by an adaptive joint-bilateral filter. More precise upsampling is achieved by adaptive joint-bilateral filtering based on an edge-uncertainty map which combines the guide and target images (Camplani *et al.*, 2014). Guided filters (He *et al.*, 2013; Wu *et al.*, 2018) provide a similar idea of considering a filtered output factor from the guidance image. Aforesaid methods are based on filter kernels where strong local guide features are utilized to enhance a low resolution depth map. The upsampling task has also been addressed as a global energy minimization problem, such as the Markov random field based technique in (Diebel and Thrun, 2006). Non-local means filtering with extended regularization for additional edge weighting has further improved joint depth upsampling (Park *et al.*, 2011). These methods all employ a regularization term which guides the target towards a structurally similar texture of the high resolution guide image. The fast bilateral solver combines these simple filtering methods and approaches this problem as a domain-specific optimization algorithm (Barron and Poole, 2016). Additionally, in (Ham *et al.*, 2018) static-dynamic filter combinations have shown significant improvements on the joint upsampling task with the help of better structural prior extraction. Other global optimization approaches with similar techniques involve an adaptive autoregressive model (Yang *et al.*, 2014), a co-sparse analysis model (Kiechle *et al.*, 2013)and a sparse-coding algorithm with reconstruction constraint (Li *et al.*, 2012).

Learning-based methods Contrary to classical techniques which do not rely on supervision, data-driven learning approaches are becoming significantly popular because of their generalization capability on upsampling tasks. Early learning-based methods utilized a dictionary in order to express structural similarity within paired guide and target images. (Kwon *et al.*, 2015) utilize a sparse representation learning of dictionaries on the geometric correlation between high-quality mesh data, ground truth target and guide images. (Yang *et al.*, 2010) presented a sparse representation of the target map, and corresponding coefficients were used to predict a high resolution depth output. Lately, CNN-based techniques have shown significant improvements on the task of joint depth super resolution. Multiscale guidance networks with an encoder-decoder architecture (Hui *et al.*, 2016) got rid of depth boundary artifacts. Moreover, in (Li *et al.*, 2019), salient structures that are consistent in both guidance and target images are selectively leveraged. The deep primal-dual network (Riegler *et al.*, 2016) with iterative optimization has shown better noise removal along with good super-resolution results. Apart from these direct encoder-decoder approaches, The Deformable Kernel Network(DKN) (Kim *et al.*, 2020) learns a sparse and spatially-variant kernel which stretches a kernel non-linearly along the given pixel neighborhood. The method in turn extracts a resid-

ual offset from the combined image features. Apart from showing better performance, a faster extension was also shown with almost similar metrics by (Kim *et al.*, 2020). (Su *et al.*, 2019) learn to predict the filter weights of a spatially-varying kernel as a function of the local pixel features. A cross-task interaction module is introduced by (Sun *et al.*, 2021) to realize bilateral cross-modality knowledge transfer to solve uncertainty depth estimation guided super resolution. In (He *et al.*, 2021), high-frequency components decomposed from the RGB image subsequently guide the super resolution task. Apart from fully CNN-powered architectures, densely connected networks have also been proposed. (Lutio *et al.*, 2019) employ an MLP for pixel to pixel mapping of the guide information to the target. Similarly, (Tang *et al.*, 2021) utilize a deep implicit neural representation based technique. It is essentially an MLP which efficiently extracts latent codes from the input and appends it to the coordinates, eventually providing a depth correction residual. They achieve state-of-the-art results on noisy joint depth super resolution tasks. Orthogonal work by (de Lutio *et al.*, 2022) directly optimizes an explicit affinity graph to regularize the reconstruction. Overall, learning-based guided joint upsampling methods usually leverage monocular depth-like datasets (Silberman *et al.*, 2012; Lu *et al.*, 2014; Scharstein and Pal, 2007; Hirschmuller and Scharstein, 2007; He *et al.*, 2021). Learned joint bilateral upsampling has been integrated into the multi-view stereo task (Yu and Gao, 2020) where the bilateral weights are selected as a function of the given reference image for sparse-to-dense depth approximation, which significantly reduces the computation effort and provides a faster reconstruction. Contrary to the existing networks, we (Mallick *et al.*, 2022) contribute additional refinement to the low resolution guided depth map inputs with the help of transformer encoded attention weights. Additionally, our residual network contributes stronger edge aware features.

Transformers and local attention Transformers (Vaswani *et al.*, 2017) have become a widely used architecture, especially in Natural Language Processing(NLP) tasks (Devlin *et al.*, 2019; Brown *et al.*, 2020). Transformers primarily operate with the concept of self-attention, which explores the relation between all tokens in a sequence to capture contextual information. The base transformer encoder models have been successfully applied to low-level computer vision tasks such as classification (Dosovitskiy *et al.*, 2020). Recently, the Texture Transformer Network for image super resolution (Yang *et al.*, 2020) uses low resolution and reference RGB images as queries and keys in a transformer. They essentially transfer the high resolution texture to a low resolution image for a super resolution task. In the context of guided depth super resolution, self-attention has just started to be explored as a part of larger architectures (Xing *et al.*, 2021; Yang *et al.*, 2022; Ariav and Cohen, 2022). The Discrete Cosine Transform module in (Zhao *et al.*, 2022) employs an edge attention mechanism to highlight the contours, which provides useful information for guided upsampling. As basic self-attention has quadratic complexity in the number of tokens, Longformer (Beltagy *et al.*, 2020) introduces a number of different sampling approaches that improve the efficiency of attention evalua-

tions. In particular, the local sliding window attention mechanism scales linearly with the sequence length, allowing it to process even very large token sets. This idea can also be found in (Zhao *et al.*, 2022), where grouped convolutions are used to compute attention maps to weight edge information. In our scenario, we apply local sliding-window attention to a 2d patch around a pixel. Local attention provides a weighting of the spatially combined guide and target feature tensors, which helps in extracting rich contextual information. A separate merge network further enhances the correlation between them, leveraging both the power of CNNs and transformers for an efficient depth residual computation. Local image attention has been explored before (Hu *et al.*, 2018b,a; Cao *et al.*, 2019) in different forms such as spatial, channel-wise scenarios, or as a mixture of global and local attention mechanisms, primarily for image retrieval tasks (Song *et al.*, 2022). After addressing the depth resolution issue, we revisit the problem of image registration due to camera movement and propose an image enhancement model which not only compensates for these movements but performs denoising as well.

2.4 Burst image denoising

We discuss related works pertaining to burst image denoising and begin with single image denoising, followed by homography-based and optical flow-based alignment powered multiple image or video related tasks, and finally contemporary progress on deep-burst imaging.

Single image denoising Image denoising is a classical computer vision problem and is still one of the most sought after deep learning based low-level image processing research topics. Due to the increasing popularity of low-cost mobile photography, effective denoising and enhancement is well sought after. Most photography hardware companies take advantage of the recently developed lightweight neural network denoising models; exploiting the significant increase in mobile computation power. In the early days of CNNs, models such as (Gu *et al.*, 2019) improved performance compared to classical image denoising models based on Markov random fields, but they could not compete with BM3D (Dabov *et al.*, 2007a) which introduced a new denoising paradigm by combining 3D block matching and domain transform. They are later surpassed by a sparse denoising autoencoder models (Xie *et al.*, 2012; Aharon *et al.*, 2006). Simple multi-layer perceptron-based models (Schmidt and Roth, 2014) and later deeper residual networks (Mao *et al.*, 2016) have shown superior performance due to enhanced receptive fields. All these models have the advantage of being trainable end-to-end, exploiting simple to generate training data. For a multitude of image processing tasks, training can be accelerated using pre-trained models and transfer learning (Chen *et al.*, 2020). In this spirit, we incorporate the pre-trained self-guided network (SGN) (Gu *et al.*, 2019) to enrich the burst input with smooth priors. SGN extracts large-scale contextual information and gradually propagates it to the higher resolution subnetworks for feature self-guidance and

denoising at multiple scales. This efficient multiscale local features extraction property allows it to efficiently recover denoised images.

Deep-burst Denoising While single image denoising relies on learned image priors, deep-burst denoising assimilates features from multiple noisy frames to predict a better image. A similar idea is used in burst motion deblurring (Wieschollek *et al.*, 2017) where a sharp image is recurrently extracted from a burst of blurry ones. Similarly, recurrent neural networks have also been used for burst denoising. Multiple frame denoising usually involves some sort of alignment (Bhat *et al.*, 2021a) of the frames in the burst for superior feature assimilation. (Tico, 2008b) demonstrates a block matching approach within the reference and the neighboring frames to support multiple frame denoising. VBM4D (Maggioni *et al.*, 2012) and VBM3D (Dabov *et al.*, 2007b) take the BM3D algorithm further to video denoising with faster homography flow-based alignment. We instead estimate per pixel correspondences for a more fine-grained alignment. When capturing a burst of images of a potentially dynamic scene with a handheld camera, each image will show slightly different content. In order to effectively utilize information from those multiple frames for denoising, the frames need to be aligned (Bhat *et al.*, 2021a).

For that purpose, we propose (Mallick *et al.*, 2023b) a novel alignment module with the help of pixel-wise iterative dense correspondence matching. (Tico, 2008b) demonstrates a block matching approach within the reference and the neighboring frames to support multiple frame denoising. VBM4D (Maggioni *et al.*, 2012) and VBM3D (Dabov *et al.*, 2007b) take the BM3D algorithm further to video denoising with block matching for alignment.

Handcrafted keypoints detectors are generally robust to domain changes, but are more time-consuming to craft than their learnable counterparts. Additionally, Strong scene changes, however, severely affect the performance of the handcrafted methods, while Neural network optical flow models can leverage information beyond patch-level correspondence information to predict dense correspondences, i.e. estimating pixel motion between consecutive frames of a video (Bruhn *et al.*, 2005) . Some of the first learning based optical flow methods used simple CNN architectures (Dosovitskiy *et al.*, 2015; Ilg *et al.*, 2017) . Recently, they were superseded by recurrent techniques like RAFT (Teed and Deng, 2020) or transformer-based architectures like FlowFormer (Huang *et al.*, 2022). This current state-of-the-art techniques are very good and very close to ground truth (Butler *et al.*, 2012) . In our approach, we utilize the success in the optical flow field by using a pretrained RAFT implementation provided in torchvision (Paszke *et al.*, 2019).RAFT provides the high-quality pixel-wise correspondence alignment that we rely on for our denoising approach. We further tackle the problem of neural correspondence localization in our next work.

2.5 Neural correspondence learning

Finally, as an ongoing work we reflect on the problems faced while matching correspondences, and we propose a novel feature-based correspondence localization network. Inspired by the clear patterns observable in the transition of slices in cost volumes, we try to change the inner working of our correspondence matching approach. In the light of MVS depth estimation, the problem would translate to regress the depth iteratively. Instead of searching for a corresponding point, we specifically encode the local 2D neighborhood into a per-pixel feature vector and train an evaluating network to directly predict the relative localization of two features. Further down the line, we obtain precise correspondences by iterative refinement.

Neural correspondence matching Correspondence matching has been a classical computer vision problem (Lowe, 2004) , and has been widely used in many applications such as robust image editing (Barnes *et al.*, 2009) , stereo correspondences (Michael Bleyer and Rother, 2011) , etc. Recent advances in deep learning has also paved the way for learning correspondences with the help of convolutional neural networks (CNNs). Fully convolutional architecture with pair wise image correspondence estimation (Choy *et al.*, 2016) have been proven to show efficient preservation of geometric or semantic similarity. Dynamic Context Correspondence Network (Huang *et al.*, 2019) has shown to overcome repetitive patterns and local ambiguities while performing semantic matching in multiple scales. Additionally, 3D generative model has been proposed which produce viewpoint and lighting invariant descriptors with self-supervision capabilities (Schmidt *et al.*, 2017) . Additionally, efficient correspondence matching have been utilized in aligning 3D surface data (Steinke *et al.*, 2007) . Dense correspondence matches have also been utilized for wide-baseline stereo, along with context normalization technique (Yi *et al.*, 2018) to process each data point separately and produce order invariant correspondences. Although these models have been proven to be efficient in certain tasks, we aim to develop a disparity estimation module which finds correspondences in a local neighborhood and directly predicts updates on the disparity. This pixel-wise method can process significantly larger number of pixels in comparison to previously proposed correspondence architectures.

Stereo and Correspondence Features for Similarity Measures. In order to estimate depth or disparity with the help of reference and source views, it is very important that a geometric and semantic correspondence matching is established. Previous works explore spatial transformer to mimic patch normalization, which in turn boost accuracy of correspondence matching (Jiang *et al.*, 2021) . One of the major problems with this method is that it is not sensitive to heavy camera motions like multi-view stereo setup, since traditional CNNs are by design equivariant to translations of their input. We leverage steerable CNNs (Weiler and Cesa, 2019) which are equivariant under all isometries of the image plane. The resultant feature learning network is designed similar to U-net (Ronneberger *et al.*, 2015) except the underlying blocks are replaced by the steerable

CNNs which makes the feature network sensitive to translations, rotations and reflections.

Iterative refinement Iterative information propagation has been proposed by (Donné and Geiger, 2019) where output depth error has been taken as input for the subsequent iteration steps. Drawbacks include lack of completeness in quantitative results as this kind of method solely focuses on structure. Additionally, conventional plane sweep based methods solely rely on fixed depth intervals. A possible workaround to compensate for this is to estimate the fractional disparities and take the distribution error rather than evaluating on final output softmax depth (Mohamed *et al.*, 2019) . This method only provides a marginal improvement over the conventional methods, and the aforesaid problems remain for any volumetric plane sweep based methods. One can think of getting rid of these problems by proposing a single view depth estimation, which has been in literature for quite a while (Eigen *et al.*, 2014; Alhashim and Torr, 2018; Laina *et al.*, 2016; Godard *et al.*, 2017a; Carion *et al.*, 2021) . A major drawback for this kind of approach is that in a multi-view scenario, one cannot remove the neighboring view feature contribution. Additionally, experiments have been shown to prove that multi-view based depth estimation has always produced better depth estimation than their single-view or stereo counterpart. Pixel flow estimation powered disparity estimation (Wang *et al.*, 2019; Ranjan *et al.*, 2019) has been proposed where optical flow information has been leveraged for assisted accurate disparity measurement. Aforesaid methods provide an accurate depth prediction, but they are computationally expensive in comparison to our proposal and are not always reliable for accurate depth prediction. Taking cues from these abundant research background, we proceed further to the details of our work.

Chapter 3

High quality frame interpolation

Frame interpolation is the task of generating intermediate frames between existing ones. It is a fundamental problem in computer vision with applications ranging from video editing to virtual reality. Traditional methods often rely on flow-based techniques, which can be computationally expensive and susceptible to errors in motion estimation. In the context of stereo vision, frame interpolation becomes even more challenging due to the inherent disparity between the left and right views. This disparity can introduce additional complexities in motion estimation and pixel synthesis, given the fact that most of the setups lack geometric input such as camera positions as well. This chapter addresses the problem of efficient and accurate frame interpolation for stereo views. Specifically, we aim to develop a method that can generate high-quality intermediate frames while minimizing computational overhead and overcoming the challenges posed by stereo disparity. We begin our initial study by proposing a method to generate intermediate frames between stereo views by training a self-supervised network called Interponet. We have used a prototype small-scale camera rig to capture multiple images per frame from horizontally displaced perspectives. From three of such images, we used the left and right one as input, while the middle frame has been taken as a ground truth that needs to be reproduced. We solve the problem of view interpolation with supervised training of our CNN powered architecture called InterpoNet which directly generates the refined, predicted intermediate perspective image from the input images. Our network performs very well compared to the classical flow-based approaches and other CNN approaches, as further elucidated in the evaluation section.

3.1 Introduction

This research addresses the challenge of efficient and accurate frame interpolation for stereo views, particularly in scenarios where geometric information, such as camera positions, is unavailable. Traditional flow-based methods often struggle with the inherent disparity between stereo images, leading to suboptimal results. To overcome these limitations, we propose a novel self-supervised approach that leverages deep learning to directly predict intermediate frames from a pair of stereo images. Our work draws inspiration from previous research on video frame synthesis, which has seen significant

advancements through the use of convolutional neural networks (CNNs). Flow-based approaches have been commonly exploited in DVF (Liu *et al.*, 2017) where video frame synthesis has been done by flowing pixel values from existing ones, rather than hallucinating pixels from scratch. Adding to this, CNNs have also been used to produce multiple intermediate frames with the help of approximating intermediate bidirectional flow for producing very high quality resulting video (Jiang *et al.*, 2017). In addition to this, (Niklaus *et al.*, 2017) also presented a robust end-to-end adaptive CNN that merged motion estimation and pixel synthesis as a single task and interpolated pixels through convolution.

Motivation View interpolation as discussed in this chapter can be used for different applications: On currently emerging hybrid zoom systems in smartphones, which consist of two cameras with different Fields of View (FoV), view interpolation could be used to transition smoothly from one perspective to the other perspective as soon as the captured FoV is smaller than the one of the larger focal length camera. It can also be used to reduce the baseline of stereoscopic 3D movies for viewers with smaller pupillary distances, like children. Beside stereo setups, high quality view interpolation is also demanded in multiview environments, e.g. on light field camera arrays to predict images from positions between cameras to increase the spatial resolution. Another application is panorama stitching, where parallax effects from pictures taken from slightly different locations lead to stitching artifacts. Duplicating the rig and capturing every direction twice, an image from the same location can be interpolated for each direction, which can then be stitched without parallax errors. This approach is realized with the Fraunhofer facetvision camera setup module, which aims at reducing the thickness of smartphone cameras by stitching a usual FoV image from smaller images from multiple thinner cameras.

Contribution In this work, our contributions primarily include the proposal of a novel view frame interpolator network called InterpoNet. Additionally, we have prepared a quadruple view stereo in different possible diverse scenarios.

3.2 Theoretical background

It is crucial to establish a theoretical foundation for the modules that are used as the backbone of our model. We will also revisit some of this knowledge in subsequent sections of this thesis. We will begin with a review of convolutional neural networks (CNNs) and then provide a detailed definition of the interpolation problem.

Convolutional neural networks (CNNs). In order to understand CNNs, we need to understand the feed forward architecture of the very basic artificial neural networks

(ANNs). ANNs laid the foundation of distinguishing data which were not linearly separable. Feed Forward ANNs or multilayer perceptrons (MLPs) consists of at least three distinct layers (Input layer, a hidden layer and an output layer). The output of each layer after a designated operation is fed into the next layer forward, giving the network its name. A forward prediction is called a forward pass. Once the output from the forward pass is obtained, a loss is calculated and the corresponding gradient is backpropagated. The network updates the weights assigned to the layers, and another forward pass is generated until the optimizer reaches and saturates at a global minimum. We will subsequently dive deeper into details in the light of CNNs. One major drawback of this kind of learning is that MLPs lack the spatial context of an input image, as it can only take a 1-dimensional array as an input. This is where CNN comes in handy, as it maintains the spatial integrity of the input images. A CNN usually takes an order 3 matrix as an input which corresponds to the height (H), width(W) and channel(C) dimensions of an image. Each processing module is called a layer as mentioned above and can be made of a convolution layer, activation layer, normalization layer or a pooling layer. Mathematically, a simple neural network can be described as a nonlinearity applied to an affine function. For a given input feature $x = \{x_1, x_2, ..., x_n\}$, passed through an affine function with given non-linearity σ , we have the following:

$$T(x) = \sigma(\sum W_i x_i + b) = \sigma(W \cdot x + b) \tag{3.1}$$

where W and b correspond to given weights and biases. The convolutional layers consist of layers of a rectangular grid of neurons called filter kernels. These filter kernels with certain fixed dimensions have the same weights for the given convolution layer. In short, the image is convolved and passes through the layers of filter kernels, where the weights of the kernels specify the particular layer. It is noteworthy that there may be multiple layers of these convolutions per layer, which in turn utilizes different filters. Each convolutional layer is usually complimented with a pooling layer (There are several types of pooling in literature such as average pooling, max pooling etc.). Given the subsampling factor, blocks from the convolution layers are downsampled with the help of assigning a value, depending on the operation, as mentioned before. For example, the maxpooling operation takes the maximum value among the selected rectangular block of the convolutional layer output. This is usually done to reduce the dimension and later utilize this for fully connected layers for classification tasks. After several convolutional and pooling layers, the image eventually is decimated to a low-dimensional array. Deconvolution(upsampling) blocks can be further added to obtain the original dimension of the input image, or one can add fully connected layers to obtain a final representation for the classification task. For better retention of feature and avoid gradient explosion, one can also complement this with batchnorm layers (Ioffe and Szegedy, 2015) or skip convolutions (He *et al.*, 2016).

In conclusion, for a given image input(I) and a filter kernel (K), the convolutional

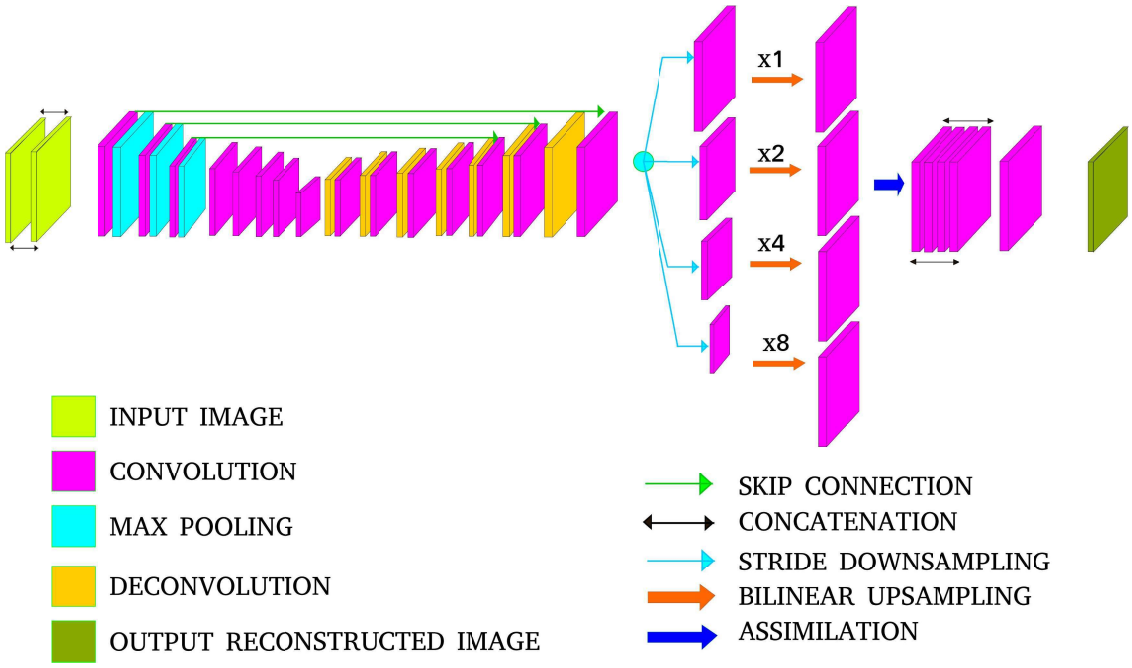


Figure 3.1: InterpoNet network architecture

operation can be summarized as :

$$I * K(i,j) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{3.2}$$

where m,n are the dimensions of the given Kernel space (J Teuwena, 2023).

View interpolation problem formulation To summarize, our goal is to generate new intermediate frames that seamlessly bridge the gap between existing frames, resulting in a smoother transition between the images. Given a set of images (I_n), we want to obtain an interpolated target frame T with the help of a learning frame interpolator model f.

$$T = f(I_n) \tag{3.3}$$

3.3 Our method

The problems faced by conventional methods to evaluate the intermediate frames given such diverse cases, including extremely varying disparities, are considered. Additionally, indoor-outdoor scenarios in different lightning conditions, which also include reflections from semitransparent surfaces and translucent materials, create a huge obstacle in reconstructing the intermediate frame with the help of disparity map analysis. One of the

major advantages in our CNN based model is that it can handle multiple disparities in some special cases as well. We propose InterpoNet, which exploits a global, semi-global and local flow estimation of the multiple stereo frames in order to evaluate the intermediate frame.
Apart from the network which specifically handles intermediate frame generation problem for multiview stereo vision, the dataset used consists of quadruple views and the training strategy includes the usage of only three frames out of the four frames and uses the middle frame as the ground truth and the left and right frames with respect to the middle frame as input to the network.

3.3.1 InterpoNet

Our network has been partially inspired from the Deep Voxel Flow (DVF) (Liu *et al.*, 2017) network, which is an end-to-end fully differentiable network for intermediate video frame synthesis. It uses triplets of consecutive video frames, where the middle frame is used for ground truth for synthesizing interpolated frames. The basic idea is to use a self-supervised network, which helps to borrow voxels from nearby frames, and this is why the network does not have to hallucinate pixels from scratch. It is basically an encoder-decoder which predicts the 3D voxel flow and a subsequent volume sampling layer predicts the target frame. The output 3D voxel flow field consists of a temporal and a spatial component, where the spatial component only denotes the flow from the target frame to previous and next frame. A closer look at the architecture also reveals that there are convolutional subnetworks which predict 3D voxel flow at reduced resolutions, which compensates for smaller to larger motions.
InterpoNet is an end to end network which directly interpolates the target frame without using any temporal flow component, and a network used here has been solely used for interpolating the intermediate frame. DVF uses a trilinear upsampling module which is required to reproduce the target voxel from the flow components, but InterpoNet directly uses the output from the fused multiscale convolutional channels to produce the 3-channel target output. Since the sole purpose of the network is to interpolate a high-quality intermediate camera frame rather than a temporally dependent video frame, only the spatial flow information is utilized, which enforces the necessity to make subsequent changes in approach to the designing of the corresponding network. This is where InterpoNet has been proposed and comes in handy when it comes to reproducing high-quality intermediate target camera frame. As mentioned before, since we are not considering a temporal component, there is no question of a trilinear interpolation in 3D voxel flow field space. Interponet directly produces a three channel interpolated target output frame via simple spatial convolution. Apart from this, few major changes has been made which makes the network unique when it comes to producing interpolating camera frame rather than generating video frames.

Network Architecture InterpoNet is made comparatively deeper than its state-of-the-art competitors (e.g. DVF) for better spatial flow-based reconstruction improvements. A direct comparison is not feasible due to the fact that we are not computing 3D voxel flow field comprising temporal flow component and prompts us to remove the trilinear voxel interpolation step. InterpoNet (Figure 3.1) consists of 7 layers of convolution, 7 layers of deconvolution and 1 bottleneck layer. 3 maxpooling layers are situated in the initial convolution layers, while 4 skip connections have been added for better recovery of the target frame information.
In addition to this, the network consists of additional multiscale flow detection layers in order to capture better global, semi-global and local motion estimations, which is similar to the reasoning employed for DVF (Liu *et al.*, 2017). We have introduced additional convolution layers for compensating minute local movements in high-resolution images. The output from the auto-encoder is taken into consideration and a convolutional down-scaling has been done in 4 different harmonic scales. The first scaling consists of the original resolution (256 x 256), the second scaling consists of half of each pair of resolution (128 x 128) and so on, i.e (64 x 64) and (32 x 32). The subsequent stages of the network consist of a convolution stage, bilinear upsampling to original resolution, and another stage of convolution. It is followed by a concatenation of all the upsampled layers with two final convolution layers, the last layer providing the 3 - channel target output. 3×3 kernels have been used for the entire autoencoder except for the last deconvolution step, which uses a 4×4 kernel. The multiscaling operation uses 5×5 kernels for the down-sampling stage and up-sampling. Down-sampling has been performed by increasing the strides in harmonic order, and upsampling to original dimension has been achieved from bilinear upsampling. *ReLU* activation has been employed for the autoencoder stage and *Tanh* activation function has been used during multiscale fusion stage.

Loss function The loss function used here consists of reconstruction loss L1, L2 and SSIM (Wang *et al.*, 2003) functions. The total cost function generated is the weighted summation of the reduced mean values of the individual loss function components generated. Let $G^{i,j}$ indicates a ground truth pixel and $T^{i,j}$ indicates the corresponding reconstructed pixel where G and T denotes the entire ground truth frame and target frame respectively (ith, jth pixel of the corresponding frame).

$$L1 = \frac{1}{N}\sum(G^{i,j} - T^{i,j}) \tag{3.4}$$

$$L2 = \frac{1}{N}\sum(G^{i,j} - T^{i,j})^2 \tag{3.5}$$

The total corresponding cost function O is as follows:

$$O = \alpha(L1) + \beta(L2) + \gamma(SSIM) \tag{3.6}$$

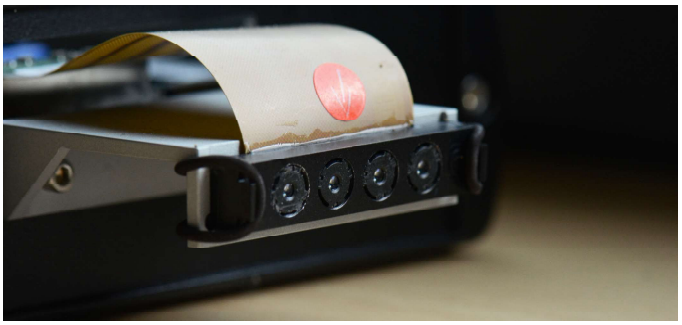


Figure 3.2: Integrated camera rig prototype that was used for capturing the training and evaluation datasets.

The respective parameters have been discussed in the hyperparameter segment in the next section.

3.3.2 Dataset

To train and evaluate our approach effectively, we constructed a unique dataset by capturing realistic scenes with the help of the camera prototype (see Figure 3.2).

Capturing hardware. We used an integrated camera rig prototype (Figure 3.2) which features four 13 MPixel sensors, that are placed behind an array of four lenses with 11mm baseline in-between. Every sensor captures approximately the same Field of View, which is about 65° wide. We captured video sequences with, 2104x1560 pixel resolution and 15 fps. The focus of all lenses, the exposure and gain were controlled manually to guarantee a fixed and common image appearance over each captured video sequence. Intrinsic calibration parameters (focal length, principal point, 2 parameters for each radial and tangential distortion) were obtained individually per sensor according to (Zhang, 2000), followed by one pass of bundle adjustment that recovered an additional 4 degrees of freedom per camera pose relative to the rig (we enforced all focal points of the camera rig to be placed on a straight line). Based on the calibration parameters, we pre-rectified the captured images to be distortion free and have common FoV and principal point as well as identical image plane axes and stored them in 1024x768 pixel resolution for training and inference.

Training dataset. We captured 21 video sequences with a wide variety of content such as indoor and outdoor scenes, different weather conditions and seasons, close- and far-range settings, lab and natural setups that contain portraits; moving objects and people; glossy, semitransparent and refractive objects; repetitive patterns and different focus and exposure settings that lead to different motion and focal blur characteristics. From this

Figure 3.3: Different scenes from our dataset. The first scene was rendered, while the others were captured with our camera rig prototype. The scenes shown here were used for evaluation in Section 3.3.4.

dataset we used 17,240 frames (= 68,960 images) for training. 11 of the captured sequences plus one rendered sequence (Figure 3.3) were used for evaluation.

3.3.3 Training

In order to maintain the high resolution characteristics of the given image datasets, rescaling has been avoided and instead, a patch based approach is employed in order to maintain the detailed spatial variation of the image frames. In the current approach, training has been done on 256 x 256 patches extracted from 1k resolution image. 16 frames per batch has been fed to the network from a dataset containing 15000(x 3) images. The network has been run approximately 182 epochs (170,534 global iteration steps) with for approximately 48 hours deploying 3 TITAN X NVIDIA GPU.

Hyper parameters α , β and γ have been set empirically and their corresponding values are $\alpha = 1$, $\beta = 1$ and $\gamma = 0.5$, to ensure a smooth fall-off in the loss function vs. iteration curve. Adam's optimizer has been used to minimize the objective function using a learning rate of 2×10^{-5}. Batch size used here 16.

3.3.4 Results

We first focused on a direct comparison with DVF, a related method that shares similarities with our approach. However, due to its resolution limitations, this evaluation was restricted to a 256px dimension.This comparison was crucial to understanding our method's performance relative to a state-of-the-art CNN-based approach that was widely recognized at the time of experimentation (see Figure 3.4).

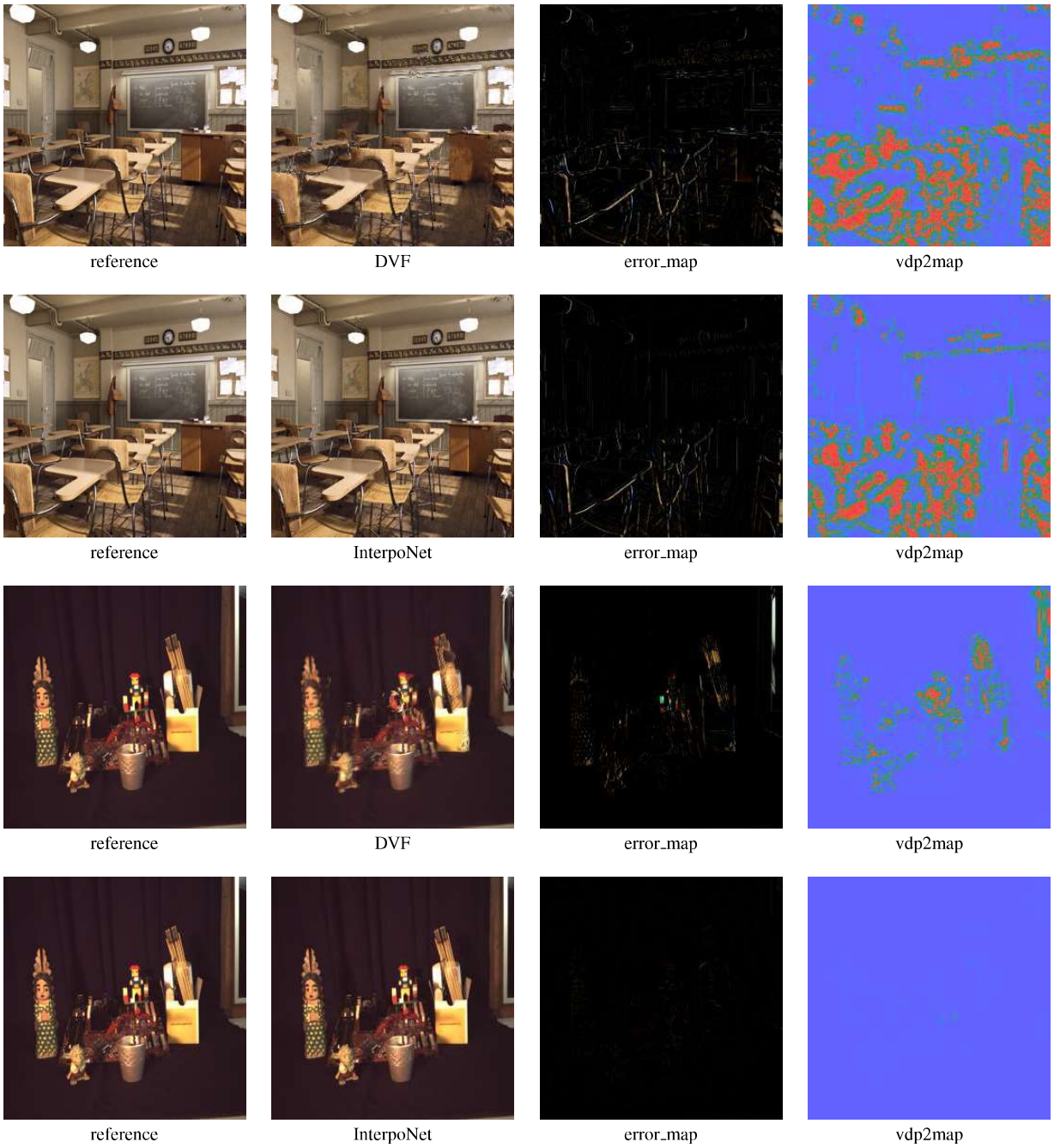


Figure 3.4: The first two rows of figures show the classroom scene evaluation between DVF and InterpoNet. The first row shows the results obtained from DVF while the second row shows the results obtained from Interponet. On closer observation of the VDP2 maps, one can notice the considerable reduction of reconstruction error in the case of InterpoNet.

Quantitative evaluation A statistical metric analysis has been done to provide a score for reconstruction accuracy for comparing DVF and InterpoNet. It can be easily observed that InterpoNet produces superior frame reconstruction. 3.1

Scene	PSNR (DVF)	PSNR (Int)	SNR(DVF)	SNR(Int)	L1(DVF)	L1(Int)
Classroom	21.55	22.46	14.7	15.67	49.45	74.32
Lab	23.64	39.10	12.02	27.48	153.67	5.85

Table 3.1: Statistical analysis of the DVF vs InterpoNet results from different scenarios. It should be noted that the higher the PSNR or SNR, the higher is the reconstruction accuracy; while the lower the L1 parameters, the higher is the reconstruction accuracy. It can be easily seen that InterpoNet beats DVF in almost all statistical parameters in different scenarios.

3.4 Conclusion

Our contribution includes a unique 4 frame multi-stereo image dataset in diverse scenarios. Additionally, this particular dataset is recorded in both high quality 1k and 2k resolution containing more than 15000(X4) frames. Apart from this, our proposed self-supervised InterpoNet reconstructs high-quality 1k images. It has a very high performance when it comes to dealing with very high-disparity cases. It is temporally consistent and contains very few artifacts. It has also shown very nice performance in semi-transparent and reflective surfaces in dark indoor scenarios. In summary, this work is focused on the reconstruction of high-quality images. The network is capable of reconstructing images with a resolution of 2k, provided it contains deeper layers. Although it is comparatively faster than its competitors, real-time implementation remains a challenge for the current model. Looking ahead, we plan to enhance real-time usability by making efficient use of GPUs with the help of TensorRT. We have also noticed that extreme disparity cases sometimes result in spatial inconsistency during reconstruction. To address this, we intend to collect more frames with extreme disparity for our dataset and retrain the network. Moreover, we believe that the implementations for extreme disparity can be leveraged to solve more complex vision problems, such as 3D reconstruction from multiple view images. This will be the focus of our next project.

Chapter 4

Multiple view stereo with semi-supervision

The classical problem of 3D scene reconstruction from multiple views is of significant importance in the field of computer vision. Recent advances in deep learning have led to remarkable results in reconstruction. The training of such models often favors self-supervised methods, as they eliminate the need for hard-to-obtain ground truth data required for supervised training. However, learned multi-view stereo reconstruction is susceptible to environmental changes and must be robust enough to generalize across different domains.

To address this challenge, we propose an adaptive learning approach tailored for multi-view stereo. Our method involves training a deep neural network to be more adaptable to diverse target domains, ensuring its robustness in real-world applications. In this work, our objective is to develop a multi-view stereo reconstruction method that can effectively adapt to new environments without requiring extensive retraining.

Our method employs Model-Agnostic Meta-Learning (MAML) to train base parameters. These parameters are then adapted for multi-view stereo on new domains through self-supervised training. Our evaluations underscore the effectiveness of the proposed adaptation method in learning self-supervised multi-view stereo reconstruction in new domains. This thesis chapter will dive deeper into the details of our approach and its implications.

4.1 Introduction

Dense 3D scene reconstruction based on images from multiple view points is a classical computer vision problem. It has widespread applications in areas such as computer aided design (CAD), virtual tours, augmented reality, cultural heritage preservation, construction maintenance and inspection, or robotics. Given the known view poses and camera intrinsic, multi-view geometry is typically used to find correspondences between pixels of reference along epipolar lines. Early approaches use handcrafted similarity measures for pixels or patches such as photometric similarity or normalized cross correlation. From engineering domains like computer-aided design (CAD) and construction maintenance

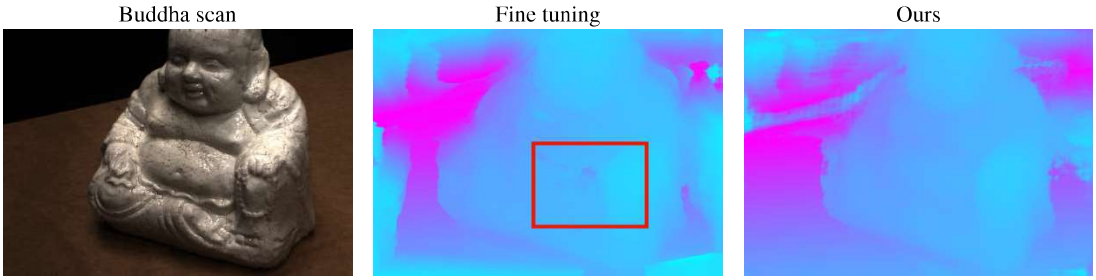


Figure 4.1: Example result for adaptive meta-learning for self-supervised domain transfer. a) 3D scan (from DTU dataset), b) Reconstruction result for pre-training on BlendedMVS dataset without meta-learning and fine-tuning on DTU training set. c) Reconstruction result for our approach with meta-learning on BlendedMVS and self-supervised fine-tuning on DTU. Note the depth artifacts in the red box by the naive fine-tuning approach, which do not occur in our meta-learning approach.

to captivating experiences like virtual tours and augmented reality, the potential impact of multi-view stereo applications are enormous. Traditional multi-view geometry algorithms tackled this challenge by finding correspondences between pixels across images based on known camera poses and intrinsic. Handcrafted similarity measures such as photometric similarity or normalized cross correlation, analyzing patches or individual pixels for matches etc. were utilized for these earlier methods to be effective. Needless to say, these methods were effective in simple scenarios, but these early approaches also had their limitations. Handcrafted features struggled to capture complex scene details, limiting accuracy and robustness. Additionally, they often required manual parameter tuning, making them less adaptable to diverse scenarios. The emergence of deep learning has revolutionized multi-view 3D scene reconstruction. Neural networks can learn powerful feature representations by analyzing large datasets, surpassing the limitations of handcrafted features. This has led to significant advancements in accuracy, scalability, and robustness. Despite the progress, challenges still exist. Deep learning models often require large amounts of labeled data for training, which can be expensive and time-consuming to acquire. Additionally, they can be prone to overfitting on the training data, hindering their performance on unseen scenarios.Addressing these challenges is crucial for the continued advancement of the field. Research is actively exploring techniques like self-supervised learning, which learns from unlabeled data, and domain adaptation, which improves model generalizability in different environments. Deep learning has recently been demonstrated as a capable alternative for learning image features from data, which can excel handcrafted measures (Paschalidou *et al.*, 2018; Yao *et al.*, 2018b; Im *et al.*, 2018; Huang *et al.*, 2018; Ji *et al.*, 2017b; Ummenhofer *et al.*, 2017).

Motivation. State-of-the-art methods for multi-view stereo reconstruction, which are based on deep learning, predominantly use supervised learning approaches. These approaches necessitate vast quantities of ground-truth 3D reconstruction data. However, acquiring such data is not only tedious but also challenging. Existing datasets such as (Jensen *et al.*, 2014; Aanæs *et al.*, 2016; Song *et al.*, 2015) lack data diversity, come with calibration artifacts between the camera and the depth measuring device, or are synthetic. Hence, self-supervised learning methods which can leverage large collections of camera images without the need of ground-truth 3D annotations are preferable. In addition to the aforementioned points, it's crucial for the algorithm to exhibit robustness against environmental or domain changes. This is because it's impractical to train a network with all possible environments included in the training data. Therefore, a learning mechanism is necessary that can offset environmental changes and swiftly adapt to different domains, such as indoor versus outdoor, low light versus bright light, and building architecture scans versus object scans. Furthermore, multi-view stereo presents challenges like the variable baseline between consecutive frames, leading to occlusion uncertainties. This, in turn, compromises the accuracy of depth inference. A motivating example in our context is highlighted in Figure 4.1. Recent developments in meta learning (Finn *et al.*, 2017) demonstrated online adaptation to new tasks of supervised regression models which have been trained on a different set of tasks. In our approach, we propose a variant of model-agnostic meta-learning (MAML (Finn *et al.*, 2017)) for training a multi-view stereo reconstruction network which facilitates self-supervised adaptation to new domains. Our method is grounded in the classical concepts of multi-view stereo (MVS) reconstruction, and it focuses on estimating dense depth in a reference view. Our model extends the network architecture of MVSNet (Yao *et al.*, 2018b) which has been demonstrated to yield state-of-the-art performance for supervised and self-supervised learning. In the initial training stage, we employ our meta-learning approach to train a network on an extensive dataset across multiple domains, all of which have ground-truth depth annotations. The network is trained to enhance its adaptability to new domains via self-supervised training on data lacking ground-truth depth. In the subsequent stage, we carry out self-supervised fine-tuning using data from the new domain. Like MVSNet, our multi-view stereo reconstruction network compares image features in cost volumes. Since our method is model agnostic, we utilize a modified version of the supervised MVSNet (Yao *et al.*, 2018b), which uses discretized depth based probability cost volume computation of neighboring warped frames to the reference frames. This volume is refined with a set of 3D convolutions, and we infer a preliminary depth map by neural regression from this refined volume. Different to the probability map for the depth as in MVSNet, we learn a confidence mask which is utilized to weight pixels for the self-supervised loss in order to compensate for outliers such as occlusions. During testing, the forward pass generates the corresponding set of masks for the different neighboring frames and efficiently computes the self-supervised loss.

Contribution. We demonstrate our adaptive learning approach by training on the BlendedMVS (Yao *et al.*, 2020) dataset which contains a large collection of outdoor scenes (e.g. views of buildings, architecture etc.) and indoor scenes. We fine-tune our pre-trained model using self-supervised training on the DTU dataset (Jensen *et al.*, 2014) which consists of high resolution close scans of objects with different environment and lighting conditions. Our method, evaluated on the DTU evaluation split, is compared to state-of-the-art MVS approaches and variants of our own method, including fine-tuning without meta-learning. The results demonstrate that meta-learning significantly enhances the accuracy of MVS beyond what naive fine-tuning can achieve. Furthermore, our approach outperforms a self-supervised baseline method in terms of reconstruction results. In fact, our experiments reveal that it holds up well against several previous supervised and classical methods across certain metrics. The efficacy of our adaptive learning technique for multi-view stereo is further validated by point cloud reconstruction results. These results show that our adaptive learning method achieves a state-of-the-art overall f-score metric, thereby confirming its effectiveness.

- We propose a novel meta-learning scheme for adaptive learning of multi-view stereo reconstruction, which improves self-supervised domain adaptation.
- We extend MVSNet to learn a confidence mask for per-pixel weighting for self-supervised learning, which handles outliers such as occlusions.
- We demonstrate that our meta-learning approach can improve self-supervised domain adaptation performance over naive pre-training in a supervised way. Our domain-adapted self-supervised multi-view stereo reconstruction achieves improved performance over a self-supervised MVS baseline.

4.2 Methodology

Meta-learning empowers learning models to rapidly acclimate to novel tasks, akin to human learning that effortlessly adapts to new challenges. To achieve this, the given model undergoes training across a diverse set of tasks during the meta-learning phase. In our context, tasks correspond to self-supervised learning in different environments and conditions or domains. Our training data consists of multi-view image collections spanning various environments, which encompasses indoor and outdoor scenarios. The model is adapted over several iterations on a set of training domains, utilizing a self-supervised loss. Following this, the model parameters are optimized using additional training samples. This optimization process facilitates updates that enhance accuracy, which is assessed through supervision using ground truth depth information. The methodology can be summarized in two stages. The model is trained on a larger dataset with ground-truth depth in the first stage using meta-learning. In our experiments, we use the BlendedMVS

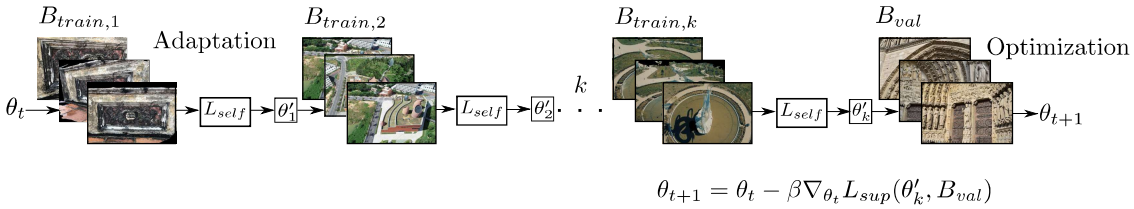


Figure 4.2: Meta-learning for self-supervised multi-view stereo. During a meta-learning iteration, adaptation is performed on k multi-view stereo reconstruction tasks with a self-supervised loss (L_{self}). The adapted parameters θ'_k are evaluated and the base model parameters θ_t are optimized on a validation set using a supervised loss L_{sup} to learn a better starting point for self-supervised parameter adaptation. For the new domain, the resulting base model trained through meta-learning is fine-tuned with self-supervised training.

dataset (Yao *et al.*, 2020) which consists of indoor and outdoor scenes with varying environment conditions - making it ideal for domain adaptation. The model is trained on the training split by first updating cloned model parameters using the self-supervised photometric losses for k 'tasks', where a task refers to multi-view reconstruction of one of the k different scenes. The actual model parameters are then in turn updated using the supervised loss in Equation (4.2) on the validation split, which involves the cloned and updated parameters from the previous step. The model trains network parameters to adapt well by self-supervised training. This is guided through the outer-loop supervised training (see Algorithm 1). The second stage involves fine-tuning the model obtained in the first stage using self-supervised learning on the training data of the target domain dataset (DTU (Jensen *et al.*, 2014) in our experiments). Thereafter, the fine-tuned model is evaluated on the DTU test split. We provide detailed explanation of the methodology in the following subsections.

4.2.1 Meta Learning for Self-supervised Multi-View Stereo

Our meta-learning algorithm for self-supervised multi-view stereo is summarized in Algorithm 1 and illustrated in Fig. 4.2. We split the training dataset D into a training and a validation split D_{train} and D_{val}, the latter with m multi-view examples. Each example consists of a reference view (image with camera pose) and N neighboring views.

We adapt the base model parameters θ for k multi-view examples, $B_{train,i} \subset D_{train}$ each consisting of one reference view and N neighboring views of a scene using a self-supervised loss (L_{self}, Eq. ((4.3))). Starting from the base parameters $\theta'_0 = \theta$, for each multi-view example i we perform the gradient update steps

$$\theta'_i = \theta'_{i-1} - \alpha \nabla_{\theta'_{i-1}} L_{self}(\theta'_{i-1}, B_{train,i}), \tag{4.1}$$

where α is a learning rate.

The base model parameters are optimized to improve the quality of the updated model parameters θ'_k with a supervised loss L_{sup} (Eq. ((4.6))) on a sampled multi-view example $B_{val} \subset D_{val}$ consisting of one reference view and N neighboring views from the validation split,

$$min_\theta(L_{sup}(\theta'_k, B_{val})). \tag{4.2}$$

Note that θ'_k is a function of θ through the updates in Eq. ((4.1)). The supervised loss measures the discrepancy between the predicted depth and the ground truth depth.

The intuition behind this two-step update scheme is that the base model parameters are changed to a better starting point for learning model parameters on different domains with the self-supervised loss. For a new dataset, we use the base parameters θ for fine-tuning to the new domain (i.e an entirely unseen dataset with different conditions and environment) using self-supervised training.

Algorithm 1 *Adaptive learning for self-supervised multi-view stereo.*

Data: *Dataset split* D_{train}, D_{val}, *hyperparameters* k, α, β
Initialize base model parameters θ
while *not converged* **do**
  Sample k multi-view examples $B_{train,i} \subset D_{train}$
  Initialize model parameters $\theta_0 = \theta$
  for $i \in \{1, \ldots, k\}$ **do**
    Compute adapted model parameters $\theta'_i = \theta'_{i-1} - \alpha \nabla_{\theta'_{i-1}} L_{self}(\theta'_{i-1}, B_{train,i})$; `// Adaptation`
  end
  Sample batch $B_{val} \subset D_{val}$
  Perform gradient descent step on base model parameters θ *to minimize* $L_{sup}(\theta'_k, B_{val})$
  with learning rate β ; `// Optimization`
end

4.2.2 Network Architecture

While our adaptation module is model-agnostic, we base our network architecture on the MVSNet (Yao *et al.*, 2018b) model. MVSNet has demonstrated state-of-the-art performance for both supervised and self-supervised (Khot *et al.*, 2019) training. Besides changing the training schemes with our meta-learning approach, we also augment the network with predicting confidence masks which are in turn used for self-supervised fine-tuning on novel domains (additional details can be found in the supplementary material). We base our network architecture on MVSNet (Yao *et al.*, 2018a). We do not use the depth refinement module, but extend the network with a subnetwork which predicts a confidence mask for the self-supervised loss. We provide a comparison of the two architectures in Fig. 4.3. The confidence mask subnetwork is a 4-layer CNN with

a sigmoid activation unit at the end to generate values between 0 to 1. The confidence mask prediction network consists of a combination of two basic sub-blocks. The first sub-block consists of a 2D convolutional layer (kernel size=3, stride=1) followed by a BatchNorm layer and ReLU as its activation function. This sub-block layer is used 3 times successively and then is followed by a final sub-block which consists of a 2D convolutional layer (kernel size=3, stride=1) followed by sigmoid activation function. The subnetwork receives as input the out-of-image projection masks and a photometric error maps for each neighboring view. The photometric error maps are determined by warping the neighboring views to the reference view and taking the difference.

4.2.3 Learning Confidence Masks for Self-supervised Domain Adaptation

A major problem in learning multi-view stereo is to handle occlusions and out-of-image projections correctly when quantifying the loss on the predicted depth maps. We take inspiration from (Tonioni *et al.*, 2019) to learn a confidence mask during meta-learning, which is used to improve the fine-tuning of the network on the new domain. While the approach in (Tonioni *et al.*, 2019) has been proposed for learning dense reconstruction from stereo images of a constant-baseline stereo rig, multi-view stereo poses additional challenges due to the varying baselines between reference image and neighboring frames.

Our network learns a confidence mask for each pair of reference image and neighboring frame in a multi-view training set. Fig. 4.4 provides an example of the confidence masks learned by our approach for different neighboring views. The out-of-image projection mask $C_{proj} : \Omega \rightarrow [0,1]$ can be directly determined from the relative camera pose between the views and the predicted depth map. We train an additional component of our network architecture during meta-learning, which predicts a confidence mask $C_\tau : \Omega \rightarrow [0,1]$ with learnable parameters τ for learning to down weight pixels in the loss of occlusions and other outliers. The final per-pixel mask is obtained by the product of the two masks at each pixel. The masks are used for the self-supervised loss to compensate for occlusions due to view pose changes. Note that the parameters τ are included into θ and updated during the meta-learning stage. They are held fixed when fine-tuning on a new domain dataset in the second stage.

The confidence mask network is a 4-layer CNN with sigmoid activation at the end to generate values between 0 and 1. The photometric warping error between reference I_{ref} and neighboring image I^i, and the out-of-image projection mask are concatenated and used as an input to the network that predicts the confidence mask.

4.2.4 Training Losses

Self-supervised Losses. Self-supervised losses are used for adaptation during meta-learning and for fine-tuning on the new domain. The self-supervised loss comprises two

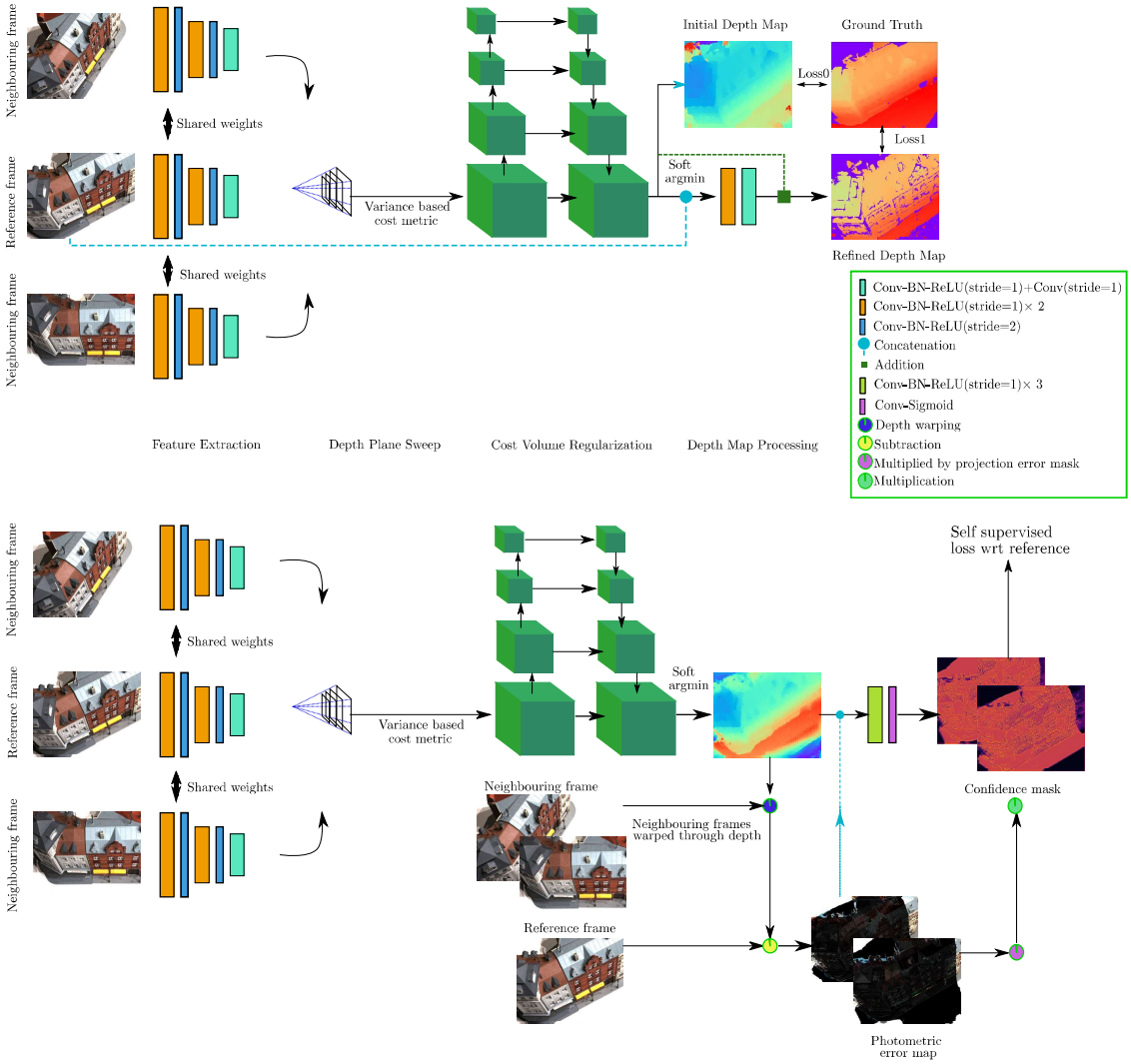


Figure 4.3: Network architecture difference between MVSNet (top) and our model (bottom). Our model builds on the initial stages of MVSNet: Deep features are extracted from the reference frame and the neighboring frames. A plane-sweep cost volume is determined by homographic warping of the neighboring feature maps to the reference view in a set of depth planes. This cost volume is refined in an encoder-decoder architecture and a depth map is obtained using a soft argmin operation. In case of MVSNet, this initial depth map is further improved by a refinement network. Supervised losses are determined that compare the refined depth with ground truth. In our model, we do not use the refinement branch. Instead, we determine photometric error maps by warping the neighboring frames to the reference view and comparing them with the reference image. These photometric error maps are input to a confidence mask prediction network, which also receives out-of-image projection masks. Finally, a self-supervised loss is computed by utilizing the confidence mask on the photometric error maps.

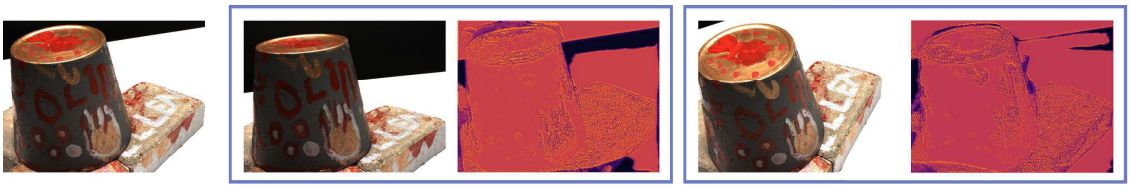


Figure 4.4: *From left to right*: reference image frame, first neighboring frame, predicted confidence mask for first frame, second neighboring frame and predicted confidence mask for second view. Different pairs of reference and neighboring frames have different outliers such as occlusions, reflections, etc. We learn a confidence mask during meta-learning to downweight uncertain pixels for the self-supervised training and fine-tuning on new domains (darker color correspond to lower confidence in the visualization).

components,

$$L_{self}(\theta, B) = L_{\text{recon}}(\theta, B) + \gamma_{\text{smooth}} L_{\text{smooth}}(\theta, B), \tag{4.3}$$

a reconstruction loss L_{recon} and a smoothness loss L_{smooth}, where θ are network parameters and B is a data example consisting of a reference frame and N neighboring frames.

The reconstruction loss measures the image-based consistency between the reference and the N neighboring views given their relative camera pose and the predicted depth map.

$$L_{\text{recon}}(\theta, B) = \sum_{i=1}^{N} \gamma_{\text{photo}} \left\| C^i_\tau(\theta, B) \odot C^i_{proj} \odot \left(I_{ref} - I^i_{warped}(\theta, B) \right) \right\|_1 + \gamma_{\text{ssim}} \left\| 1 - SSIM(C^i_{proj} \odot I_{ref}, C^i_{proj} \odot I^i_{warped}(\theta, B)) \right\|_1, \tag{4.4}$$

where γ_{photo} and γ_{ssim} are weighting factors and $\odot$ denotes pixel-wise multiplication. We use a combination of a photoconsistency measure and the structural similarity index (SSIM (Wang *et al.*, 2004)). The reference image is I_{ref}, $I^i_{warped}(\theta, B)$ is the i^{th} neighboring frame warped to the reference frame given the depth predicted by the network and the known camera parameters. C^i_{proj} is the out-of-image projection mask which excludes the out of bound pixels while warping and $C^i_\tau(\theta, B)$ is the predicted confidence mask for the i^{th} frame. The structural similarity index (Wang *et al.*, 2004) quantifies the similarity between I_{ref} and I_{warped} in patches centered on the pixels and has been used in the literature (Khot *et al.*, 2019; Godard *et al.*, 2017b) since it measures texture similarity while being more robust to lighting changes than the photometric L1 loss.

An edge-dependent smoothness prior on the predicted depth maps with respect to the reference image is applied in order to encourage smoothness of the depth map. The smoothness loss for the predicted depth map $D(\theta, B)$ is

$$L_{\text{smooth}}(\theta, B) = \sum_{(x,y)} \left| \partial_x D_{x,y}(\theta, B) \right| e^{-\left\| \partial_x I_{x,y} \right\|_2} + \left| \partial_y D_{x,y}(\theta, B) \right| e^{-\left\| \partial_y I_{x,y} \right\|_2}, \tag{4.5}$$

where x, y range over the pixels in the reference frame.

Supervised Loss. For evaluation during meta-training, we use an L1 supervised loss on the depth map $D(\theta, B)$ predicted by the network to compare it with the ground truth D_{gt},

$$L_{sup}(\theta, B) = \left\| D(\theta, B) - D_{gt} \right\|_1 . \tag{4.6}$$

4.3 Experiments

We evaluate our approach on a large-scale MVS dataset with ground-truth for the meta-learning stage and demonstrate domain adaptation on a smaller-scale MVS dataset from a different domain. For meta-learning, we use the BlendedMVS dataset (Yao *et al.*, 2020) which has a mix of outdoor and indoor scenes. The dataset contains over 17k high-resolution images covering a variety of scenes, including cities, architectures, sculptures and small objects. The dataset is divided into training and validation sets, which we use for the meta-learning. Domain adaptation is tested on the DTU (Jensen *et al.*, 2014) dataset, where we fine-tune the model on the training split and evaluate its final performance on the test split.The DTU scans consist of different objects in a different indoor environment with varied lighting conditions.

4.3.1 Training Details

The number of neighboring frames N is 2 for meta-learning and fine-tuning. The model is tested with $N = 4$ frames. The number of depths ($d = 256$), input resolution ($H = 512, W = 640$) and output depth resolution ($H = 128, W = 160$) are initialized as in the original MVSNet setup for fair comparison (Yao *et al.*, 2018b). Learning rates are selected as $\alpha = 10^{-4}$ and $\beta = 10^{-4}$. The model is fine-tuned with a learning rate of 10^{-7} and a batch size of 4 multi-view examples with one reference frame and N neighboring frames each. The self-supervised loss weights are set to $\gamma_{\text{photo}} = 5, \gamma_{\text{ssim}} = 1$ and $\gamma_{\text{smooth}} = 0.01$. For meta-learning we use $k = 3$ multi-view examples in each update cycle. The total number of batched data chunk during meta-training consists of $((k+1) \times 4 =)$ 16 datapoints. The meta-training and testing have been performed on the same hardware configuration (4 NVIDIA Titan RTX GPUs) using a PyTorch implementation. We used the Learnable (Arnold *et al.*, 2019) library for implementing first-order MAML.

4.3.2 Depth Map Fusion

Similar to MVSNet (Yao *et al.*, 2018b), we fuse the predicted depth maps into point cloud reconstructions using (Merrell *et al.*, 2007)[1]. The method determines a subset of

[1] We use the open-source implementation at `https://github.com/xy-guo/MVSNet_pytorch` with its default parameter setting

the images using the view selection score of COLMAP (Schönberger and Frahm, 2016). Their depth maps are projected to 3D points in a common coordinate frame. Matches of points in neighboring views are found through reprojection into the images. Points with reprojection distance error with threshold < 1 and relative depth difference with threshold < 0.01 are averaged to obtain the final point cloud. We reconstruct point clouds by fusing the generated depth maps for those pixels with confidence above threshold > 0.8.

4.3.3 Quantitative Results

The fine-tuned model is evaluated on the DTU test split (Yao *et al.*, 2018b; Ji *et al.*, 2017b). We use the evaluation metrics as in (Aanæs *et al.*, 2016). The *accuracy* distance metric is measured as the distance from the estimated reconstruction to the ground-truth, encapsulating the accuracy of the estimated points. The *completeness* is measured as the distance from the ground-truth reconstruction points to the estimated reconstruction, encapsulating how much of the surface is captured by the MVS reconstruction.*Overall* is the mean of *accuracy* and *completion* (see Table 4.2). Additionally, we report the *overall F-score* metric (Knapitsch *et al.*, 2017) at inlier thresholds of 1 mm and 2 mm. We utilize (Zhou *et al.*, 2018) for calculating the precision and recall (see Table 4.2: %-age (percentage) columns). The F-score is the harmonic mean of precision and recall.

The results in Table 4.2 demonstrate that our method can improve results over its self-supervised baseline MVSNet in (Khot *et al.*, 2019). It is second to (Dai *et al.*, 2019) in terms of overall metric among self-supervised methods. Filtering with the confidence mask can lead to higher accuracy in favor of lower completeness. Note that our method achieves state-of-the-art results in the overall F-score measures at 1 mm and 2 mm inlier threshold compared to self-supervised and classical methods. Remarkably, it fares similar to one of the supervised methods (SurfaceNet) in several metrics. We also provide evaluation results on the DTU Buddha scan (see Table 4.1).Results of several classical and supervised methods are taken from (Paschalidou *et al.*, 2018). Supervised MVS-Net (Yao *et al.*, 2018a) fares best, while our self-supervised method ranks second and outperforms supervised and classical methods, highlighting the efficacy of our meta-learning approach. All the aforesaid results are in comparison to the state-of-the-art work at the time when this work was done.

4.3.4 Qualitative Results

Figure 4.6 display the qualitative evaluation of our proposed method with respect to supervised methods ((Yao *et al.*, 2018b; Ji *et al.*, 2017b)). Our method provides a superior completeness and as it can be observed from the reconstruction, some surrounding structures are also reconstructed which are not present in the ground truth.

method	accuracy	completeness	overall
MVSNet (Yao *et al.*, 2018a) (Sup DTU)	**0.234**	**0.278**	**0.257**
Ours best (Meta PT bMVS, Sup FT DTU)	0.455	0.335	0.395
Ours (no mask) (Meta PT bMVS, Sup FT DTU)	0.483	0.339	0.412
SurfaceNet (Ji *et al.*, 2017a) (Sup DTU, from (Paschalidou *et al.*, 2018))	0.738	0.677	0.707
Hartmann et al. (Hartmann *et al.*, 2017)(Sup DTU, from (Paschalidou *et al.*, 2018))	0.637	1.057	0.847
RayNet (Paschalidou *et al.*, 2018) (Sup DTU, from (Paschalidou *et al.*, 2018))	1.993	0.481	1.237
Ulusoy et al. (Ulusoy *et al.*, 2015) (C, from (Paschalidou *et al.*, 2018))	4.784	0.953	2.868
ZNCC (Häne *et al.*, 2014)(C, from (Paschalidou *et al.*, 2018))	6.107	0.646	3.376
SAD (Häne *et al.*, 2014)(C, from (Paschalidou *et al.*, 2018))	6.683	0.753	3.718

Table 4.1: Ranking of several methods by overall metric on the DTU Buddha dataset. Lower is better (best as bold). C: classical, Sup: supervised, Self: self-supervised, Meta: meta-learning. Our self-supervised meta-learning approach performs better than several supervised and classical methods. PT bMVS, FT DTU denotes pre-trained with Blended MVS dataset and fine-tuned with DTU dataset. DTU denotes trained on DTU dataset.

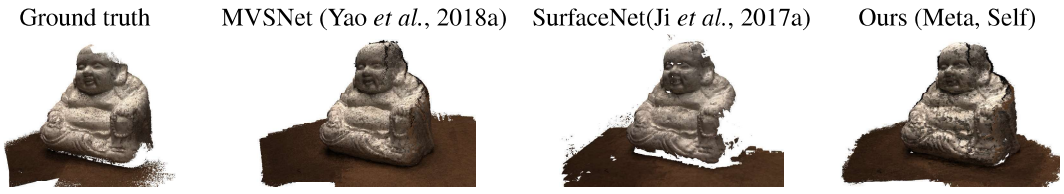


Figure 4.5: Point-cloud reconstruction results on the DTU Buddha dataset. The qualitative results of our meta-learning approach appear superior to supervised SurfaceNet (Ji *et al.*, 2017a) and fairly close to MVSNet (Yao *et al.*, 2018a).

method	acc.	comp.	over.	prec.	rec.	over. F	prec.	rec.	over. F
				(1 mm) in %			(2mm) in %		
Camp (Campbell *et al.*, 2008) (C)	0.835	0.554	0.695	71.75	64.94	66.31	84.93	69.93	74.36
Furu (Furukawa and Ponce, 2010)(C)	0.612	0.939	0.775	69.55	61.52	63.26	77.3	64.06	70.06
Tola (Tola *et al.*, 2012)(C)	0.343	1.19	0.766	90.49	57.83	68.07	92.35	60.01	72.75
MVSNet (Yao *et al.*, 2018b)(Sup DTU)	0.396	0.527	0.462	86.46	71.13	75.69	91.06	75.70	80.25
Ours (Sup PT bMVS, Sup FT DTU)	0.441	0.387	0.414	83.55	74.25	76.93	88.56	77.63	81.09
Surfacenet (Ji *et al.*, 2017b)(Sup DTU)	0.450	1.043	0.746	83.8	63.38	69.95	87.44	67.87	74.81
MVSNet (Khot *et al.*, 2019) (Self DTU)	0.881	1.073	0.977	61.54	44.98	51.98	85.15	61.08	71.13
MVS2 (Dai *et al.*, 2019)(Self DTU)	0.760	**0.515**	**0.633**	70.56	**66.12**	68.27	-	-	-
Ours (Meta PT bMVS, Self FT DTU)	**0.5942**	0.7787	0.6865	**80.18**	63.58	**68.67**	**90.95**	**69.08**	**76.22**

Table 4.2: Evaluation scores for reconstruction metrics (C: classical, Sup: supervised, Self: self-supervised, Meta: meta-learning). PT: pre-trained, FT: fine-tuned. bMVS: trained on Blended MVS. DTU: trained on DTU. Lower score is better for accuracy (acc.), completeness (comp.) and overall (over.) metrics. Higher score is better for precision (prec.), recall (rec.) and overall F-score (over. F) metric. Blue indicates best among all methods. Best results among methods trained self-supervised on DTU are shown in bold. Our approach demonstrates improved results over its self-supervised baseline MVSNet (Khot *et al.*, 2019). Our method achieves state-of-the-art results in the overall F-score measures at 1 mm and 2 mm inlier threshold compared to self-supervised and classical methods. We even fare similar to a supervised approach (SurfaceNet) in several metrics.

tb

method	acc.	comp.	over. F	prec.	rec.	over. F
				(1 mm) in %		
Self DTU(d=128)	0.881	1.073	0.977	61.54	44.98	51.98
Self DTU (d=256)	1.159	**0.6083**	0.8837	64.85	**64.68**	63.57
Self PT bMVS, Self FT DTU	0.9448	0.6345	0.7896	68.43	63.38	64.42
Sup PT bMVS, Self FT DTU	0.7808	0.6769	0.7288	74.54	64.35	67.49
Ours (Meta PT bMVS, Self FT DTU, no conf. mask)	0.7242	0.8422	0.7832	75.22	60.25	65.31
Ours (Meta PT bMVS, Self FT DTU)	**0.5942**	0.7787	**0.6865**	**80.18**	63.58	**68.67**

Table 4.3: Ablation study (bold shows best results). Acronyms follow Table 4.2. Our meta learning approach achieves better overall scores than the other training variants.

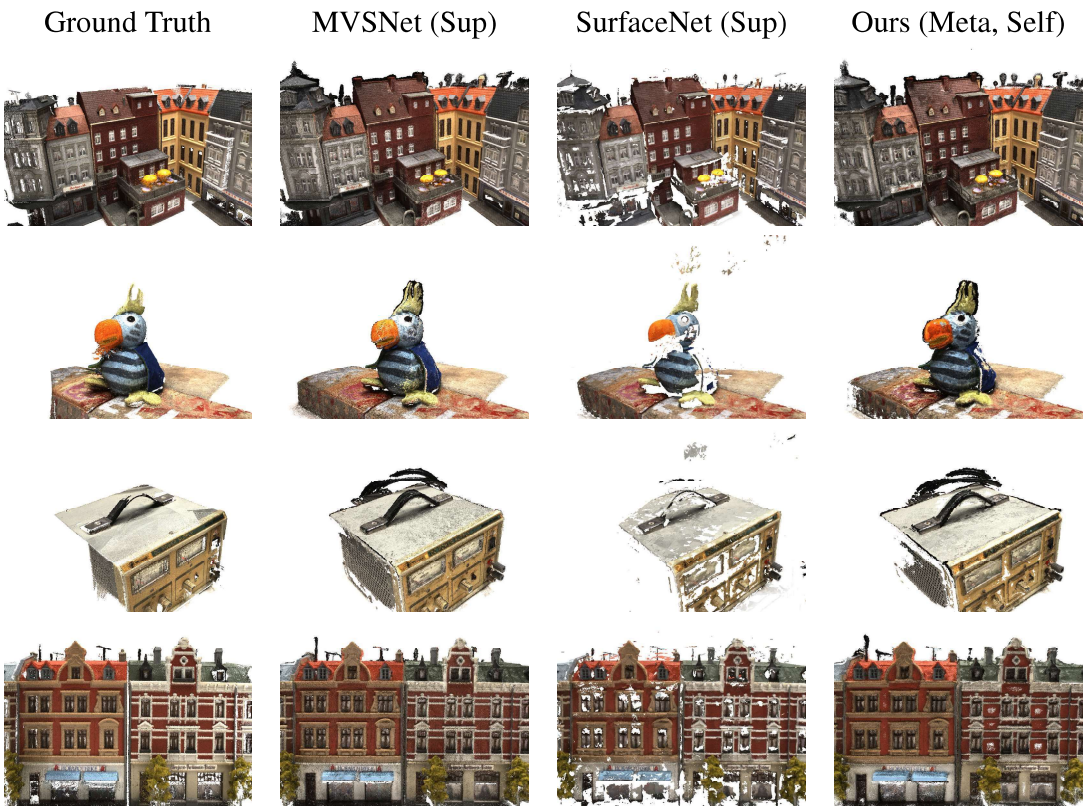


Figure 4.6: Point cloud reconstructions. From left to right: ground truth, MVSNet, SurfaceNet and ours. Our reconstruction results provide a better completeness than SurfaceNet and appear similar to the supervised MVSNet results.

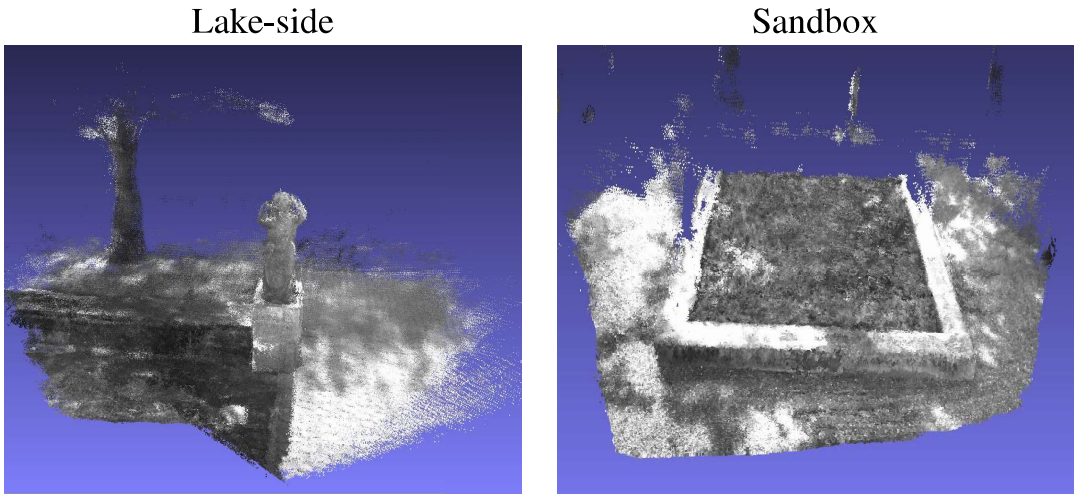


Figure 4.7: Point-cloud reconstruction results of our meta-learning approach on ETH3D low resolution multiview dataset reconstruction.

ETH3D. Further reconstruction evaluation was performed on the ETH3D test dataset for low-resolution multiview stereo. The model was meta-trained on BlendedMVS and fine-tuned on ETH3D low resolution training dataset. The test point-clouds show clear reconstruction results (see Fig. 4.7).

DTU. We provide additional point-cloud reconstruction results on the DTU dataset in Fig. 4.9. We also show depth maps predicted by our approach in Fig. 4.9).

4.3.5 Ablation Studies

We perform ablation studies on the following training conditions:

- Self-supervised MVSNet setup (*Self DTU (d=256)*) similar to (Khot *et al.*, 2019), with twice the depth discretization level (d=256). It was trained on DTU train split, and has different loss hyperparameters (such as reprojection loss weights as proposed in (Khot *et al.*, 2019)).
- Similar as the previous setup, but pre-trained (PT) on BlendedMVS using self-supervised learning (*(Self PT bMVS, Self FT DTU)*) and supervised learning (*Sup PT bMVS, Self FT DTU)*). The model is fine-tuned (FT) on DTU using self-supervised learning.
- Our meta-training setup without the confidence mask training (*Ours (Meta PT bMVS, Self FT DTU, no conf. mask)*).
- Our proposed meta-training setup with the confidence mask training (*Ours (Meta PT bMVS, Self FT DTU)*).

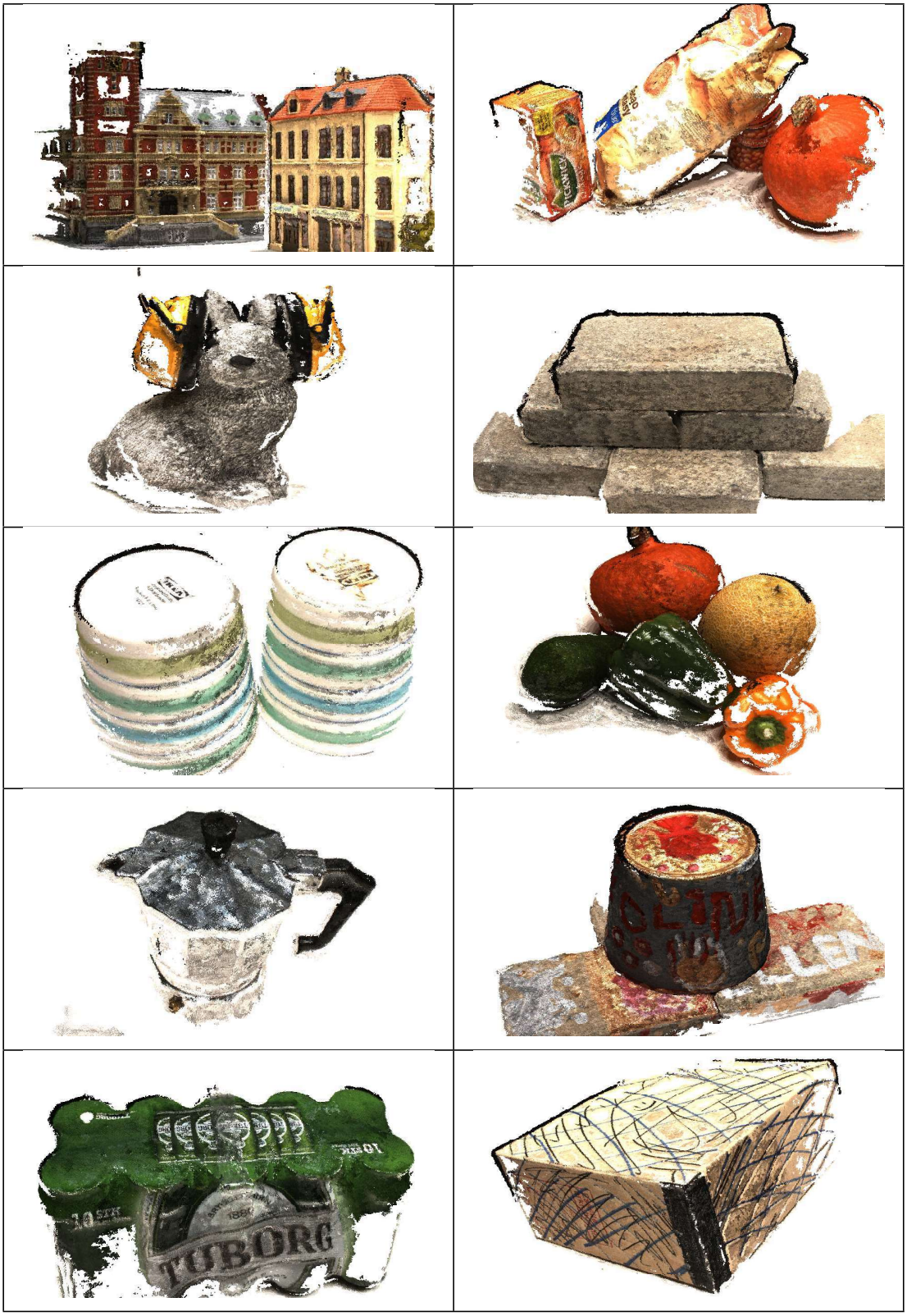


Figure 4.8: Point-cloud reconstruction of DTU evaluation scans using our approach.

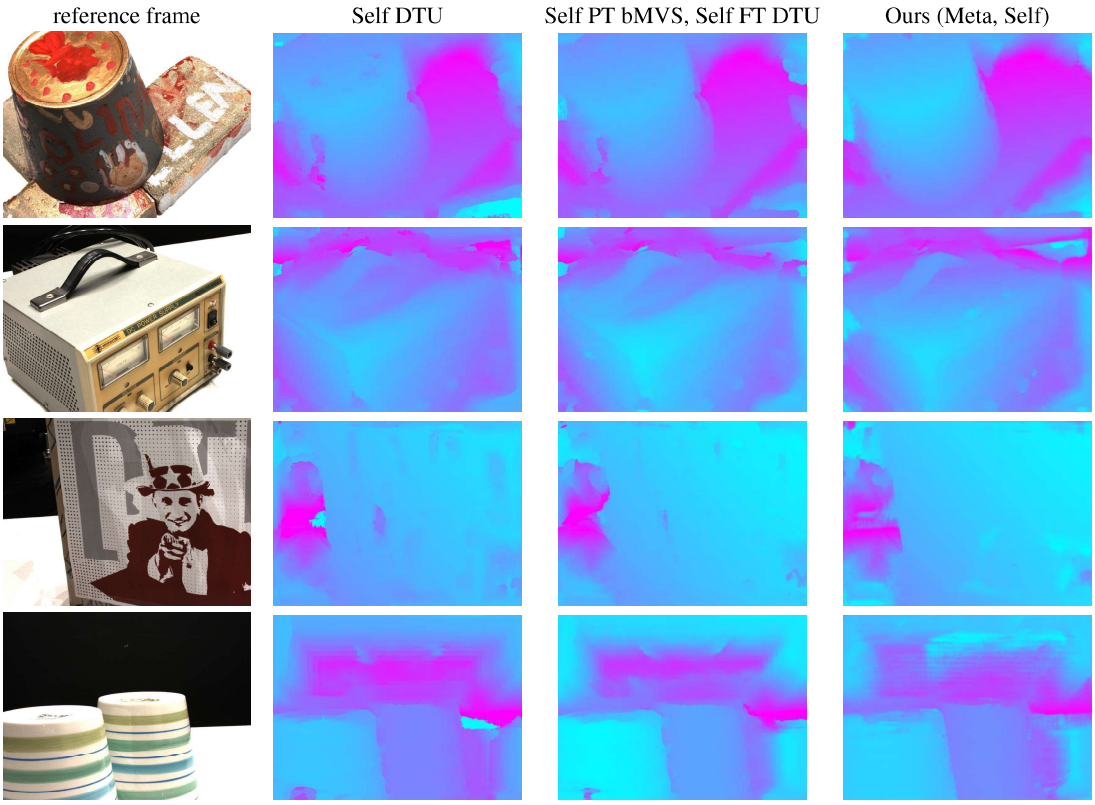


Figure 4.9: Depth maps predicted on the DTU test set. From left to right: reference frame, depth maps predicted by our network trained self-supervised on DTU only, depth maps predicted by our network pretrained self-supervised on BlendedMVS and fine-tuned self-supervised on DTU, our meta-learning approach pretrained on BlendedMVS and fine-tuned on DTU. Our approach predicts smoother depth maps at homogeneous surfaces and provides better completeness.

Table 4.3 shows the results for these variations of our model. The *overall* scores highlight that meta learning outperforms the straightforward fine-tuning strategy (PT bMVS) with the same sequence of datasets, even if it is pre-trained supervised. Confidence weight masks are effective for decreasing the effect of outliers during learning which improves performance.

4.4 Conclusions

Adaptability to new domains through self-supervision is a powerful property, especially for a multi-view stereo learning module where dense ground-truth depth data is tedious and difficult to obtain. We propose a meta learning approach which trains a network for self-supervised adaptation to a novel data domain with changes in environment and conditions. Our approach learns a loss confidence mask for self-supervised learning. In our experiments, we demonstrate that our meta-learning helps to train the network for adapting to new domains using self-supervision. Our approach can improve self-supervised domain adaptation performance over naive pre-training using depth supervision. It achieves reconstruction results which well compare with a previous supervised method and classical methods, and can improve performance over a self-supervised baseline.

Meta learning and multi-view stereo learning is a popular topic in the field of machine learning and computer vision. In the future, we will investigate architectures for high resolution images for an improved and more detailed reconstruction. Apart from this, plane sweep cost volume based learning depth maps generates inaccuracies due to discretization limitations. Pruning methods could reduce memory consumption and improve the results of our meta-learning MVS approach.

One of the significant challenges we encountered while training a model based on volumetric feature correspondence was the requirement for substantial computational power. To mitigate this, we had to limit the resolution of the depth map to a quarter of its original resolution derived from the multi-view image input. This limitation inevitably led to a loss of many reconstruction details during the process of depth downsampling and subsequent depth fusion to create a point cloud.

However, simply upsampling a depth map does not always guarantee an increase in reconstruction precision. In fact, it might introduce additional noisy artifacts, thereby deteriorating the quality of the reconstruction.

To address this issue, our next work will delve deeper into the exploration of effective depth upsampling methodologies. We aim to leverage a guided reference image to enhance the upsampling process and preserve the intricate details of the reconstruction.

In addition, we will also focus on developing computationally efficient solutions for the correspondence matching problem. This is a critical aspect of this research, as it directly impacts the effectiveness and efficiency of our model. We will provide a brief overview of our approach to this problem in the concluding sections of this thesis. This

will offer insights into our strategies for tackling this complex issue and improving the overall performance of our model.

Chapter 5

Joint depth upsampling with vision transformers

Image super resolution (SR) is a classical computer vision problem. A branch of super resolution tasks deals with guided depth super resolution as objective. Here, the goal is to accurately upsample a given low resolution depth map with the help of features aggregated from the high resolution color image of that particular scene. Recently, the development of transformers has improved performance for the traditional image processing tasks, credited to self-attention. Unlike previous methods for guided joint depth upsampling which rely mostly on CNNs, we efficiently compute self-attention with the help of local image attention, which avoids the quadratic growth typically found in self-attention layers. Our work combines CNNs and transformers to analyze the two input modalities and employs a cross-modal fusion network in order to predict both a weighted per-pixel filter kernel and a residual for the depth estimation. To further enhance the final output, we integrate a differentiable and a trainable deep guided filtering network, which provides an additional depth prior. to further establish the robustness of our model, we have performed ablation study and the overall empirical trials demonstrate the importance of each proposed module.

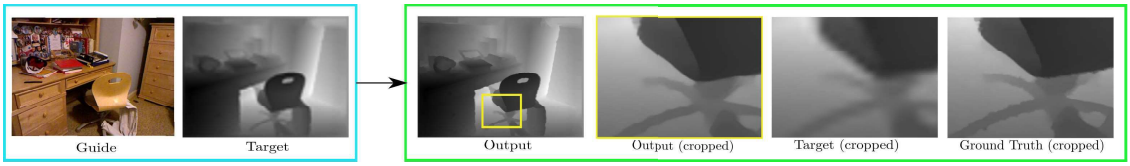


Figure 5.1: Our proposed image local attention guided joint depth upsampling network takes the high resolution guide image and the low resolution bicubic upsampled target image as input. Note the upsampling enhancement in the marked patch generated by our network compared to the 8x upsampled input next to it, as well as the corresponding ground truth patch.

5.1 Introduction

Given a low resolution input, the algorithm tries to compute a corresponding high resolution image. Recent advancements in smartphone photography to satellite imagery employ super resolution for improved image quality and visual clarity. One major application of image SR is joint depth map super resolution, which is applied in robot vision (Islam *et al.*, 2020) and multi-view 3D reconstruction enhancement (Yu and Gao, 2020). Until now, these methods have been computationally expensive and generally produce low resolution depth maps whose clarity, however, can be increased with RGB image guided depth super resolution. In this chapter, we deal with the classic joint depth super resolution (SR) problem. Given a low resolution depth map (target) and a corresponding high resolution RGB image (guide), our task is to compute the corresponding high resolution depth map. Classical depth super resolution methods usually rely on a spatial filtering technique (Kopf *et al.*, 2007) where the input is upsampled by filtering the local neighborhood with weights directly based on the corresponding patch in the guide image which utilizes a spatial representation of the target low resolution depth image and a patch-based guided image extraction method to infer the weights for a kernel to estimate the final depth as a weighted sum of the neighboring pixels. One of the downsides of this kind of method is that it can be time and memory consuming for very high resolution images. Additionally, it can miss homogeneous background information.

Recent developments in machine learning for computer vision applications have also paved the way for guided depth super resolution methods. In general, these applications try to infer the filter kernel weights for each target pixel with the help of a guide RGB input image to perform an adaptive, spatially-varying convolution on the target image. Inspired by the classical joint depth upsampling task, spatial filter weights for the joint bilateral filter have been replaced by a learnable variant for the multi-view stereo task (Yu and Gao, 2020) in the past. We take inspiration from this application and try to infer the neighborhood pixel weights based on an additional local image attention block to extract detailed neighborhood features. Local image attention (Yang *et al.*, 2020) has made it possible to cheaply extract expressive image features for this super-resolution tasks.

Our architecture in Figure 5.2 combines two main ideas. First, we generate an enhanced target input by deep guided filtering networks (Wu *et al.*, 2018) and in parallel estimate per-pixel adaptive filter-kernels for upsampling the target images by fusing the features of both the guide and the target. Second, we estimate corresponding residuals, which are added onto the guided filtered results to refine the depth-map. Both approaches rely on features which are first extracted separately from the guide and target images, then merged for each task employing local attention. The last module of the network incorporates the original low resolution bicubic upsampled target image, guided filtered (GF) target depth and deep guided filtered (DGF) target depth which subsequently proposes the final output from a trainable, weighted per-pixel prediction module.

Features are extracted by a U-Net followed by a self-attention-based transformer encoder to extract local neighborhood information for improved edge-aware guidance. A

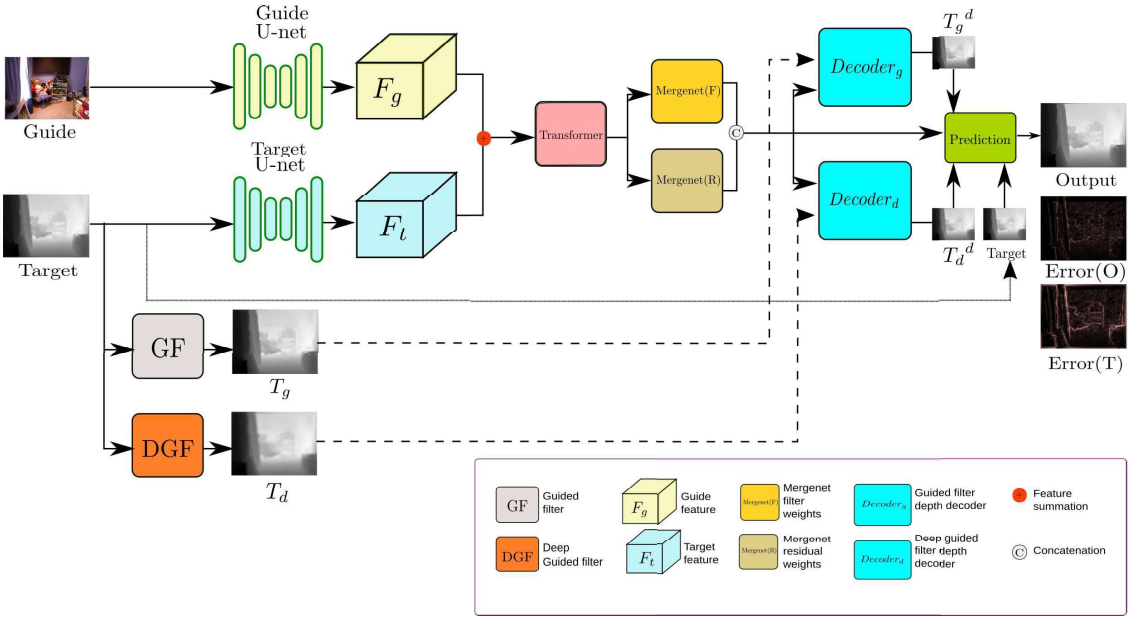


Figure 5.2: Attention Guided Upsampling: Extracted features of the guide and the target are fused by a local transformer to predict filter weights as well as an additional residual. This information is used in the decoder block (shared weights) to predict an upsampled version of T_g and T_d separately. A final prediction layer combines all target predictions. Error(0) and Error(T) visualize the difference to the original and the target input image.

deep merge network (Mergenet) performs efficient cross-modal fusion of local neighborhood features. By combining the RGB and the depth domain, we produce a representation which includes the high-frequency detail from the guide image as well as the coarse depth information from the target image. We use those results twice: As input for the filter-kernels estimation and as input for constructing the residuals from the GF as well as the DGF target map, both of which are a function of the RGB guide image and original bicubic upsampled target image.

Our filter pathway can be interpreted as a generalized adaptive filter with trainable pixel similarity measure. We demonstrate the validity and importance of each module in our ablation study. Our contributions are as follows:

- Local attention and merge block for fusing spatial information from both the guide RGB image and the target depth to provide better super resolution guidance
- Performance comparable to state-of-the-art methods and superior performance in some cases

Figure 5.3: Guide RGB image

Figure 5.4: Target patch

Figure 5.5: GF patch

Figure 5.6: DGF patch

Figure 5.7: Output patch

Figure 5.8: Ground truth

Table 5.1: We demonstrate the contribution of the guided filter blocks. While the target patch is clearly affected by the upsampling artifacts, guided filter map post residual refinement clearly shows remarkable improvements wrt. feature sharpness. Deep guided filtering contributes additional edge aware features. Finally, the weighted pixel prediction module provides additional improvements in the output patch and when compared to ground truth, the overall upsampling artifacts are hardly noticeable.

5.2 Method

The goal in joint depth upsampling is to use a high resolution *guide image* for adding missing detail in an aligned low resolution *target image*. Our network solves this task in four major steps: Guided depth proposals, feature extraction, cross-modality merging and final guided filtering. In addition to the low resolution target image, we obtain a guided filtered target and a deep guided filtered target (Wu *et al.*, 2018). During feature extraction, the guide image and target image are processed independently to generate feature vectors for each pixel. In the merging stage, the features from the guide image are combined with the target features to produce the inputs to the filter section. The GF target and the DGF target are filtered with adaptive kernels and augmented with a residual estimate to the detailed *output* depth map. The final output is a pixel-wise weighted combination of the original target, filtered GF target and the filtered DGF target. In this

section, we describe our architecture from Figure 5.2 in detail.

We propose a combination of image local attention guided depth super resolution which leverages the power of both CNNs and transformer encoders for refinement and residual prediction for the joint depth super resolution task. We discuss the functionalities of our modules in details in the following section.

Our network consists of several subnets(see Figure 5.2). The first block consists of a U-net-based (Ronneberger *et al.*, 2015) feature extractor which generates guide and image features. It is followed by Mergenet for comparing guide and target features to generate guide and target weights. Subsequently, a depth prediction block is implemented to derive two outputs: a filter mask and a fine-grained residual prediction.

Depth guided filters

Direct bicubic upsampling of low-resolution images produces significant artifacts as it does not consider the spatial context. In order to provide better input, we utilize two simple guided filter modules. The first block is a differentiable Guided Filter (GF) layer which takes the low resolution target image I_t and high resolution image I_g to generate a high resolution proposal $T_g = GF(I_t, I_g)$ by a linear transformation (He *et al.*, 2013). The second block consists of a Deep Guided Filter network which integrates the previous guided transformation layers into CNNs and generates corresponding guidance maps $T_d = DGF(I_t, I_g)$. See (Wu *et al.*, 2018) for further details on the gradient propagation through guided filtering and convolutional guided filtering layers.

Feature extraction

Upsampling an image is inherently a local operation, however in order to fill in local details it can help to consider the global context, such as reoccurring patterns or regularities in the occurrence of depth discontinuities. We use U-Nets (Ronneberger *et al.*, 2015) f_g and f_t to extract primary features $F_g = f_g(I_g)$ and $F_t = f_t(I_g)$, with separate weights for the guide image I_g and the target image I_t. I_t has been upsampled with a bicubic filter to the same resolution. Those features are based on the local neighborhood of each pixel in different scales. It should be noted that we do not extract features for T_d and T_g, as they are already jointly encoded with vital guide and target image information and F_g, F_t have sufficient information for further operations in the rest of the architecture.

Next, spatial self-attention compares and relates each stacked pixel feature (F_g, F_t) against its neighbors to better judge its relative importance and to localize important information for the final task of edge-aware upsampling. The self-attention is only computed locally over a sliding window similar to 2D-convolutions (Child *et al.*, 2019; Beltagy *et al.*, 2020; Rae and Razavi, 2020) but with content dependent filter weights. The query, key and value for the attention mechanism are extracted using a linear transformation across the channel dimension which is implemented as 1×1 convolutions. With a quadratic window of side-length p (here $p = 5$) the memory requirement of lo-

cal self-attention is limited to $O(n * p^2) = O(n)$, where n is the number of pixels in the input. Since the patch-size is a constraint for varying and higher resolution cases, a combined feature map would provide richer edge-aware pixel neighborhood information during attention computation in the following stage. Hence, we use a transformer encoder (Vaswani *et al.*, 2017) $\mathscr{T}t$ block built with the aforementioned local attention to enrich our spatially combined target and guide features with detailed local information. The final features $\mathscr{A} = \mathscr{T}(F_g + F_t)$ are computed by applying the transformer to the combined U-Net-feature-maps. We train a variant of the U-net (denoted by f) to aggregate information about the spatial arrangement of the larger local 2D neighborhood of the target I_t and the guide image I_g. The feature extractors use separate weights, as the tasks and the image detail are different for the target and guide image. This module provides us with high dimensional guide image features F_g and target image features F_t. For a given set of queries (Q) and a collection of key-value (K,V) pairs, attention is computed as the weighted sum of all values (Vaswani *et al.*, 2017). The weights are obtained based on the similarities between the query and keys which are usually measured by the dot products of the query with all keys, and divide it with the numeric value of the dimension of keys d_k (see Eq.(5.1)).

$$Attention(Q,K,V) = softmax(\frac{Q.K^T}{\sqrt{d_k}})V \tag{5.1}$$

It is noteworthy that unlike pure convolution based networks which computes the dot product with a set of static kernels, the additional Q,K,V matrices compute dynamic kernels for the given position. Hence, local attention provides additional context based information and tries to figure out which input regions are important. Meanwhile, it is expensive to implement such a module while dealing with a large number of keys, such as ours.We generate our Q,K,V with convolutions and represent them as $C \times H \times W$ (channel, height, width respectively) dimensional tensors. Hence, the term "local" in our context is a region around a pixel with patch size p. The corresponding patch p will have a tensor dimension $C \times p \times p$.On an implementation basis, we divide the feature into $C \times H \times W \times p \times p$ dimension sequence and compute self-attention. It is worth mentioning that in our context of self-attention, they Q,K,V are generated from the same sequence. We utilize this aforesaid self-attention module as a transformer encoder $\mathscr{T}$ block. To summarize, this module takes into account a local region or patch around a given pixel and performs self attention operation over the given keys and values. We perform this separately on the features (F_t, F_g).

$$\mathscr{A}_i = \mathscr{T}(F_i) \qquad \text{where } i = t, g$$

Mergenet

During feature extraction, there is no cross-talk between the information extracted from the guide image I_g and the target image I_t. Although the transformer encoder block

enhances the combined features for depth guidance, only self-attention is not sufficient. The Mergenet is responsible to not only combine both modalities, but provides further enhanced guidance cues. The Mergenet consists of 8 2D convolution layers with ReLU blocks as activations. It consists of two separate blocks (F and R), both working on the same input, that produce the weights W_F and W_R needed for the *Filter* and *Residual* steps to create the final depth prediction. Once we have obtained separate attention-based target $\mathscr{A}_t$ and guide feature $\mathscr{A}_g$ embeddings, our task is to find a proper correlation between these attention embeddings. We propose a Mergenet *Merge*, which is basically a set of 3D convolution blocks. The guide and the target attention embeddings are channel-wise concatenated and used as input to this block. We use two separate sub-nets to obtain guide weights W_g and target weights W_t.

Depth decoder

The decoder module combines the result of a *Filter* module $\mathscr{F}$ with a separately computed depth *Residual* $\mathscr{R}$. The adaptive filter module converts W_F into a per-pixel filter kernel which is convolved with the guided target images T_g and T_d. As the adaptive filter can only produce a weighted average of already existing depth values, the residual module estimates a depth-correction from W_R and W_F. Here, W_R can potentially infuse some additional information from the guide image features estimated in the Mergenet. We utilize two depth decoder blocks with shared weights for refining, We apply the same operation with shared weights to T_g and T_d separately.

$$T_{g,d}{}^{d} = \mathscr{F}_{g,d}(T_{g,d}, W_F) + \mathscr{R}_{g,d}(W_F, W_R) \tag{5.2}$$

This final module is a combination of depth residual $\mathscr{R}$ and depth filter module $\mathscr{F}$. It predicts the final upsampled depth map D_{final} based on the original bicubic upsampled target image I_t and the merged weight tensors (W_g, W_t).

$$D_{final} = \mathscr{R}(W_g, W_t) + \mathscr{F}(I_t, W_g)$$

Depth filter module $\mathscr{F}$ The joint bilateral upsampler (Kopf *et al.*, 2007) has been employed as a classical solution for guided depth upsampling. This method uses a range filter and a spatial filter for predicting the filtered depth output. We take inspiration from its learned counterpart in FastMVSNET (Yu and Gao, 2020) who implicitly try to encode the spatial information with the help of a simple CNN. We utilize a learned version to filter the low resolution target with kernels constructed from W_F, rather than directly using image features. This can be viewed as a generalized adaptive upsampler with an estimated kernel for every pixel coordinate.

To predict the adaptive per-pixel filter mask, we reduce the dimensionality of W_F with the help of a simple 1×1 convolutional network f_{reduce} and utilize it to convolve the target image and obtain $W_{reduce} = softmax(f_{reduce}(W_F))$.

Note that W_{reduce} has k^2 channels, where k is the chosen kernel-size in the filter module. Let $N_k(x)$ be the list of indices in the pixel neighborhood centered at x, then the filter operation can be written as:

$$\mathscr{F}_{g,d}(T_{g,d}, W_F)[x] = \sum_{i=1}^{k^2} W_{reduce}[x,i] \cdot T_{g,d}[N_k(x)[i]] \tag{5.3}$$

Depth residual module $\mathscr{R}$ The depth values produced by the filter are formed by building kernels which are convolved with the low-detail target images. However, the features produced in the Mergenet module already contain the detailed information from the guide image as well as the depth information from the target image. We therefore use W_R and W_F directly to compute an additional residual, which is added to the filter result as indicated in Equation (5.2). With W_R and W_F having the same spatial and channel dimensions we can combine them in an element-wise multiplication and sum up the channels to produce a one-channel residual map. This module can be interpreted as a simple pixel-wise weighted residual prediction from the filter and residual weights (W_R, W_F). We will take cues from the original aligned feature maps in order to provide proper weights for the different target proposals. We interpret W_F as a confidence score for the residual contribution W_R and hence compute the overall weighted residual as :

$$\mathscr{R}_{g,d}(W_F, W_R) = \sum_{i=1}^{C} \text{softmax}(W_F) \cdot W_R \tag{5.4}$$

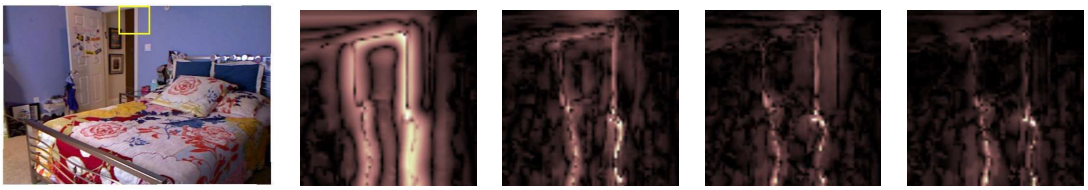
where C is the feature channel dimension of W_R and W_F.

The depth residual module $\mathscr{R}$ computes an edge-aware depth residual with the help of a probabilistic summation approach. For each pixel values in the low resolution target image, the Mergenet evaluates values W_t for possible edge compensation. Our network learns to properly weight these values along the channel dimension. We utilize the guide weight tensor W_g as the texture probability matrix. Hence, the overall residual depth D_{res} is the element-wise multiplication of the weighted probability and W_t.

$$D_{res} = \sum_{i=1}^{C} softmax(W_g) \cdot W_t \qquad \text{where } C \text{ is feature channel dimension}$$

Depth prediction

The final module is a simple pixel-wise weighted depth prediction (*pred*) module that estimates the final output from the three proposed depth maps $(I_t, {T_g}^d, {T_c}^d)$. We will take cues from the original guide feature map and just computed W_R and W_F in order to provide proper weights for the different target proposals. This block consists of 4 convolution layers which estimate the final weights $W_{pred} = softmax(pred(F_g, W_F, W_R))$.



Input HR | Bicubic | DKN | JIIF | Ours

Figure 5.9: Qualitative analysis of joint depth upsampling with the help of our network. We demonstrate (5x) upscaled absolute error maps with respect to the ground truth for the patches marked in green in the input HR (High Resolution) images. We compare our network output with DKN (Kim *et al.*, 2020) and JIIF (Tang *et al.*, 2021) on the NYUv2 (Silberman *et al.*, 2012) test dataset. Brighter regions indicate higher error.

Thus, the final upsampled depth prediction is given as:

$$D_{final} = \sum \text{softmax}(W_{pred}) \cdot (I_t, {T_g}^d, {T_d}^d) \tag{5.5}$$

Loss function

Given the ground truth high resolution target image D_{gt}, and the network output as D_{final}, we train our network with a L1 loss.

$$Loss = \frac{1}{N_p} \sum_{y=1}^{N_p} |(D_{final} - D_{gt})| \tag{5.6}$$

Here, N_p is the total number of pixels in the target image.

5.3 Experiments and Results

Datasets and training setup

Our network is trained on NYUv2 (Silberman *et al.*, 2012) training dataset which consists of 1000 images. We test our trained network on the test split of NYUv2 consisting of 449 images, following the established split protocol of (Kim *et al.*, 2020). Additionally, we also test our network on (Lu *et al.*, 2014) test split and (Scharstein and Pal, 2007) test split, following the test convention of (Lu *et al.*, 2014; Kim *et al.*, 2020). It is to be noted that the network is trained separately with 4x, 8x, and 16x downsampling as input following the mentioned conventions. The downsampled target image is upsampled with the help of bicubic upsampling and is used as an input along with the high resolution RGB guide image. We use an NVIDIA RTX3090 to train our network for approximately 14 hours. Keeping in mind the massive self-attention computation cost which involves memory cost proportional to $p \times p$ per pixel, we use an efficient implementation without

rearranging keys and values in memory with custom CUDA kernels for the attention computations (Zhang, 2019).

Hyperparameters

For the experiments presented in the following sections, our feature channel dimension is set to 128. The NYUv2 training dataset is trained on its full resolution of 480x640 pixels at a batch size of 1. We use 1e-3 as the learning rate for the Adam optimizer. We further decay our learning rate by a factor of 0.2 every 22 epochs. We use a patch size of 5 for the image local attention in all scale scenarios. Our number of heads for the transformer encoder block is set to 1 and the dimension of the feedforward channels is 128. The filter kernel size is set at 7 for all upscale factors. The network is trained end-to-end for 50 epochs.

Quantitative results

We compare our method with other learnable joint upsampling algorithms on multiple test datasets (NYUv2, Lu, Middlebury) as shown in Table 5.4. Our algorithm's performance is comparable and near superior to state-of-the art (during the time of this work) methods with regard to Root Mean Square Error (RMSE). Our network achieves state of the art on the 4x upsampling task of the NYUv2 dataset, Middlebury dataset; and on 8x upsampling task of NYUv2 and Lu dataset. It demonstrates near state-of-the-art performance for 16x upsampling task with respect to the leading methods, which proves our method's generalization capabilities. Although we outperform JIIF (Tang *et al.*, 2021) on multiple test sets for 4x, 8x upsampling, performance drops slightly behind JIIF in 16x upsampling test scenarios. With the increased upsampling factor (16x) the neighborhood context of a fixed-sized local attention encoder is decreased relative to the target image size. We identify this as the main cause for the limited performance. JIIF uses joint implicit function to predict weighted average from the four nearest coordinates in the LR domain for interpolation but learn the interpolation weights and values through a deep implicit representation, which presumably has stronger responses with respect to neighborhood context and eventually has a better generalizable performance. In our test scenario (Lu, Middlebury) we believe that training on a fixed resolution scale is not sufficient for the attention module to be able to gather information on differently scaled structures in the downgraded target images. Still, we are runner-up to JIIF (Tang *et al.*, 2021) for the Middlebury and the Lu dataset. Similarly, as the upsampling factor increases (8x,16x), the neighborhood context interpretation power of a fixed-sized local attention transformer encoder decreases. Nevertheless, our network performs effectively well in comparison to leading methods within a reasonable training period. Nevertheless, our network performs effectively well in comparison to leading methods within a reasonable training period, as we are runner-up to JIIF (Tang *et al.*, 2021) for Middlebury and Lu test datasets.

Qualitative results

We provide a qualitative comparison of the visual clarity of our results with examples from different datasets. In addition, tabulated Figure 5.2 shows a comprehensive comparison with DKN (Kim *et al.*, 2020). Additionally, one can also visualize the NYUv2 (Silberman *et al.*, 2012) test image input along with the 8x upsampled bicubic target input. Overall, one can notice that our network provides a sharper depth output compared to the naive upsampling as well as to the advanced DKN (Kim *et al.*, 2020) approach. For a more comprehensive insight on the generalization, we have also provided the results on the Middlebury test set (Scharstein and Pal, 2007). Compared to the degraded (8x bicubic upsampled) input and the corresponding ground truth, the network output is able to preserve sharp details and only introduced very few interpolation artifacts. Visually, the network recovers a substantial amount of depth data in all settings and displays low absolute error along edges of image structures.

Ablation study

To investigate the importance of individual components in our network, we performed an ablation study by removing each of the six primary training modules from our overall architecture. As presented in Table 5.3, removing the transformer or depth filter hinders the performance of our network. Additionally, one can also observe that the absence of the residual module significantly deteriorates the performance, as the enhanced cross-model transfer from the target embedding during the final depth computation at the end of the pipeline is missing. Additionally, if we do not enhance the transformer fused target and the guide embedding with the help of our proposed Mergenet, the network struggles to transfer the high-resolution texture features to the final depth. Additionally, absence of the depth prediction module also highlights the need of careful pixel selection provided by the fused guidance weights. Introducing GF and DGF provides a much-needed prediction prior, which again infuses the guide RGB features from the beginning and helps the network to predict the final depth from a better target standpoint. This underlines the effect and importance of all the proposed modules in our pipeline.

5.4 Conclusion

We propose a novel architecture to combine the power of CNNs and transformer-based encoders to solve the guided depth upsampling task with efficient local image attention. Our network consists of a local attention block for extracting edge-aware features from both input modalities, followed by merge networks for cross-modal fusion. Eventually, we obtain a depth map with the help of an adaptive depth filter followed by residual prediction computation with corresponding subnetworks. To predict the final depth map, we extend a set of learned adaptive filters by adding a novel depth residual computation. This increases the sharpness of the upsampled depth map.To predict the final depth

map, we extend a set of learned adaptive filters setup by adding a novel depth residual computation to increase the sharpness of the upsampled depth map. The approach yields state-of-the-art results in smaller upsampling cases and performs well on larger upsampling tasks when compared to leading methods. We tune the local attention patch size for the optimal trade-off between compute time and performance. An ablation study demonstrates how each submodule of our network architecture plays an important role in understanding, gathering, and subsequently merging the image features. In future work, we would like to improve performance on a wider range of upscaling factors, minimizing the effort for retraining. For example, certain parts of the network can be fine-tuned to accommodate for different input scales, while the large U-nets stay fixed. Also, a training scheme that trains on multiple datasets and upsampling factors at the same time can improve the generality of the model.

In the previous chapter, we discussed the fundamental building blocks required for the proper reconstruction of learning environments using multi-view images. One of the key tasks in this process is the estimation of depth maps from multiple view inputs. This is further enhanced by the use of a guided image, as demonstrated in this chapter. The primary task, however, is to estimate accurate correspondences between multiple view image inputs. Accurate correspondence alignment is crucial as it provides a comprehensive overview of the homography and motion estimation of the source images. This information can then be utilized to estimate disparities, thereby enriching our understanding of the scene.

With this in mind, we revisit the task of correspondence alignment-based applications. We explore how these applications can be leveraged to enhance our knowledge of the given scene. Specifically, we delve into the role of correspondence alignment in tasks such as burst image denoising and image enhancement. In the upcoming chapter, we will dive deeper into one such application. We will explore how correspondence alignment can be used for burst image denoising and image enhancement. This will involve a detailed discussion on the techniques involved, their implementation, and the impact they have on the overall quality of the reconstructed scene.

Figure 5.10: Guide RGB image I_g

Figure 5.11: Bicubic upsampled target input I_t

Figure 5.12: Network output D_{final}

Figure 5.13: Ground truth depth map D_{gt}

Table 5.2: Qualitative analysis of joint depth upsampling with the help of our network. We demonstrate (5x) upscaled absolute error maps with respect to the ground truth for the patches marked in green in the input HR (High Resolution) images. We compare our network output with DKN (Kim *et al.*, 2020) and JIIF (Tang *et al.*, 2021) on the NYUv2 (Silberman *et al.*, 2012) test dataset. Brighter regions indicate higher error.

Table 5.3: Ablation study for 8x resolution on NYUv2: Numbers indicate RMSE (lower the better) for the case of 8x bicubic upsampling.

Method	RMSE
Without transformer	2.80
Without Mergenet	2.95
Without prediction block (mean)	2.79
Without dgf	2.81
Without cdgf	2.73
Without filters	2.84
Without residuals	2.85
Ours	**2.71**

Table 5.4: Quantitative evaluation (lower is better) for different methods. The evaluation is done in accordance with conventional evaluation metric protocols (Kim *et al.*, 2020; Tang *et al.*, 2021). Here, the RMSE is taken in units of centimeter. Best results are in blue, and second-best results are highlighted in pink.

Method	NYUv2			Middlebury			Lu		
	4x	8x	16x	4x	8x	16x	4x	8x	16x
Bicubic	4.28	7.14	11.58	2.28	3.98	6.37	2.42	4.54	7.38
DMSG((Hui *et al.*, 2016), from (Kim *et al.*, 2020))	3.02	5.38	9.17	1.88	3.45	6.28	2.30	4.17	7.22
DJF((Li *et al.*, 2016), from (Kim *et al.*, 2020))	2.80	5.33	9.43	1.68	3.24	5.62	1.65	3.96	6.75
DJFR((Li *et al.*, 2019), from (Kim *et al.*, 2020))	2.38	4.94	9.18	1.32	3.19	5.57	1.15	3.57	6.77
PAC((Hui *et al.*, 2016), from (Kim *et al.*, 2020))	1.89	3.33	6.78	1.32	2.62	4.58	1.20	2.33	5.19
DKN(Kim *et al.*, 2020)	1.62	3.26	6.51	1.23	2.12	4.24	0.96	2.16	5.11
FDSR(He *et al.*, 2021)	1.61	3.18	5.86	1.13	2.08	4.39	1.29	2.19	5.00
CTKT(Sun *et al.*, 2021)	1.49	**2.73**	**5.11**	-	-	-	-	-	-
DCTNet(Zhao *et al.*, 2022)	1.59	3.16	5.84	1.10	2.05	4.19	**0.88**	1.85	4.39
JIIF(Tang *et al.*, 2021)	**1.37**	2.76	**5.27**	**1.09**	**1.82**	**3.31**	**0.85**	**1.73**	**4.16**
Ours	**1.34**	**2.71**	5.39	**1.07**	**1.86**	**3.57**	0.89	**1.73**	**4.25**

Chapter 6

Burst image denoising

In the era of mobile phone photography and point-and-shoot cameras, the technique of deep-burst imaging has gained significant popularity. This method is extensively employed to achieve a variety of photographic effects, including depth of field, super-resolution, motion deblurring, and image denoising, to name a few. In this study, we aim to address the challenge of deep-burst image denoising. Our proposed solution incorporates a pre-trained optical flow-based correspondence estimation module. This module aligns all the input burst images in relation to a chosen reference frame. To effectively handle the issue of fluctuating noise levels, each of the burst images undergoes a pre-filtering process. This process is tailored with different settings for each image, ensuring optimal noise reduction. Exploiting the established correspondences, one network block predicts a pixel-wise spatially varying filter kernel to smooth each image in the original and prefiltered bursts before fusing all images to generate the final denoised output. The resulting pipeline achieves state-of-the-art results by combining all available information provided by the burst.

6.1 Introduction

The field of burst photography has experienced a significant surge in popularity. This can be attributed to the recent advancements in the development of portable CPUs. These CPUs are not only faster but also lightweight, making them particularly suitable for mobile devices and point-and-shoot cameras. The enhanced processing power of these CPUs allows for rapid-fire sequences of photos, a feature central to burst photography. Moreover, these technological improvements have led to notable enhancements in image quality, specifically in terms of noise reduction and motion blur removal. As a result, the images produced are sharper and clearer, further contributing to the growing prominence of burst photography. Burst photography (Molini *et al.*, 2020; Kawulok *et al.*, 2020) can also be understood as the multi-frame image restoration task (Liba *et al.*, 2019; Hasinoff *et al.*, 2016; Bhat *et al.*, 2021a; Wronski *et al.*, 2019) that has a wider range of applications, even in satellite photography (Valsesia and Magli, 2022; Deudon *et al.*, 2020) for remote sensing. The sensors and lenses in smartphones are much smaller and more lightweight than those of professional cameras, but they collect less light per pixel, which

leads to noisier images. Compensating this by longer exposures could introduce motion blur. As an alternative, a burst of many short and noisy images could be computationally combined into one sharp image. Camera phone APIs already provide denoising algorithms, but they are not optimized for burst denoising.

Current burst denoising methods perform the alignment simply based on the estimated homographies between the burst images (Godard *et al.*, 2018; Bhat *et al.*, 2021b), followed by some pixel denoising. Our proposed pixel-wise alignment based on optical flow is significantly more powerful in compensating for scenes with complex depth and camera or object motion. We still expect the motion between the burst images to be rather small. Our overall architecture is depicted in Figure 6.1. First, we generate enhanced burst inputs by applying the pre-trained self-guided filtering network (SGN) (Gu *et al.*, 2019) for each image to generate pre-denoised bursts. Both, the original and the denoised images are aligned with respect to the reference frame using the RAFT (Teed and Deng, 2020) optical flow network. Based on the aligned images and extracted features, a network block predicts a per-pixel adaptive filter kernel to denoise every pixel in every image. A final fusion block merges all predictions across all bursts into a single output. Secondly, we estimate pixel adaptive filter-kernels, which per pixel describe where to collect color information from the aligned input bursts. The decoder then only applies those kernels, thus produces weighted averages over neighboring pixels from all aligned images. We demonstrate the importance of each module in an ablation study. Our contributions are as follows:

- Optical flow-based alignment of multiple pre-denoised burst images
- Adaptive per-pixel filtering of aligned burst images followed by cross-burst fusion
- Improved denoising performance, especially in low-noise scenarios

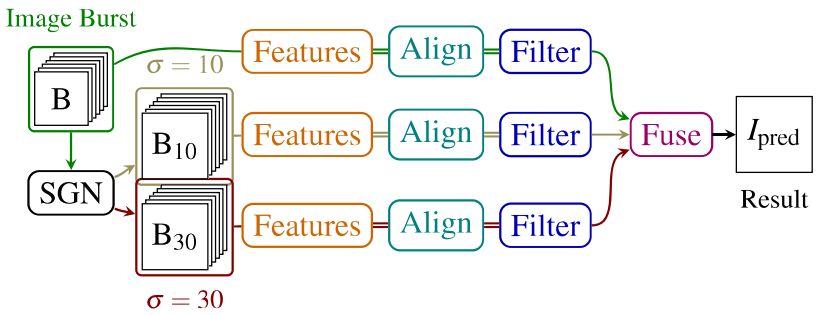


Figure 6.1: Method overview. The input image burst is pre-filtered twice using SGN (Gu *et al.*, 2019) with different filter strengths. For each stack we extract features, align both features and images and then apply a content-adaptive spatial filter with weights derived from the aligned features. The results from all three bursts are fused to predict the denoised output.

Reference | before alignment | after alignment

Figure 6.6: Alignment error between the reference and the last burst image scaled 5 times. Note, how after alignment, differences in the silhouette are no longer present.

6.2 Method

The core idea of our burst denoising method is to first spatially align the pixels in the burst stack. Afterward, each aligned image is denoised by a content-adaptive spatially-varying filter step followed by an adaptive fusion of all processed images (see Figure 6.1, left).

6.2.1 Pre-filtering with SGN

Our approach initiates with the filtration of burst images. The amount of noise in input bursts can vary significantly, even within the same burst. Due to varying degrees of noise and blur due to abrupt camera motion, precise alignment might be difficult. We, therefore, duplicate the input burst into three processing streams. The first stream B uses the original burst, the second stream B_{10} ($\sigma = 10$) a mildly pre-denoised version of the burst and the last one B_{30} ($\sigma = 30$) a strongly denoised version (see Figure 6.1). For this task, any single-frame denoising algorithm could be applied to each individual frame. We apply the pretrained SGN (Gu *et al.*, 2019) to each individual frame, but any single-frame denoising algorithm could be used.The effect of both denoising levels is shown in Figure 6.1. The intermediate results from the different streams will be fused in the last step of our pipeline.

6.2.2 Feature Extraction

To add local context to each pixel, we enrich each image by processing it with a simple CNN. In addition, the estimated noise level of the image is concatenated as the fourth channel before processing.In each processing stream, we produce corresponding feature

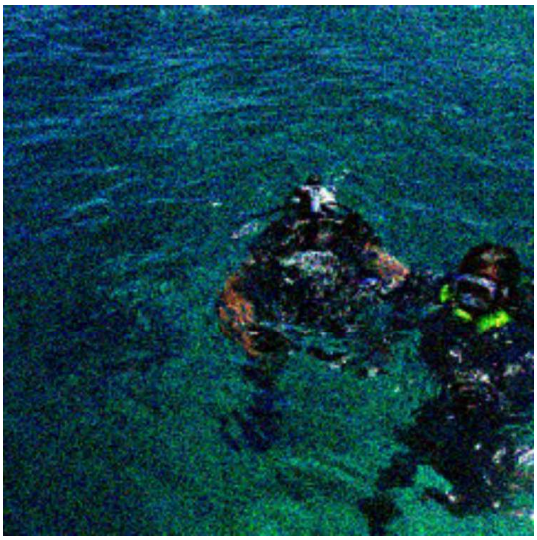


Figure 6.2: Input

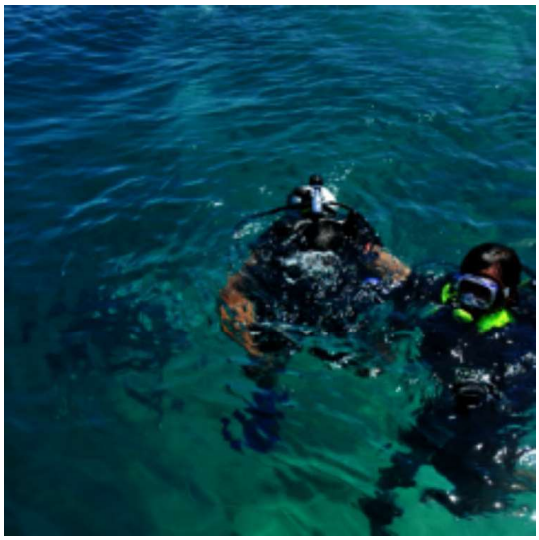


Figure 6.3: Ground truth

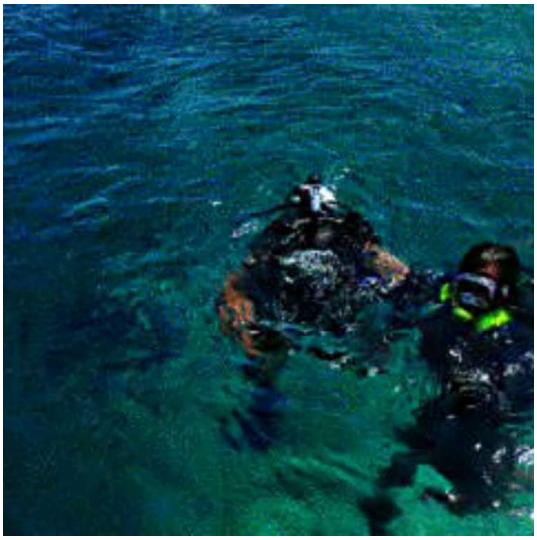


Figure 6.4: SGN $\sigma = 10$

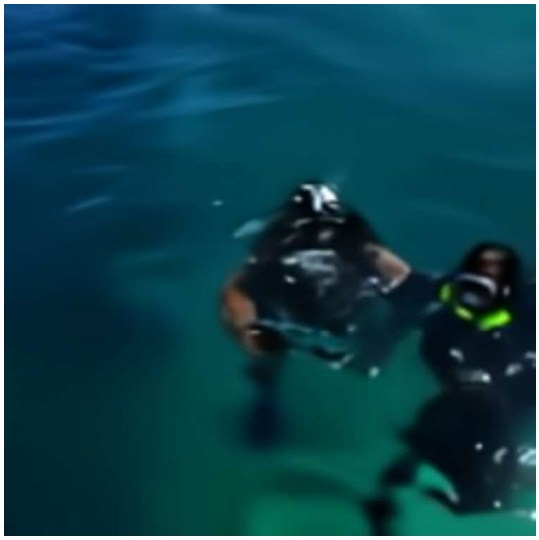


Figure 6.5: SGN $\sigma = 30$

Table 6.1: Prefiltering with SGN

stacks. The same shared weights are used for each image in each stream. Both the image stack and the feature stack are used as inputs for the alignment module.

6.2.3 Alignment

The central property that is exploited with burst denoising is that the content captured in the individual frames of the burst is very similar. In the original images, the scene content however might be shifting due to camera shake or scene dynamics. We use the pre-trained RAFT (Teed and Deng, 2020) model that is shipped with torchvision (Paszke *et al.*, 2019) to estimate the optical flow between the reference image frame and any other image frame in the burst. The estimated flow computed from the reference and secondary images is used to warp the secondary image frames and their corresponding feature maps with respect to the reference image frame and the reference feature frame respectively. The effectiveness of the RAFT-based alignment is visualized in Figure 6.6.

6.2.4 Collaborative Content-adaptive Spatial Filtering

At this point, the images and features in the bursts are all aligned with respect to the reference frame. The next step is to filter the images spatially and combine the results pixel-wise for the final result. The spatial filtering is implemented with content-dependent per-pixel kernels. Those kernels are estimated by a CNN from the aligned feature stack, i.e. collaboratively considering all feature maps at the same time. The output activations of this CNN are reshaped into 3×3 and 5×5 filter kernels for all images and all pixels. The result is two kernels of shape $[N,H,W,3,3]$ and $[N,H,W,5,5]$ with the number of images N, height H and width W. The kernels are normalized via *softmax* and applied to each image in the burst individually, effectively computing a weighted average color over the 3×3 and 5×5 neighborhood of each pixel as shown in Figure 6.7.

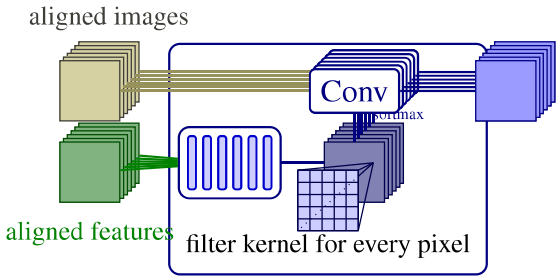


Figure 6.7: The spatial content-adaptive filter kernels for every pixel are estimated by a CNN based on all aligned features. They are applied individually to the aligned images to produce the spatially filtered burst.

6.2.5 Burst Fusion

Readers are reminded that in Section 6.2.1 the burst was split into three processing streams B, B_{10} and B_{30}, which are all processed individually in the same way so far. This means at this stage we have aligned and spatially filtered images and the corresponding aligned image features for each stream. The final step is to fuse all information from the different bursts into a single denoised image I_{pred}. This denoised result is computed as a weighted average over the spatially filtered images from all three processing streams. As indicated in Figure 6.8, we concatenate the aligned features of the streams with the spatially filtered images and process them together in a 4-layer CNN. This CNN produces the weight volume. This volume contains a weight for every pixel and color of every image. A softmax over the image dimension is applied to the weights in order to ensure that summing up the weights over this dimension yields 1 for every color channel. The result I_{pred} is finally computed as a weighted sum per pixel. This is implemented as element-wise multiplication between weight volume and spatially filtered images, followed by a sum over the burst dimension. Every channel for every input image is therefore weighted individually, which is more powerful than just mixing the existing colors of the spatially filtered images.

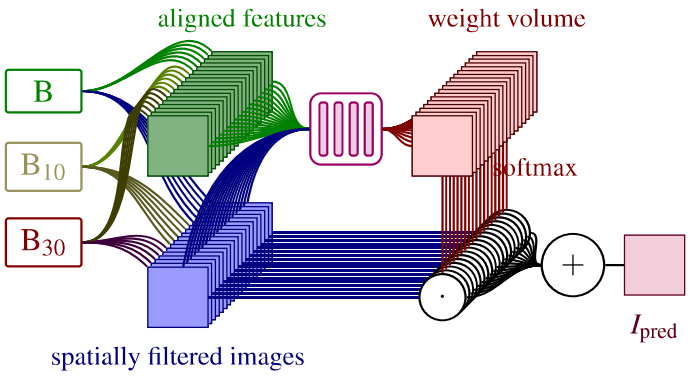


Figure 6.8: Fusion Network. The aligned features from all three bursts are concatenated with the spatially filtered images and processed by a CNN to obtain a weight volume. The weights are used to compute the denoised result as a weighted per-pixel sum over the spatially filtered images.

6.2.6 Training

Some components of the denoising pipeline like the SGN and the alignment module are pretrained. We stopped the gradients from going through the SGN networks, which effectively turns the three burst streams B, B_{10} and B_{30} into separate inputs. The RAFT network in the alignment module was frozen and used as fixed differentiable operation.

The remaining trainable weights are in the CNNs for the feature extractor, the content-adaptive cooperative spatial filter, and the burst fusion module. We train end-to-end with ADAM (Kingma and Ba, 2014) from a simple $L1$-loss $\mathcal{L} = \left\| I_{\text{pred}} - I_{\text{gt}} \right\|_1$ on the ground truth I_{gt}.

6.3 Experiments

We evaluate our method by comparing to state of the art and validate our architecture choices with an ablation study (Mallick *et al.*, 2023b). During the experimentation phase, the results were benchmarked against the contemporary state-of-the-art method.

6.3.1 Training and experimental setup

For the pre-denoising we use the SGN pre-trained with $\sigma = 10, 30$ (ZHAO, 2019). For the burst denoising training, both the SGN pre-denoising and the RAFT alignment model are frozen. We trained on the OpenImages (Krasin *et al.*, 2017) dataset and evaluated on the grayscale burst benchmark (Mildenhall *et al.*, 2018) and RGB burst benchmark, following the usual conventions (Cho *et al.*, 2021; Bhat *et al.*, 2021b; Dudhane *et al.*, 2022). The ground truth images are shifted and corrupted by adding heteroscedastic Gaussian noise (Healey and Kondepudy, 1994) with variance $\sigma_r^2 + \sigma_s^2 x$. Here x is the clean pixel value, while σ_r and σ_s denote the readout and shot noise parameters, respectively. Those noise parameters are assumed to be known both during training and testing, and are used in the feature extractor. During training, they are sampled uniformly in the log-domain from the range $\log(\sigma_r) \in [-3, -1.5]$ and $\log(\sigma_s) \in [-4, -2]$. The comparisons are evaluated with 2 different noise lvl.1 and lvl.2, corresponding to noise parameters (-2.2, -2.6) and (-1.8, -2.2) respectively. Training was done on 2 TITAN Xp GPUs and took about 96 hours to converge.

6.3.2 Results

The quantitative comparison with other methods shows that our model delivers overall state-of-the-art performance, the aforementioned benchmark in the evaluation of the model-unseen dataset.On deep introspection, we can say that due to SGN and further multiple kernel-based filtering, the model successfully recovers the image even from the heavy noise scenarios. In the future, one could add additional SGN-based denoising stages with different pre-trained noises to analyze whether further boosting of *lvl.1* and *lvl.2* would be possible. Additionally, larger filter kernels can be added to the model in order to enhance the results for higher noise scenarios.

Exemplar qualitative results on individual images are shown in Figure 6.9

	Color		Grayscale	
model	lvl. 1	lvl. 2	lvl. 1	lvl. 2
KPN (Mildenhall *et al.*, 2018)	38.38	35.96	36.47	33.93
BPN (Xia *et al.*, 2020)	40.16	37.08	38.18	35.42
MFIR-1 (Bhat *et al.*, 2021b)	40.16	37.08	39.37	36.51
MFIR-2 (Bhat *et al.*, 2021b)	42.21	39.13	39.37	36.51
BIPNET (Dudhane *et al.*, 2022)	42.28	**40.20**	**41.26**	**38.74**
Ours	**42.49**	39.18	**41.35**	36.61

Table 6.2: PSNR values of the evaluation grayscale burst dataset. Blue shows the best results, while green shows the second best results. *lvl.* indicates the level of Gaussian noise added according to evaluation convention.

6.3.3 Ablation Study

Model	*lvl.1 gray*
without SGN pre-denoising	40.81
without RAFT alignment	38.54
without content-adaptive filtering	41.27
Ours – complete pipeline	**41.35**

Table 6.3: Ablation study. Removing the individual parts of the pipeline and training the model from scratch reveals the importance of each component. Only by combining SGN-based pre-denosing, flow-based alignment and content-adaptive filtering, good performance in high noise levels can be achieved.

Since our module consists of several pretrained blocks and trainable submodules, we analyze the effectiveness of each of the components with the corresponding ablation in Table 6.3. Here, we removed individual parts of the pipeline and trained the network from scratch. Removing all SGN blocks effectively suppresses pre-denoising of the input. Although the low-noise evaluation performs comparatively well, image quality deteriorates as the noise increases due to the lack of cleaner proposals at the initial stages. Without the alignment module, the final fusion step is impaired, and we see lower performance on all noise levels. Particularly, *lvl.1* is impacted. Finally, the cooperative content-adaptive filtering adds almost equally to the reconstruction quality of all noise levels.

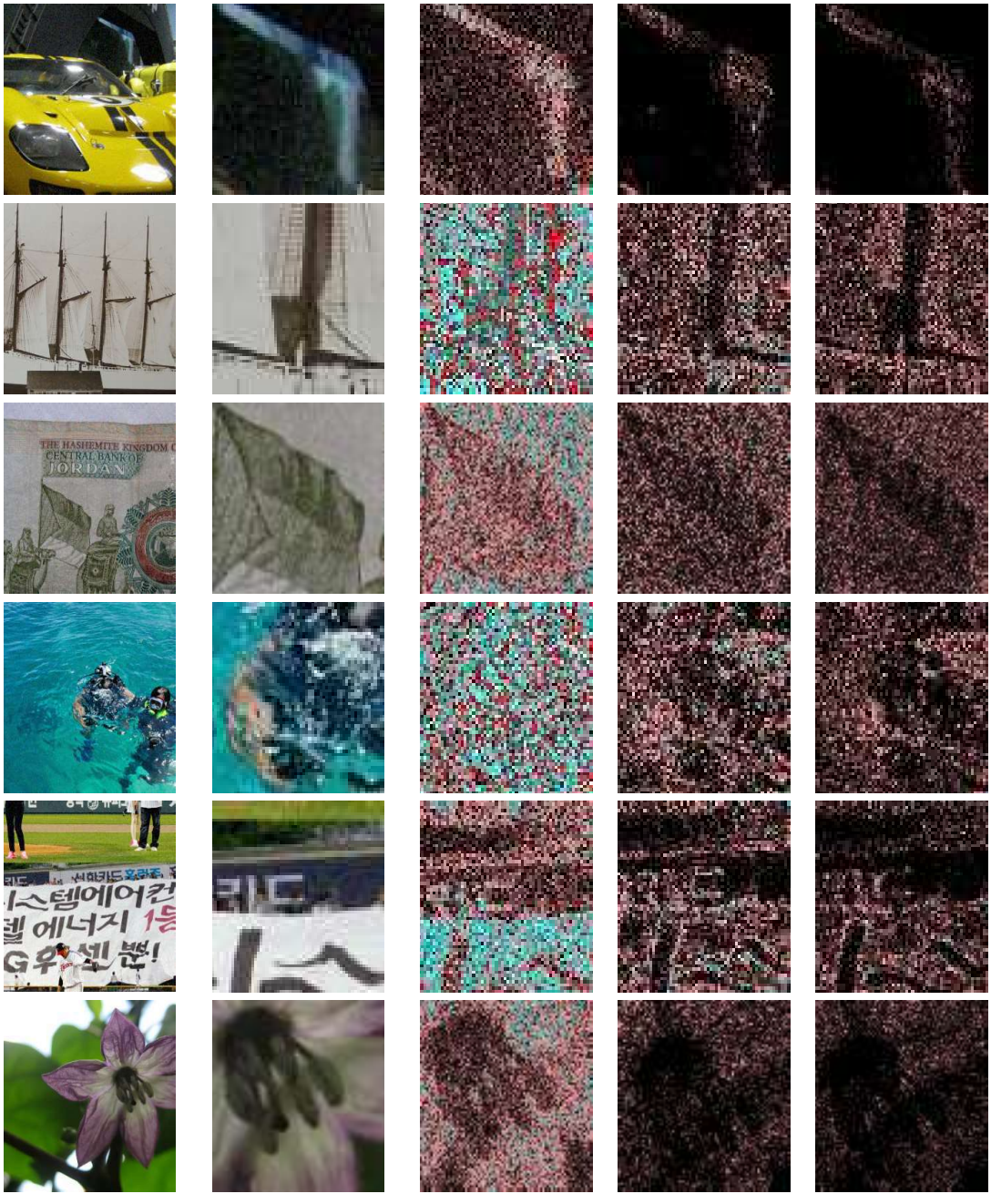
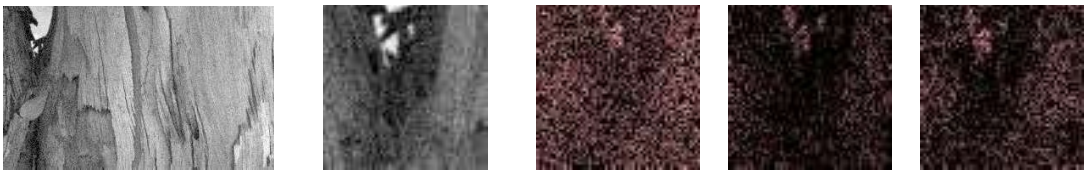


Figure 6.9: Qualitative performance of our algorithm with respect to BIPNET (Dudhane *et al.*, 2022) on the evaluation color dataset. Notice the excessive smoothing by BIPNET which removes the sharper features in comparison to the ground truth. Our network retains the details and removes the noise as well. The darker region corresponds to lesser error, which indicates better denoising. It is to be noted that the error maps have been scaled 5 times for better visual understanding.



Noisy input | Region of interest | Input error | BIPNET error | Our error

Figure 6.10: Worst performing images in the test set. The excessive noise in the input combined with strong camera motion deteriorates the denoising performance. It is to be noted that the error maps have been scaled 2 times for better visual understanding.

6.3.4 Qualitative Results

In Figure 6.9 we demonstrate the improved performance of our pipeline on a number of example images. Even for drastically different amount of noise our approach outperforms BIPNET (Dudhane *et al.*, 2022) on every image. The denoised image is significantly closer to the ground truth result, as is evident from the error maps.

A failure case is shown in Figure 6.10. In this example, apart from the high motion, the image consists of sharp features which is preserved by our network which is not detected as noise.

6.4 Conclusions

We propose a deep-burst denoising model based on optical flow guided alignment and cooperative filtering. A well-established single image denoising module generates pre-denoised burst input images for two different assumed noise levels. Alignment to the reference frame is performed using a state of the art optical flow network. Providing the original input burst and the pre-denoised stacks ensures the good performance of the optical flow alignment. Based on the aligned features and images, a set of content-adaptive spatially-varying filter kernel is predicted to smooth each input image individually. A fusion block finally combines all intermediate results to the final denoised output. In the future, one can also compare the effect of state of the art optical flow based correspondence alignment on the quality of the burst image denosing.

Our approach yields state-of-the-art results across low noise levels on the standard benchmark data sets. Higher noise scenarios working on different pre-denoised images shows a comparable benefit.

With this, we draw to a close the primary research objectives of this thesis, which focused on understanding scenes using multiple view image inputs. We have underscored the significance of correspondence alignment in image enhancement, demonstrating its crucial role in our study.

Our research has shown that correspondence alignment is not just a theoretical concept, but a practical tool that can significantly improve the quality of image enhance-

ment. By aligning multiple views accurately, we can achieve a more comprehensive and detailed understanding of the scene, leading to more accurate reconstructions.

Chapter 7

Prospects in Correspondence Localization

In multi-view stereo applications, the primary challenge of identifying correspondences between reference and source images is typically addressed through neural feature encoding or a quality metric for depth proposals. This is usually followed by an explicit search, often employing brute-force methods, to optimize the matching cost. To streamline the process of determining the optimal depth, we try to propose a simplified network for correspondence localization. A feature network is trained to encode the vicinity of a point in a highly specific manner. When this encoding is combined with the feature vector of a reference sample, the output directly indicates the location of the corresponding point. Instead of conducting an explicit search for the optimal point, it is directly predicted or at least estimated. This correspondence localization network is integrated into an MVS setting, where the predictions from multiple input views are fused using multi-head attention. An outer loop will refine the initial depth prediction. Typically, the evaluation of only three depth proposals is necessary for precise convergence. Our approach attempts to shift the algorithmic complexity of the correspondence search into training the feature encoding, thereby drastically simplifying the search process. This increases both the efficiency and memory requirements for MVS depth regression while maintaining high quality.

7.1 Learning correspondence localization for depth Regression

The common procedure for depth estimation in multi-view and stereo approaches is to compare the pixel neighborhood between a reference view pixel and possible corresponding points in secondary views, often using neural feature encoding to establish a similarity measure that is robust against changes of view or of illumination.

The best matching depth values are often estimated by searching along the epipolar line (Sormann *et al.*, 2023), by plane sweeping (Yao *et al.*, 2018b; Xue *et al.*, 2019) or the calculation and evaluation of 3D cost volumes (Dai *et al.*, 2019), to name a few.

Figure 7.1: Training samples for correspondence localization shown on a multi-view dataset.Overlaid are two different views. Given the two points q^A (blue) and the corresponding point p^B (green), samples s_i^B (orange) are spawned around p. Given the features $f(q^A)$ and $f(s_i^B)$, the network tries to predict the location of p for each sample. Most often the correct direction and distance is predicted, overdrawing the green ground truth edge.

In all of these approaches, there is a more or less dense sampling of the potential depth candidates, in some cases with an adaptive refinement of the search window to reduce the number of steps necessary to find the optimum depth.

Discrete sampling of depth values however will limit the resulting resolution, particularly in the case of 3D cost volumes due to the significant amount of memory required. In addition, dense sampling is cost-intensive as it requires performing numerous projections or neural network-based matching cost evaluations.

In contrast, we introduce a novel feature-based correspondence localization network. Inspired by the clear patterns observable in the transition of slices in cost volumes, we change the inner working of our MVS approach. Rather than just measuring how well the current depth matches across the different views, the novel task of our system is to tell where to find the correct depth or at least in which direction to find it from a single depth sample. Instead of searching for a corresponding point, we specifically encode the local 2D neighborhood into a per-pixel feature vector and train an evaluating network to directly predict the relative localization of two features. Based on this feature-based localization, an efficient MVS approach is easily set up. Given a depth proposal along a reference ray, the lookup and localization in all secondary views jointly predicts

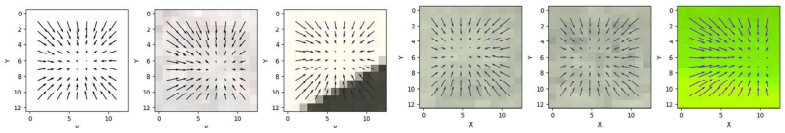


Figure 7.2: Convergence field map displayed by the direction of arrows in the pixel patches. Beginning from any point in the spatial neighborhood, the direction towards the true corresponding point in the center consistently determined. The arrow lengths in the quiver plots are scaled to avoid visual clutter.

the estimated true surface depth, using multi-head attention to carefully weight each contribution according to its reliability. This process is repeated alternating with spatial propagation in order to properly converge to the optimal depth in just a few iterations with high precision. Compared to traditional MVS networks, our approach drastically reduced the number of probed depth samples. At the same time it is more memory efficient, highly accurate and can easily deal with arbitrary many high-resolution input images.

7.2 Possible model architecture

The key to our approach is the correspondence localization network that directly determines where the correspondence point will be, rather than searching for it. In more detail, we train a variant of the U-net (Ronneberger *et al.*, 2015) extended with rotation equivariance (Jenner and Weiler, 2022; Weiler and Cesa, 2019) to aggregate information about the spatial arrangement of the larger local 2D neighborhood. The network produces a feature vector $f(j^I)$ for every pixel j and for every view I. Assume a query point q in image A with feature $f(q^A)$, the task is to find the location p of the corresponding point in image B. The exact location p is initially unknown. Instead, one might take a point s, potentially in the vicinity of p and its feature $f(s^b)$ to predict where p is located:

$$\mathcal{N}_{\text{loc}}(f(q^A), f(s^B)) = \Delta s, \text{ with } s + \Delta s = p' \approx p, \tag{7.1}$$

with N_{loc} representing the actual localization network, a two layer MLP to extract the location given the two features. Note, that the same feature $f(q^a)$ with any other feature $f(s_i^b)$ for another point s_i should produce the same prediction, while for another query, point q_i with $f(q_i^a)$ the result of $N_{loc}(f(q_i^A), f(s^B))$ should reveal a different corresponding point p_i. This should even hold for another view.

The prediction will not necessarily always be exact, but it will typically be more precise the lower the distance between s and the true correspondence p. As the process almost always predicts a point that is closer to p ($\|\mathcal{N}_{loc}(f(q^A), f(s^B)) - p\|^2 < \|s - p\|^2$), a few iterative steps will regress to the true point.

7.3 Conclusion

With the network presented in the previous section, the 2D location of corresponding feature points is efficiently and precisely determined in pixel coordinates for an image pair. In a multi-view setting, correspondence localization needs to be performed in multiple images at the same time for one reference ray and its feature. Fusing the individual disparity predictions requires transforming the localization information into the space along the reference ray. For a fixed number of views, the fusion is carried out using multi-head attention, followed by spatial convolution to quickly propagate and regularize depth prediction with neighboring rays. An outer loop iterates over randomly selected neighboring views to update and refine the per-pixel depth prediction. In conclusion, we propose a streamlined approach to model a Multi-View Stereo (MVS) learning model using inherently simpler and cost-effective submodules. Our method involves pre-training a U-net for feature localization, which is a well-established architecture known for its efficiency in dealing with image segmentation tasks.

Once the U-net is trained, we enhance its functionality by estimating disparities between reference and source images. This is achieved through an iterative process of pixelwise correspondence matching. The process of correspondence matching is crucial as it helps in identifying the similarities and differences between the reference and source images at a pixel level. This, in turn, aids in improving the depth estimation, a critical aspect of any MVS system.

It's important to note that this is an ongoing research work. While we have made significant strides, there is still much to explore and understand. However, we are optimistic that our initial findings will serve as a valuable guide for others venturing in this direction. We believe that our approach, with its focus on simplicity and cost-effectiveness, has the potential to contribute significantly to the field of MVS learning models. We look forward to sharing more of our findings as our research progresses.

Chapter 8

Final remarks

This thesis presents a comprehensive study on advanced multi-view image processing techniques. By enhancing the quality and interpretability of input images, we have made significant strides towards leveraging multiple views for more accurate scene understanding. The primary objective was to address several computer vision problem statements in this direction and propose robust and efficient learning models to solve the same. This work has been done to encapsulate the key findings, their implications, and to suggest potential directions for future research.

Key findings

We discuss the key findings, beginning with proposing an end-to-end learning module for multiple view frame interpolation with the help of CNNs as building blocks. We find out with the help of our module by directly proposing a new frame instead of learning trilinear upsampling weights proposed by the state-of-the art method. Our supervised model also achieves improved image view reconstruction. As we advance, the complexity of our target task increases, particularly in the context of reconstruction powered by multiple source images. Our next challenge is multi-view stereo reconstruction. In tackling this problem, we also address issues of adaptability and the scarcity of abundant ground truth data. We have sought to enhance the performance of an unsupervised MVS model by integrating a meta-learning module, thereby boosting the model's adaptive capabilities. Our meta-learning training proposal enhances the 3D reconstruction capabilities of the given unsupervised model. This method is model agnostic, and it can be extended to any upcoming state-of-the-art training module. In addition, we have addressed an inherent issue in this method where the resultant depth images were downsampled due to model complexity and constraints in computation power to train such expensive models. This would pave the way for our next task. For this, we propose a robust vision transformers and CNN combination module to tackle the problem of guided depth upsampling. We find out that the RGB reference image finds better geometric correspondences with the help of attention aggregation via our mergenet and hence, obtain state-of-the-art performance in depth upsampling and enhancement task. Recognizing that the backbone of any multiple-view scene understanding task is an efficient and accurate correspondence matching block, we subsequently draw inspiration from this and delved deeper

into correspondence alignment related tasks. The penultimate work deals with the classic burst image denoising problem. We fuse several pre-trained module such as initial denoising and subsequent proposal of intermediate pseudo-burst images and efficiently fuse them to obtain denoised results. We obtain state-of-the-art results in low noise regions. Ultimately, we address the correspondence localization problem. While we leave this problem open-ended for now, we provide valuable insights derived from empirical methodology.

Implications

While the primary focus of this thesis has been on fundamental multi-view vision and image processing tasks, our research lays a strong foundation for advancing scene understanding. By improving the efficiency and accuracy of core tasks like frame interpolation, multi-view stereo reconstruction, guided depth upsampling, and burst-image denoising, we have made significant strides towards enabling more sophisticated scene understanding applications. A central theme that emerged across these tasks is the critical need for lightweight and highly accurate correspondence matching methods. Such algorithms would serve as the backbone for various scene understanding applications, enabling more efficient and computationally inexpensive depth regression. Our research highlights the potential of such methods to significantly impact the field and encourages future research to address this challenge.

In conclusion, the research provided in this thesis is focused on fundamental multi-view vision and image processing tasks, which lays a valuable foundation for higher-level scene understanding, by improving the processing of the raw input. By delving into these fundamental tasks and laying the groundwork for efficient correspondence matching, our work offers a stepping stone towards achieving more comprehensive and robust scene understanding capabilities.

Future Directions

Scene understanding guided by multiple view images is a vast and complex task. We trust that this thesis has adequately covered some of its crucial aspects and will serve as a valuable resource for advancing research in this direction. In conclusion, we believe that the insights and methodologies presented in this thesis will inspire and guide future research in this field. We look forward to seeing how our work will contribute to the ongoing evolution of scene understanding techniques with the help of multiple view fusion models. We are excited about the potential impact of our research and eager to see how it will shape the future of this fascinating field. We hope that our work will serve as a stepping stone for future researchers, providing them with the tools and insights they need to push the boundaries of what is currently possible. On a final remark, while we have achieved significant milestones in our research, the journey is far from over. We are excited about the possibilities that lie ahead and look forward to continuing our work in

this fascinating, yet challenging field of learning multiple view features for robust scene understanding.

Acknowledgements

This research would not have been possible without the immense support of my supervisor, Prof. Dr. Hendrik PA Lensch. Special thanks go to my colleagues at the department of Computer Graphics at the University of Tübingen for their continuous support and the positive feedbacks. Additional thanks go to the staff, coordinators and colleagues of the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for their constant words of support and initiating events for exchange of ideas among the researchers of the University and the Institute for encouraging interdisciplinary networking. Furthermore, I am grateful to my co-supervisors Dr. Jörg Stückler and Prof. Dr. Andreas Geiger for their valuable feedback and collaboration. This journey would have been far from complete without the unwavering support of my family and friends, who have been a constant pillar of strength and encouragement.
Works achieved on this thesis has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC number 2064/1 - project number 390727645 and SFB 1233 - project number 276693517. It was supported by the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A and Cyber Valley.
The author also wishes to dedicate this work as a heartfelt tribute to the collective efforts of the open source research community. Their invaluable contributions regarding code-sharing for meticulous validation, and comprehensive benchmarking of the proposed computer vision learning models have been instrumental. Without their tireless efforts, the realization of this thesis would have remained a distant dream.

Abbreviations

ANN	Artificial neural network
BN	Batch Normalization
API	Application Programming Interface
CNN	Convolutional Neural Network
CPU	Central processing unit
CUDA	Compute Unified Device Architecture
DGF	Deep Guided Filter
DKN	Deformable Kernel Network
DTU	Technical University of Denmark MVS dataset
DVF	Deep Voxel Flow
ETH3D	Swiss Federal Institute of Technology Zurich SLAM and MVS benchmark dataset
FoV	Field of View
FT	Fine-tuned
GF	Guided Filter
GPU	Graphics processing unit
JIIF	Joint Implicit Image Function
MAML	Model Agnostic Meta Learning
MLP	Multi-layer perceptron
MRF	Markov Random Field
MVS	Multiple view stereo
MP	Mega Pixel
NLP	Natural Language Processing
NYUv2	New York University depth dataset version 2
PSNR	Peak signal-to-noise ratio
PT	Pre-trained
px	Pixel
RAFT	Recurrent All Pairs Field Transforms for Optical Flow
ReLU	Rectified linear unit
RGB	Red Green Blue image channels
RMSE	Root mean square error
Sfm	Structure from motion
SGN	self-guided filtering network
SLAM	Simultaneous Localization and Mapping

SNR	Signal-to-noise ratio
SR	Super Resolution
SSIM	Structural similarity index measure
TDNN	Time delay neural network
VDP2	Visible Difference Predictor 2

Bibliography

A. Dosovitskiy, P. Fischer, E. I. P. H. C. H. V. G. P. v. d. S. D. C. and Brox., T. (2015). Flownet: Learning optical flow with convolutional networks. *In ICCV*, page pages 2758–2766.

A. Dosovitskiy, J. T. S. and Brox, T. (2015). Learning to generate chairs with convolutional neural networks. In *In IEEE Conference on Computer Vision and Pattern Recognition*, page 1538–1546.

Aanæs, H., Jensen, R. R., Vogiatzis, G., Tola, E., and Dahl, A. B. (2016). Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, pages 1–16.

Aharon, M., Elad, M., and Bruckstein, A. (2006). K-svd: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, **54**(11), 4311–4322.

Alhashim, I. and Torr, P. (2018). High quality monocular depth estimation via transfer learning. *arXiv preprint arXiv:1812.11941*.

Amari, S. (1967). A theory of adaptive pattern classifier. *IEEE Transactions on Electronic Computers*, **16**(3), 279–307.

Ariav, I. and Cohen, I. (2022). Depth Map Super-Resolution via Cascaded Transformers Guidance. In *Frontiers in Signal Processing*.

Arnold, S. M., Mahajan, P., Datta, D., and Bunner, I. (2019). learn2learn.

Barnes, C., Shechtman, E., Finkelstein, A., and Goldman, D. B. (2009). PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, **28**(3).

Barron, J. T. and Poole, B. (2016). The fast bilateral solver. *ECCV*.

Bay, H., Ess, A., Tuytelaars, T., and Gool, L. V. (2008). Surf: Speeded up robust features. *Computer Vision and Image Understanding*, **110**(3), 346–359.

Beltagy, I., Peters, M. E., and Cohan, A. (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Benchmark, G. T. S. R. (2023). Gtsrb results.

Bhat, G., Danelljan, M., Van Gool, L., and Timofte, R. (2021a). Deep burst super-resolution.

Bhat, G., Danelljan, M., Yu, F., Gool, L. V., and Timofte, R. (2021b). Deep reparametrization of multi-frame super-resolution and denoising.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners.

Brox, T., Bruhn, A., Papenberg, N., and Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. In *European conference on computer vision*, pages 25–36. Springer.

Bruhn, A., Weickert, J., and Schnörr, C. (2005). Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International journal of computer vision*, **61**(3), 211–231.

Butler, D. J., Wulff, J., Stanley, G. B., and Black, M. J. (2012). A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag.

Campbell, N. D. F., Vogiatzis, G., Hernández, C., and Cipolla, R. (2008). Using multiple hypotheses to improve depth-maps for multi-view stereo. In D. Forsyth, P. Torr, and A. Zisserman, editors, *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 766–779.

Camplani, M., del Blanco, C. R., Salgado, L., Jaureguizar, F., and García, N. (2014). Multi-sensor background subtraction by fusing multiple region-based probabilistic classifiers. *Pattern Recogn. Lett.*, **50**(C), 23–33.

Cao, Y., Xu, J., Lin, S., Wei, F., and Hu, H. (2019). Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 1971–1980.

Carion, N., Massa, F., Brostow, G. J., Mirowski, P., and Zisserman, A. (2021). Vision transformers for dense prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6308–6316.

Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., and Gao, W. (2020). Pre-trained image processing transformer.

Chen, Y., Zhang, Y., Liu, Y., Li, Y., Chen, K., Wang, Y., Wang, Y., Yan, S., and Lin, X. (2022). Datasets and benchmarks for learning multiple view reconstruction. *NeurIPS Proceedings*.

Child, R., Gray, S., Radford, A., and Sutskever, I. (2019). Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Cho, W., Son, S., and Kim, D.-S. (2021). Weighted multi-kernel prediction network for burst image super-resolution. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 404–413.

Choy, C. B., Gwak, J., Savarese, S., and Chandraker, M. (2016). Universal correspondence network.

Ciresan, D. C., Meier, U., Masci, J., Gambardella, L. M., and Schmidhuber, J. (2011). Flexible, high performance convolutional neural networks for image classification. In *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Two*, pages 1237–1242. IJCAI/AAAI.

Cireşan, D. C., Meier, U., Gambardella, L. M., and Schmidhuber, J. (2010). Deep, big, simple neural nets for handwritten digit recognition. *Neural Computation*, **22**(12), 3207–3220.

Cireşan, D. C., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3642–3649.

D. Mahajan, F.-C. Huang, W. M. R. R. and Belhumeur, P. (2009). Moving gradients: A path-based method for plausible image interpolation. *TOG, 28(3)*.

Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. (2007a). Image denoising by sparse 3d transform-domain collaborative filtering.

Dabov, K., Foi, A., and Egiazarian, K. (2007b). Video denoising by sparse 3d transform-domain collaborative filtering. In *2007 15th European Signal Processing Conference*, pages 145–149.

Dai, Y., Zhu, Z., Rao, Z., and Li, B. (2019). MVS^2: Deep unsupervised multi-view stereo with multi-view symmetry. In *Proceedings of the International Conference on 3D Vision (3DV)*.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, **1**, 886–893.

de Lutio, R., Becker, A., D'Aronco, S., Russo, S., Wegner, J. D., and Schindler, K. (2022). Learning Graph Regularisation for Guided Super-Resolution. page 10.

Deudon, M., Kalaitzis, A., Goytom, I., Arefin, M. R., Lin, Z., Sankaran, K., Michalski, V., Kahou, S. E., Cornebise, J., and Bengio, Y. (2020). Highres-net: Recursive fusion for multi-frame super-resolution of satellite imagery.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2019). BERT: pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Didyk, P., Sitthi-Amorn, P., Freeman, W., Durand, F., and Matusik, W. (2013). Joint view expansion and filtering for automultiscopic 3d displays. *ACM Transactions on Graphics (TOG)*, **32**(6), 221.

Diebel, J. and Thrun, S. (2006). An application of markov random fields to range sensing. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural Information Processing Systems*, volume 18. MIT Press.

Donné, S. and Geiger, A. (2019). Learning non-volumetric depth fusion using successive reprojections. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.

Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., and Brox, T. (2015). Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale.

Dudhane, A., Zamir, S. W., Khan, S., Khan, F. S., and Yang, M.-H. (2022). Burst image restoration and enhancement.

Eigen, D., Puhrsch, C., and Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, pages 2366–2374.

Fabbri, R., Kimia, B. B., and Giblin, P. J. (2012). Camera pose estimation using first-order curve differential geometry. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part IV*, ECCV'12, page 231–244, Berlin, Heidelberg. Springer-Verlag.

Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1126–1135.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, **36**(4), 193–202.

Furukawa, Y. and Ponce, J. (2010). Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(8), 1362–1376.

Godard, C., Lecun, A., Boulch, Z., Choblet, G., and Vallet, M. (2017a). Unsupervised monocular depth estimation with left-right consistency. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **39**, 1699–1711.

Godard, C., Mac Aodha, O., and Brostow, G. J. (2017b). Unsupervised monocular depth estimation with left-right consistency. In *CVPR*.

Godard, C., Matzen, K., and Uyttendaele, M. (2018). Deep burst denoising. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Gu, S., Li, Y., Gool, L. V., and Timofte, R. (2019). Self-guided network for fast image denoising. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2511–2520.

G¨uney, F. and Geiger, A. (2016). Deep discrete flow. *Asian Conference on Computer Vision*, **10114**, 207–224.

Ham, B., Cho, M., and Ponce, J. (2018). Robust guided image filtering using nonconvex potentials. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **40**(1), 192–207.

Hampshire, J. B. and Waibel, A. (1990). Connectionist architectures for multi-speaker phoneme recognition. In *Advances in Neural Information Processing Systems*. Morgan Kaufmann.

Han, Z., Zhang, C., Fu, H., and Zhou, J. T. (2021). Trusted multi-view classification. In *International Conference on Learning Representations*.

Hartmann, W., Galliani, S., Havlena, M., Van Gool, L., and Schindler, K. (2017). Learned multi-patch similarity. In *The IEEE International Conference on Computer Vision (ICCV)*.

Hasinoff, S. W., Sharlet, D., Geiss, R., Adams, A., Barron, J. T., Kainz, F., Chen, J., and Levoy, M. (2016). Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Trans. Graph.*, **35**(6).

He, K., Sun, J., and Tang, X. (2013). Guided image filtering. *IEEE Trans. Pattern Anal. Mach. Intell.*, **35**(6), 1397–1409.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778.

He, L., Zhu, H., Li, F., Bai, H., Cong, R., Zhang, C., Lin, C., Liu, M., and Zhao, Y. (2021). Towards fast and accurate real-world depth super-resolution: Benchmark dataset and baseline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9229–9238.

Healey, G. and Kondepudy, R. (1994). Radiometric ccd camera calibration and noise estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(3), 267–276.

Hirschmuller, H. and Scharstein, D. (2007). Evaluation of cost functions for stereo matching. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.

Hu, J., Shen, L., Albanie, S., Sun, G., and Vedaldi, A. (2018a). Gather-excite: Exploiting feature context in convolutional neural networks.

Hu, J., Shen, L., and Sun, G. (2018b). Squeeze-and-excitation networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141.

Huang, P.-H., Matzen, K., Kopf, J., Ahuja, N., and Huang, J.-B. (2018). DeepMVS: Learning multi-view stereopsis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huang, S., Wang, Q., Zhang, S., Yan, S., and He, X. (2019). Dynamic context correspondence network for semantic alignment. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2010–2019.

Huang, Z., Shi, X., Zhang, C., Wang, Q., Cheung, K. C., Qin, H., Dai, J., and Li, H. (2022). Flowformer: A transformer architecture for optical flow. *arXiv preprint arXiv:2203.16194*.

Hui, T.-W., Loy, C. C., and Tang, X. (2016). Depth map super-resolution by deep multi-scale guidance. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision – ECCV 2016*, pages 353–369, Cham. Springer International Publishing.

Häne, C., Heng, L., Lee, G. H., Sizov, A., and Pollefeys, M. (2014). Real-time direct dense matching on fisheye images using plane-sweeping stereo. In *2014 2nd International Conference on 3D Vision*, volume 1, pages 57–64.

Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., and Brox, T. (2017). Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2462–2470.

Im, S., Jeon, H.-G., Lin, S., and Kweon, I. S. (2018). DPSNet: End-to-end deep plane sweep stereo. In *Proceedings of International Conference on Learning Representations (ICLR)*.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, **abs/1502.03167**.

Islam, M. J., Sakib Enan, S., Luo, P., and Sattar, J. (2020). Underwater image super-resolution using deep residual multipliers. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 900–906.

J. Flynn, I. Neulander, J. P. and Snavely, N. (2016). Deepstereo: Learning to predict new views from the world's imagery. *In IEEE Conference on Computer Vision and Pattern Recognition*, page 5515–5524.

J Teuwena, N. M. (2023). Convolutional neural networks.

J. Yang, S. E. Reed, M. Y. and Lee, H. (2015). Weaklysupervised disentangling with recurrent transformations for 3d view synthesis. In *In NIPS*, page 1099–1107.

Jenner, E. and Weiler, M. (2022). Steerable partial differential operators for equivariant neural networks. In *International Conference on Learning Representations*.

Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., and Aanæs, H. (2014). Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413. IEEE.

Ji, M., Gall, J., Zheng, H., Liu, Y., and Fang, L. (2017a). Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *IEEE International Conference on Computer Vision (ICCV), 2017*. IEEE Computer Society.

Ji, M., Gall, J., Zheng, H., Liu, Y., and Fang, L. (2017b). SurfaceNet: An end-to-end 3D neural network for multiview stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2307–2315.

Jiang, H., Sun, D., Jampani, V., Yang, M., Learned-Miller, E. G., and Kautz, J. (2017). Super slomo: High quality estimation of multiple intermediate frames for video interpolation. *CoRR*, **abs/1712.00080**.

Jiang, W., Trulls, E., Hosang, J., Tagliasacchi, A., and Yi, K. M. (2021). Cotr: Correspondence transformer for matching across images.

Kan, H., Zhang, S., Shan, Y., and Zhang, X. (2016). Multi-view deep learning for 3d scene reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kawulok, M., Benecki, P., Piechaczek, S., Hrynczenko, K., Kostrzewa, D., and Nalepa, J. (2020). Deep learning for multiple-image super-resolution. *IEEE Geoscience and Remote Sensing Letters*, **17**(6), 1062–1066.

Khot, T., Agrawal, S., Tulsiani, S., Mertz, C., Lucey, S., and Hebert, M. (2019). Learning unsupervised multi-view stereopsis via robust photometric consistency. *CoRR*, **abs/1905.02706**.

Kiechle, M., Hawe, S., and Kleinsteuber, M. (2013). A joint intensity and depth co-sparse analysis model for depth map super-resolution. In *2013 IEEE International Conference on Computer Vision*, pages 1545–1552.

Kim, B., Ponce, J., and Ham, B. (2020). Deformable kernel networks for joint image filtering. *International Journal of Computer Vision*, **129**(2), 579–600.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Knapitsch, A., Park, J., Zhou, Q.-Y., and Koltun, V. (2017). Tanks and temples: benchmarking large-scale scene reconstruction. *ACM Trans. Graph.*, **36**, 78:1–78:13.

Kopf, J., Cohen, M. F., Lischinski, D., and Uyttendaele, M. (2007). Joint bilateral upsampling. *ACM Trans. Graph.*, **26**(3), 96–es.

Krasin, I., Duerig, T., Alldrin, N., Ferrari, V., Abu-El-Haija, S., Kuznetsova, A., Rom, H., Uijlings, J., Popov, S., Kamali, S., Malloci, M., Pont-Tuset, J., Veit, A., Belongie, S., Gomes, V., Gupta, A., Sun, C., Chechik, G., Cai, D., Feng, Z., Narayanan, D., and Murphy, K. (2017). Openimages: A public dataset for large-scale multi-label and multi-class image classification. *Dataset available from https://storage.googleapis.com/openimages/web/index.html*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 25 (NIPS 2012)*, pages 1097–1105.

Kwon, H., Tai, Y.-W., and Lin, S. (2015). Data-driven depth map refinement via multi-scale sparse representation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 159–167.

Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., and Navab, N. (2016). Deeper depth prediction with fully convolutional residual networks. In *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2394–2403.

LeCun, Y. and Bengio, Y. (1995). Convolutional networks for images, speech, and time-series. In M. A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 276–278. The MIT Press, second edition.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, **1**(4), 541–551.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11), 2278–2324.

Li, Y., Xue, T., Sun, L., and Liu, J. (2012). Joint example-based depth map super-resolution. In *2012 IEEE International Conference on Multimedia and Expo*, pages 152–157.

Li, Y., Huang, J.-B., Ahuja, N., and Yang, M.-H. (2016). Deep joint image filtering. In B. Leibe, J. Matas, N. Sebe, and M. Welling, editors, *Computer Vision – ECCV 2016*, pages 154–169, Cham. Springer International Publishing.

Li, Y., Huang, J.-B., Ahuja, N., and Yang, M.-H. (2019). Joint image filtering with deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, **41**(8), 1909–1923.

Liba, O., Murthy, K., Tsai, Y.-T., Brooks, T., Xue, T., Karnad, N., He, Q., Barron, J. T., Sharlet, D., Geiss, R., Hasinoff, S. W., Pritch, Y., and Levoy, M. (2019). Handheld mobile photography in very low light. *ACM Transactions on Graphics*, **38**(6), 1–16.

Linnainmaa, S. (1970). *The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors*. Master's thesis, University of Helsinki.

Liu, Z., Yeh, R. A., Tang, X., Liu, Y., and Agarwala, A. (2017). Video frame synthesis using deep voxel flow. *CoRR*, **abs/1702.02463**.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the 7th IEEE International Conference on Computer Vision*, **2**, 1150–1157.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, **60**, 91–110.

Lu, S., Ren, X., and Liu, F. (2014). Depth enhancement via low-rank matrix completion. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3390–3397.

Lutio, R. D., D'aronco, S., Wegner, J. D., and Schindler, K. (2019). Guided Super-Resolution As Pixel-to-Pixel Transformation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8828–8836, Seoul, Korea (South). IEEE.

M. Werlberger, T. Pock, M. U. and Bischof, H. (2011). Optical flow guided tv-l 1 video interpolation and restoration. *In Energy Minimization Methods in Computer Vision and Pattern Recognition*, **6819**, 273–286.

Maggioni, M., Boracchi, G., Foi, A., and Egiazarian, K. (2012). Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms. *IEEE Transactions on Image Processing*, **21**(9), 3952–3966.

Mallick, A., Stückler, J., and Lensch, H. (2020a). Learning to adapt multi-view stereo by self-supervision.

Mallick, A., Stückler, J., and Lensch, H. (2020b). Learning to adapt multi-view stereo by self-supervision. In *Proceedings of the British Machine Vision Conference (BMVC)*. preprint https://arxiv.org/abs/2009.13278.

Mallick, A., Engelhardt, A., Braun, R., and Lensch, H. P. A. (2022). Local Attention Guided Joint Depth Upsampling. In J. Bender, M. Botsch, and D. A. Keim, editors, *Vision, Modeling, and Visualization*. The Eurographics Association.

Mallick, A., Braun, R., and Lensch, H. P. (2023a). Candid: Correspondence alignment for deep-burst image denoising. In *2023 20th Conference on Robots and Vision (CRV)*, pages 241–247.

Mallick, A., Braun, R., and Lensch, H. P. (2023b). Candid: Correspondence alignment for deep-burst image denoising. In *2023 20th Conference on Robots and Vision (CRV)*, pages 241–247.

Mao, X.-J., Shen, C., and Yang, Y.-B. (2016). Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections.

Merrell, P., Akbarzadeh, A., Wang, L., Frahm, J.-M., Yang, R., and Nister, D. (2007). Real-time visibility-based fusion of depth maps. In *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Meyer, S., Wang, O., Zimmer, H., Grosse, M., and Sorkine-Hornung, A. (2015). Phase-based frame interpolation for video. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 1410–1418. IEEE.

Michael Bleyer, C. R. and Rother, C. (2011). Patchmatch stereo - stereo matching with slanted support windows. In *Proceedings of the British Machine Vision Conference*, pages 14.1–14.11. BMVA Press. http://dx.doi.org/10.5244/C.25.14.

Mikolajczyk, K. and Schmid, C. (2005). A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(10), 1615–1630.

Mildenhall, B., Barron, J. T., Chen, J., Sharlet, D., Ng, R., and Carroll, R. (2018). Burst denoising with kernel prediction networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Mohamed, A., Black, M., and Taylor, G. W. (2019). Wasserstein distances for stereo disparity estimation. In *Proceedings IEEE International Conference on Computer Vision (ICCV)*, pages 3752–3761.

Molini, A. B., Valsesia, D., Fracastoro, G., and Magli, E. (2020). DeepSUM: Deep neural network for super-resolution of unregistered multitemporal images. *IEEE Transactions on Geoscience and Remote Sensing*, **58**(5), 3644–3656.

Niebles, J. C., Wang, H., and Li, F.-F. (2006). Unsupervised learning of human action categories using spatial-temporal words. In *Proceedings of the British Machine Vision Conference (BMVC)*.

Niklaus, S., Mai, L., and Liu, F. (2017). Video frame interpolation via adaptive convolution. *CoRR*, **abs/1703.07514**.

Oh, K.-S. and Jung, K.-H. (2004). Gpu implementation of neural networks. In *Proceedings of the 2004 International Joint Conference on Neural Networks (IJCNN'04)*, volume 1, pages 449–454. IEEE.

Park, J., Kim, H., Tai, Y.-W., Brown, M. S., and Kweon, I.-S. (2011). High quality depth map upsampling for 3d-tof cameras. *2011 International Conference on Computer Vision*, pages 1623–1630.

Paschalidou, D., Ulusoy, O., Schmitt, C., Van Gool, L., and Geiger, A. (2018). RayNet: Learning volumetric 3D reconstruction with ray potentials. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Rae, J. and Razavi, A. (2020). Do transformers need deep long-range memory? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

Ranjan, A., Jampani, V., Balles, L., Kim, K., Sun, D., Wulff, J., and Black, M. J. (2019). Competitive collaboration: Joint unsupervised learning of depth, camera motion, optical flow and motion segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12240–12249.

Riegler, G., Ferstl, D., Rüther, M., and Bischof, H. (2016). A deep primal-dual network for guided depth super-resolution. *CoRR*, **abs/1607.08569**.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation.

S. Baker, D. Scharstein, J. P. L. S. R. M. J. B. and Szeliski, R. (2011). A database and evaluation methodology for optical flow. *International Journal of Computer Vision*, page 1– 31.

S. Meyer, O. Wang, H. Z. M. G. and SorkineHornung, A. (2015). Phase-based frame interpolation for video. *In CVPR*.

Scharstein, D. and Pal, C. (2007). Learning conditional random fields for stereo. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8.

Schmidhuber, J. (2015). Deep learning. *Neural Networks*, **61**, 85–117.

Schmidhuber, J. (2017). History of computer vision contests won by deep cnns on gpu.

Schmidt, T., Newcombe, R., and Fox, D. (2017). Self-supervised visual descriptor learning for dense correspondence. *IEEE Robotics and Automation Letters*, **2**(2), 420–427.

Schmidt, U. and Roth, S. (2014). Shrinkage fields for effective image restoration. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2774–2781.

Schönberger, J. L. and Frahm, J.-M. (2016). Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Schönberger, J. L., Zheng, E., Pollefeys, M., and Frahm, J.-M. (2016). Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*.

Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer.

Sirmacek, B. and Unsalan, C. (2009). Urban-area and building detection using sift keypoints and graph theory. *IEEE Transactions on Geoscience and Remote Sensing*, **47**(4), 1156–1167.

Smolyanskiy, K., Kamenev, A., Kovalenko, I., and Lepetit, V. (2018). On the importance of scene understanding for autonomous vehicles. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

Song, C. H., Han, H. J., and Avrithis, Y. (2022). All the attention you need: Global-local, spatial-channel attention for image retrieval. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2754–2763.

Song, S., Lichtenberg, S. P., and Xiao, J. (2015). Sun RGB-D: A rgb-d scene understanding benchmark suite. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 567–576.

Sormann, C., Santellani, E., Rossi, M., Kuhn, A., and Fraundorfer, F. (2023). Dels-mvs: Deep epipolar line search for multi-view stereo. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3086–3095.

Steinke, F., Schölkopf, B., and Blanz, V. (2007). Learning dense 3d correspondence. In *Advances in Neural Information Processing Systems 19*, pages 1313–1320, Cambridge, MA, USA. Max-Planck-Gesellschaft, MIT Press.

Steinkraus, D., Simard, P., and Buck, I. (2005). Using gpus for machine learning algorithms. *Proceedings of the Eighth International Conference on Document Analysis and Recognition (ICDAR 2005)*, pages 824–828.

Su, H., Jampani, V., Sun, D., Gallo, O., Learned-Miller, E., and Kautz, J. (2019). Pixel-adaptive convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sun, B., Ye, X., Li, B., Li, H., Wang, Z., and Xu, R. (2021). Learning scene structure guidance via cross-task knowledge transfer for single depth super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7792–7801.

T. D. Kulkarni, W. F. Whitney, P. K. and Tenenbaum, J. B. (2015). Deep convolutional inverse graphics network. In *In NIPS*, page 2539–2547.

Tang, J., Chen, X., and Zeng, G. (2021). Joint implicit image function for guided depth super-resolution. *Proceedings of the 29th ACM International Conference on Multimedia*.

Teed, Z. and Deng, J. (2020). Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer.

Teney, D. and Hebert, M. (2016). Learning to extract motion from videos in convolutional neural networks.

Tico, M. (2008a). Multi-frame image denoising and stabilization. In *2008 16th European Signal Processing Conference*, pages 1–4.

Tico, M. (2008b). Multi-frame image denoising and stabilization. In *2008 16th European Signal Processing Conference*, pages 1–4.

Tola, E., Strecha, C., and Fua, P. (2012). Efficient large-scale multi-view stereo for ultra high-resolution image sets. *Mach. Vision Appl.*, **23**(5), 903–920.

Tomasi, C. and Manduchi, R. (1998). Bilateral filtering for gray and color images. In *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pages 839–846.

Tonioni, A., Rahnama, O., Joy, T., Di Stefano, L., Thalaiyasingam, A., and Torr, P. (2019). Learning to adapt for stereo. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ulusoy, A. O., Geiger, A., and Black, M. J. (2015). Towards probabilistic volumetric reconstruction using ray potentials. In *2015 International Conference on 3D Vision*, pages 10–18.

Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., and Brox, T. (2017). DeMoN: Depth and motion network for learning monocular stereo. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Valsesia, D. and Magli, E. (2022). Permutation invariance and uncertainty in multitemporal image super-resolution. *IEEE Transactions on Geoscience and Remote Sensing*, **60**, 1–12.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Waibel, A. (1987). Phoneme recognition using time-delay neural networks. In *Meeting of the Institute of Electrical, Information and Communication Engineers (IEICE)*.

Waibel, A., Hanazawa, T., Hinton, G. E., Shikano, K., and Lang, K. J. (1989). Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, **37**(3), 328–339.

Wang, Y., Sundararaman, S., Li, M. Y., and Fei-Fei, L. (2019). Unos: Unified unsupervised optical-flow and stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10463–10471.

Wang, Z., Simoncelli, E. P., and Bovik, A. C. (2003). Multi-scale structural similarity for image quality assessment.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *Trans. Img. Proc.*, **13**(4), 600–612.

Weiler, M. and Cesa, G. (2019). General E(2)-Equivariant Steerable CNNs. In *Conference on Neural Information Processing Systems (NeurIPS)*.

Wieschollek, P., Hirsch, M., Scholkopf, B., and Lensch, H. P. A. (2017). Learning blind motion deblurring. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Wronski, B., Garcia-Dorado, I., Ernst, M., Kelly, D., Krainin, M., Liang, C.-K., Levoy, M., and Milanfar, P. (2019). Handheld multi-frame super-resolution. *ACM Trans. Graph.*, **38**(4).

Wu, H., Zheng, S., Zhang, J., and Huang, K. (2018). Fast end-to-end trainable guided filter. In *CVPR*.

Xia, Z., Perazzi, F., Gharbi, M., Sunkavalli, K., and Chakrabarti, A. (2020). Basis prediction networks for effective burst denoising with large kernels. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11841–11850.

Xie, J., Xu, L., and Chen, E. (2012). Image denoising and inpainting with deep neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, page 341–349, Red Hook, NY, USA. Curran Associates Inc.

Xing, X., Cai, Y., Wang, Y., Lu, T., Yang, Y., and Wen, D. (2021). Dynamic Guided Network for Monocular Depth Estimation. *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5459–5465.

Xue, Y., Chen, J., Wan, W., Huang, Y., Yu, C., Li, T., and Bao, J. (2019). Mvscrf: Learning multi-view stereo with conditional random fields. In *The IEEE International Conference on Computer Vision (ICCV)*.

Yamaguchi, K., Sakamoto, K., Akabane, T., and Fujimoto, Y. (1990). A neural network for speaker-independent isolated word recognition. In *First International Conference on Spoken Language Processing (ICSLP 90)*.

Yang, F., Yang, H., Fu, J., Lu, H., and Guo, B. (2020). Learning texture transformer network for image super-resolution. In *CVPR*.

Yang, J., Wright, J., Huang, T. S., and Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, **19**(11), 2861–2873.

Yang, J., Ye, X., Li, K., Hou, C., and Wang, Y. (2014). Color-guided depth recovery from rgb-d data using an adaptive autoregressive model. *IEEE Transactions on Image Processing*, **23**(8), 3443–3458.

Yang, Y., Cao, Q., Zhang, J., and Tao, D. (2022). CODON: On Orchestrating Cross-Domain Attentions for Depth Super-Resolution. *International Journal of Computer Vision*, **130**(2), 267–284.

Yao, Y., Luo, Z., Gao, S., Zhang, M., Li, X., Tai, C., and Quan, L. (2018a). Mvsnet: Depth inference for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*.

Yao, Y., Luo, Z., Li, S., Fang, T., and Quan, L. (2018b). Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783.

Yao, Y., Luo, Z., Li, S., Zhang, J., Ren, Y., Zhou, L., Fang, T., and Quan, L. (2020). BlendedMVS: A large-scale dataset for generalized multi-view stereo networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yi, K. M., Trulls, E., Ono, Y., Lepetit, V., Salzmann, M., and Fua, P. (2018). Learning to find good correspondences. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2666–2674.

Yu, Z. and Gao, S. (2020). Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement. In *CVPR*.

Z. Yu, H. Li, Z. W. Z. H. and Chen, C. W. (2013). Multi-level video frame interpolation: Exploiting the interaction among different levels. *IEEE Trans. Circuits Syst. Video Techn*, page 1235–1248.

Zhang, W. (1988). Shift-invariant pattern recognition neural network and its optical architecture. In *Proceedings of the Annual Conference of the Japan Society of Applied Physics*.

Zhang, W. (1991). Image processing of human corneal endothelium based on a learning network. *Applied Optics*, **30**(29), 4211–4217.

Zhang, W. (1994). Computerized detection of clustered microcalcifications in digital mammograms using a shift-invariant artificial neural network. *Medical Physics*, **21**(4), 517–524.

Zhang, W., Itoh, K., Tanida, J., and Ichioka, Y. (1990). Parallel distributed processing model with local space-invariant interconnections and its optical architecture. , **29**(32), 4790–4797.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, **22**(11), 1330–1334.

Zhang, Z. (2019). Image local attention: a better pytorch implementation.

ZHAO, Y. (2019). PyTorch implementation of Self Guided Network (ICCV).

Zhao, Z., Zhang, J., Xu, S., Lin, Z., and Pfister, H. (2022). Discrete cosine transform network for guided depth map super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5697–5707.

Zhou, Q.-Y., Park, J., and Koltun, V. (2018). Open3D: A modern library for 3D data processing. *arXiv:1801.09847*.