Noname manuscript No.
(will be inserted by the editor)

Learning Inverse Depth Regression for Pixelwise Visibility-Aware Multi-View Stereo Networks

Qingshan Xu[1] · **Wanjuan Su**[1] · **Yuhang Qi**[1] · **Wenbing Tao**[1] · **Marc Pollefeys**[2,3]

Received: date / Accepted: date

Abstract Recently, learning-based multi-view stereo methods have achieved promising results. However, most of them overlook the visibility difference among different views, which leads to an indiscriminate multi-view similarity definition and greatly limits their performance on datasets with strong viewpoint variations. To deal with this problem, a pixelwise visibility-aware multi-view stereo network is proposed for robust dense 3D reconstruction. We present a pixelwise visibility estimation network to learn the visibility information for different neighboring images before computing the multi-view similarity, and then construct an adaptive weighted cost volume with the visibility information. Unlike previous methods that treat multi-view depth inference as a depth regression problem or an inverse depth classification problem, we recast multi-view depth inference as an inverse depth regression task. This allows our network to achieve sub-pixel estimation and be applicable to large-scale scenes. To achieve scalable high-resolution depth map estimation, we construct cost volumes by group-wise correlation and design an ordinal-based uncertainty estimation to progressively refine depth maps. Through extensive experiments on DTU dataset, Tanks and Temples dataset and ETH3D high-res benchmark, we show that our method generalizes well to various datasets and achieves promising results, demonstrating its superior performance on robust dense 3D reconstruction.

Keywords Multi-view stereo networks · Visibility estimation · Anti-noise training strategy · Inverse depth regression · Average group-wise correlation · Ordinal-based high-resolution depth map refinement

Q. Xu
E-mail: qingshanxu@hust.edu.cn
W. Su
E-mail: suwanjuan@hust.edu.cn
Y. Qi
E-mail: qiyuhang@hust.edu.cn
W. Tao (Corresponding author)
E-mail: wenbingtao@hust.edu.cn
M. Pollefeys
E-mail: marc.pollefeys@inf.ethz.ch
[1]Huazhong University of Science and Technology, Wuhan, China
[2]ETH Zürich, Zürich, Switzerland
[3]Microsoft

1 Introduction

Multi-View Stereo (MVS) has attracted great interest in the past few years for its wide applications in autonomous driving, virtual/augmented reality, 3D printing etc. The goal of MVS is to establish the 3D model of a scene from a collection of 2D images with known camera parameters. Recently, this task is always decomposed into two separate steps, depth map estimation and fusion, due to their high efficiency and flexibility. Of these two stages, depth map estimation plays an important role in the whole pipeline and many MVS methods [58, 14, 41, 48, 23, 53, 54] have put effort into accurate depth sensing.

The core of depth map estimation is to compute the correspondence of each pixel across different images by measuring the similarity between these pixels. Traditional methods [14, 41, 48] depend on hand-crafted similarity metrics, e.g., sum of absolute differences (SAD) and normalized cross correlation (NCC), and thus these metrics are sensitive to textureless areas, reflective surfaces and repetitive patterns. To deal with the above challenges, some methods [33, 21] resort to regularization technologies, such as graph-cuts and cost filtering.

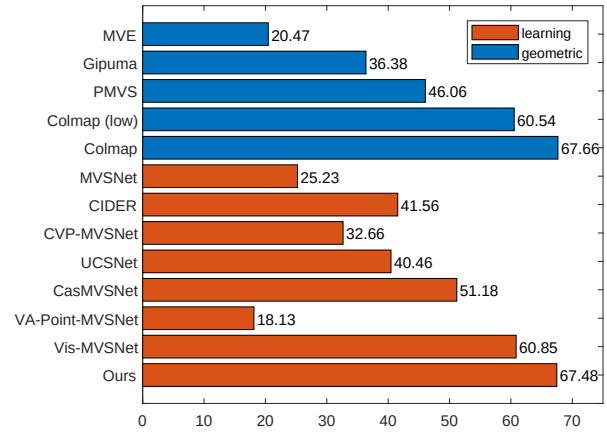


Fig. 1 Comparison between learning and geometric MVS methods on ETH3D high-res multi-view benchmark [43].

However, these engineered regularization methods still struggle in the above challenging areas.

To overcome the above difficulties, recent works [53, 54, 23, 15] leverage deep convolutional neural networks (DCNN) to learn multi-view depth inference. Thanks to their powerful cost volume representation and filtering, some learning-based multi-view stereo methods achieve promising results [53, 54, 8, 49, 9] and even outperform some traditional geometric methods, e.g., Colmap [41] on some datasets, e.g., Tanks and Temples dataset [32]. However, they are rarely evaluated on ETH3D high-res benchmark [43] in their experiments. Therefore, we test the performance of recently published learning-based methods on ETH3D high-res benchmark and show the performance comparison between geometric-based and learning-based MVS methods in Fig. 1. It is obvious that there exists a clear gap between geometric-based and learning-based MVS methods on this dataset. The results in Fig. 2 further demonstrate that the estimated depth maps of some learning-based MVS methods seriously degrade compared with Colmap. To explain this phenomenon, we first analyze the difference in characteristics of these two datasets. Tanks and Temples dataset provides video sequences as input, and the viewpoint between adjacent images often changes slightly. This means that almost all areas of some source images are visible in the reference image (Fig. 2). Unlike the Tanks and Temples dataset, the images provided by ETH3D high-res benchmark contain strong variations in viewpoint, resulting in complicated visibility association (Fig. 2). This requires MVS methods to consider the pixelwise visibility information of different source images. On the other hand, the existing learning-based methods [53, 54, 8, 49, 9] are almost tailored for video sequences with continuous viewpoint changes. Since they assume that there exist neighboring images that have strong visibility association with the reference image, they usually select these images as input from the perspective of global view selection and treat these images equally to construct an indiscriminate multi-view aggregated cost volume. Therefore, visibility estimation is totally ignored in almost all networks. However, treating each source image equally will make the cost volume susceptible to the noise from unrelated source images. This greatly limits the performance of learning-based methods on datasets like ETH3D high-res benchmark with wide baselines. Note that, there exist two concurrent works [7, 57] that also estimate visibility information to improve the performance of learning-based MVS. However, these two works still focus on the datasets with narrow baselines, making their performance still limited on datasets with wide baselines (See Fig. 1 and Fig. 2). *Therefore, to make learning-based MVS methods truly feasible in practice, it is significant to learn the pixelwise visibility information of source images in deep neural networks.*

In this work, we propose a Pixelwise Visibility-aware multi-view Stereo Network (PVSNet) to achieve the robust 3D reconstruction on different datasets. Specifically, through plane-sweeping with multiple sampling depths, we first construct the two-view cost volume for each neighboring image. Then, we regress the 2D visibility probabilistic maps from two-view cost volumes by using the proposed pixel visibility estimation network. To this end, we can explicitly establish the visibility association between each neighboring image and the reference image. With the help of visibility information, the previous two-view cost volumes can be aggregated into a robust unified one in a weighted manner. This greatly reduces the influence of noise from unrelated neighboring images. To make the pixelwise visibility network more discriminative to unrelated views, we further propose an anti-noise training strategy to introduce disturbed views during model training while the existing learning-based methods only use the best two neighboring views for training.

Additionally, existing multi-view depth inference networks [23, 53, 54] usually cast this task as a depth regression problem or an inverse depth classification problem. Depth regression samples depth hypotheses in depth space and regresses sub-pixel depth estimation by calculating the expectation of sampling depths. In contrast with depth regression, inverse depth classification samples depth hypotheses in inverse depth space and picks the depth value with the maximum probability as the estimation. As shown in Fig. 3, although the sampled depth values in depth regression are uniformly distributed in depth space, their projected 2D points in a source image are not distributed uniformly along the epipolar line. This impairs the feature discrimination

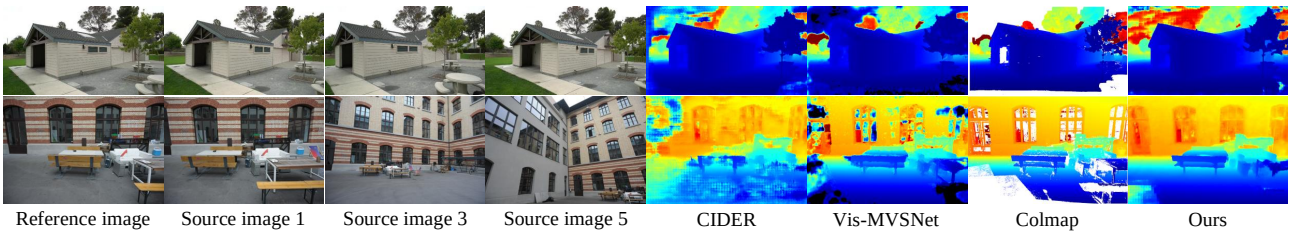


Fig. 2 Depth map results using CIDER [49], Vis-MVSNet [57], Colmap [41] and Ours on Tanks and Temples dataset [32] and ETH3D high-res multi-view benchmark [43] respectively. The former dataset is with narrow baselines while the later one is with wide baselines.

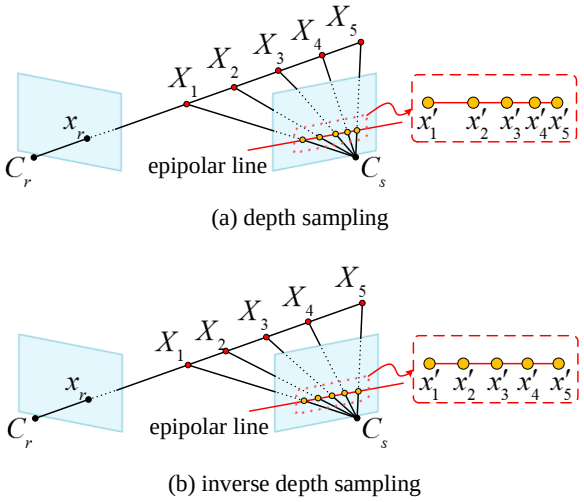


Fig. 3 Illustration of depth sampling and inverse depth sampling. (a) Depth sampling. Depth hypotheses are uniformly sampled in depth space. When these uniformly distributed depth hypotheses are projected to a source image, their corresponding 2D points are not distributed uniformly along the epipolar line. (b) Inverse depth sampling. Depth hypotheses are uniformly sampled in inverse depth space. When these sampled depth hypotheses are projected to a source image, their corresponding 2D points are distributed more uniformly along the epipolar line than depth sampling.

along the epipolar line. In contrast, inverse depth classification can better alleviate this problem. In this way, inverse depth sampling allows networks to reconstruct large-scale and complex scenes while depth sampling is usually suitable for object reconstruction. On the other hand, it is obvious that the depth classification always introduces stair effects while depth regression achieves sub-pixel estimate. In this work, we absorb both advantages of the above strategies and cast the multi-view depth inference as an inverse depth regression problem. We sample depth hypotheses in inverse depth space and record their corresponding ordinals. In this way, we first regress sub-pixel ordinal and then convert it to the final depth value, which is used to guide the training of our network. The inverse depth regression enables our network to be applied in large-scale scenes.

To make our network more scalable and achieve high-resolution estimation, it is important to construct lightweight cost volumes. In general, the size of cost volumes is $C \times D \times H_f \times W_f$, where C is the channel size, D is the number of sampled depth values, $H_f \times W_f$ is the spatial resolution of feature maps. In order to reduce the size of cost volumes, we achieve this from two aspects. Inspired by [16] in stereo matching, we apply the group-wise correlation to directly reduce channel size without extra model parameters. This can not only reduce the memory consumption but also reduce the computational burden in the cost volume filtering. Moreover, following the coarse-to-fine strategy used in [15,9,52], we further design an ordinal-based uncertainty estimation strategy to perform progressive depth map refinement. This strategy determines the sampling space of high-resolution refinement by estimating the variance of the previously regressed ordinal. This greatly reduces the depth sampling number of each stage.

With the above proposed strategies, our network achieves promising reconstruction results on DTU dataset [1], Tanks and Temples dataset [32] and ETH3D high-res benchmark [43]. Our contributions are four-fold. 1) We propose a pixelwsie visibility-aware group-wise correlation similarity measure to construct a lightweight cost volume. This measure not only allows our network to be truly applied to datasets with strong viewpoint changes, but also greatly eases the memory burden of our network. 2) We propose a pixelwise visibility estimation network to regress 2D visibility maps from two-view cost volumes and develop an anti-noise training strategy to train the network. The visibility maps can reflect the influence of occlusion, illumination, and unstructured viewing geometry. This allows good views to have larger weights in the final cost volume representation. 3) We treat the multi-view depth inference problem as an inverse depth regression task and demonstrate that the inverse depth regression can reach more robust and accurate results in large-scale scenes. 4) We design an ordinal-based uncertainty estimation strategy for high-resolution depth map refinement. This strategy fits in the 3D reconstruction of large-scale scenes.

This paper is an extension of our conference paper [49]. In the current paper, we first conduct a comprehensive performance evaluation of several popular learning-based MVS methods on ETH3D high-res benchmark and provide an insightful analysis on their limitation on the datasets with wide baselines. Then, we present a pixelwise visibility-aware network to estimate visibility maps of neighboring images and design an anti-noise training strategy to train the network. In addition, we present an ordinal-based uncertainty estimation for high-resolution depth map estimation. In experiments, besides DTU dataset and Tanks and Temples dataset, we also evaluate our method on ETH3D high-res benchmark. This further demonstrates the good generalization of our method in practice.

2 Related Work

Our proposed method is closely related to some learning-based works in stereo matching and multi-view stereo. We briefly introduce these works in the following. Also, we briefly reviews regularization technologies in traditional multi-view stereo.

Traditional Multi-View Stereo Kolmogorov and Zabih [33] formulate an energy minimization problem to enforce visibility constraints and spatial smoothness for regularization. As the energy function is NP-hard to minimize exactly, they also give a graph cut algorithm to solve the problem [4]. As the graph cuts solvers are usually computationally expensive, Hirschmuller [20] fisrt proposes Semi-Global Matching (SGM) to approximately optimize a 2D MRF problem with several 1D optimization problems in stereo matching. It has been shown that SGM can also be applied for multiple images [17]. In [10], the authors propose a plane-sweeping algorithm for stereo reconstruction. This leads to a cost volume representation for 3D scenes, which can be casted as labeling problems and efficiently solved by cost volume filtering technique [21]. As the above methods all treat the stereo reconstruction as discrete labeling problems, they cannot achieve sub-pixel estimation. To solve this problem, Bleyer et al. [3] adopt the sampling and propagation idea of PatchMatch [2] to solve a continuous stereo problem. The PatchMatch Stereo implicitly imposes the smoothness priors in scene space. [58,14,41, 48,50,19] extend the PatchMatch idea in multi-view stereo and achieve promising performance for the 3D reconstruction.

Learning-based Stereo Matching Stereo matching aims to estimate disparity for a pair of rectified images with small baselines. It can be deemed as a special case of multi-view stereo. With the development of DCNN, many learning-based stereo matching methods have been proposed. Žbontar and LeCun [47] first introduce a Siamese network to compute matching costs between two image patches. After getting unary features for left and right image patches, these features are concatenated and passed through fully connection layers to predict matching scores. Instead of concatenating unary features, Luo et al. [36] propose a inner product layer to directly correlate unary features. This accelerates the computation of matching cost prediction. In order to achieve end-to-end disparity estimation, DispNet [37] is proposed with an encoder-decoder architecture. Kendall et al. [28] leverage geometry knowledge to form a cost volume by concatenating left and right image features and utilize multi-scale 3D convolutions to regularize the cost volume. Chang and Chen [6] employ a spatial pyramid pooling module to incorporate global context information and use a staked hourglass architecture to learn more context information. Tulyakov et al. [46] compress the concatenated left-right image descriptors into compact matching signatures to decrease the memory footprint. Guo et al. [16] propose a group-wise correlation to measure feature similarities, which will not lose too much information like full correlation but reduce the memory consumption and network parameters.

Learning-based Multi-View Stereo Hartmann et al. [18] propose to learn a multi-patch matching function by n-way Siamese networks and the mean operation. To select credible view pairs, SurfaceNet [25] crops two representative patches around the projected center voxel and learns the relative weight for different image pairs by global view selection. LSM [26] unprojects image features into 3D feature grids by perspective geometry and fuses them into a unified one by recurrent neural networks. It implicitly considers the importance of different views but may be sensitive to the ordering of input images. To make networks order-agnostic and adapt to an arbitrary number of input images, DeepMVS [23] utilizes the max-pooling to select the most useful view but cannot make full use of multi-view information. MVSNet [53] and many following works [51,54, 8,55,9,52] adopt a variance-based multi-view similarity metric to capture the second moment information. In essence, this metric treats each source view equally and cannot distinguish useful views. Similarly, the mean operation used in [34,35,24,49] also faces the same problem. Note that, although some [34,35] of them use attention mechanism or hybrid 3D U-Nets to boost the performance, the constructed initial cost volumes are more noise-contaminated than our visibility-aware one. Despite there exist two concurrent works [7,57] that also estimate visibility information, their performance on the datasets with wide baselines is still limited by

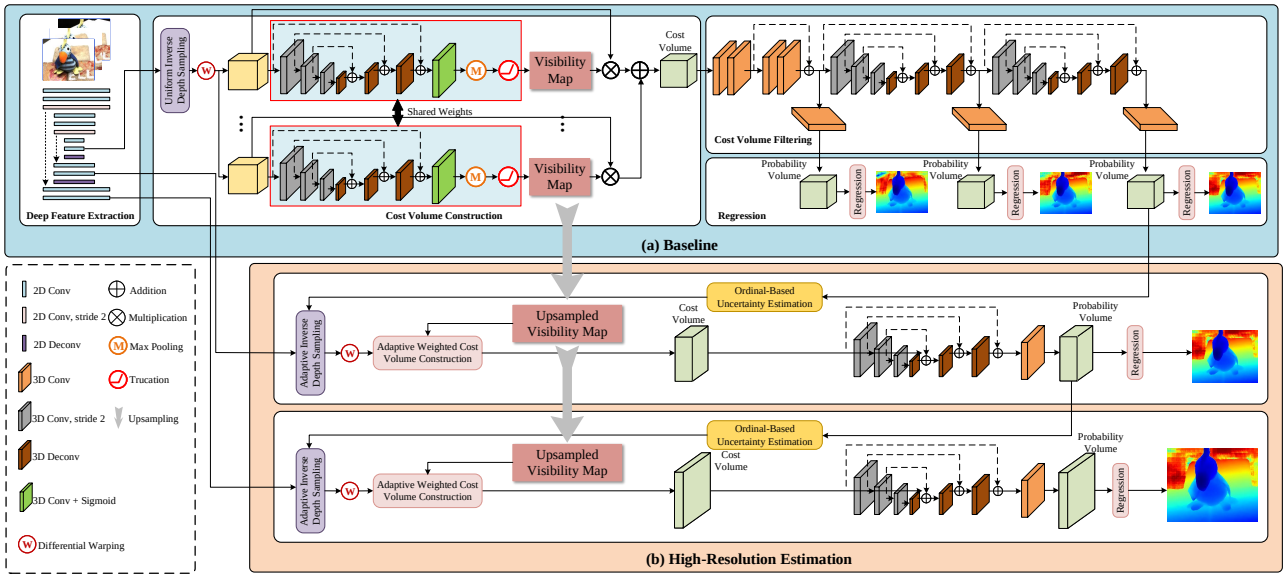


Fig. 4 Network architecture of our proposed PVSNet. It consists of two parts: (a) baseline and (b) high-resolution estimation. (a) Baseline (Vis-CIDER): Feature maps are extracted via a weigh-sharing deep feature extraction module for a reference image and source images. The feature maps of source images are warped into the coordinate of the reference image by differential warping with the uniform inverse depth values. Two-view cost volumes of the reference view and each source view are constructed by the group-wise correlation module, and the pixelwise visibility learning network in the red box of (a) is used to regress the two-view cost volumes to obtain the visibility maps. The multiple two-view cost volumes are further aggregated into a unified cost volume weighted by the visibility maps. The predicted depth maps are obtained by imposing cost volume filtering and regression on the cost volume. (b) High-resolution estimation: To generate the adaptive inverse depth hypotheses, the ordinal-based uncertainty is computed based on the probability volume obtained from the previous stage. With the adaptive inverse depth values, a thin two-view cost volume is built by differential warping and group-wise correlation. And the multiple two-view cost volumes are also aggregated into a unified cost volume weighted by the upsampled visibility maps obtained at the previous scale. The high-resolution depth map of the reference image is produced through a 3D U-Net and inverse depth regression. This progress is iterated until the depth map with the same resolution as the reference image is obtained.

their visibility estimation networks or training strategies. The VA-Point-MVSNet [7] directly processes the target scene as point clouds, where MLP with max-pooling operation is used to learn visibility information. The Vis-MVSNet [57] integrates the pixel-wise occlusion information in the MVS network through inferring the pair-wise uncertainty map and the pair-wise depth map jointly. Different from them, we regress pair-wise visibility map directly, and introduce a new anti-noise training strategy to improve the robustness of the network.

3 Method

Our proposed network takes as input a reference image $I_{\text{ref}} = I_0$ and source images $\boldsymbol{I}_{\text{src}} = \{I_i | i = 1 \dots N-1\}$ with their camera parameters to predict the depth map for the reference image, where N is the total number of input images. As shown in Fig. 4, our proposed network, PVSNet, contains two parts: (a) baseline and (b) high-resolution estimation. The baseline (Vis-CIDER) produces the depth map of size $\frac{H}{4} \times \frac{W}{4}$ for the reference image, where H and W denote height and width of the input image respectively. Based on our Vis-CIDER, high-resolution estimation focuses on estimating the depth map with the same resolution as the reference image. Specifically, the Vis-CIDER consists of four modules: deep feature extraction, visibility-aware cost volume construction, cost volume filtering and regression. Multi-scale deep features are first extracted for all input images by a shared feature extraction module. According to the uniform inverse depth hypotheses, we construct two-view cost volumes by differential warping and group-wise correlation. The visibility map of each source image is regressed based on each two-view cost volume. With these visibility maps, multiple two-view cost volumes are further aggregated into a unified cost volume. The depth maps of size $\frac{H}{4} \times \frac{W}{4}$ are inferred by cost volume filtering and inverse depth regression. Note that, without visibility learning when constructing cost volumes and aggregating two-view cost volumes using visibility map of each source image filled with the scalar value 1, the multi-view similarity metric becomes an average group-wise correlation measure and the baseline model will turn into CIDER. Afterwards, our high-resolution estimation starts by computing the

ordinal-based uncertainty estimation according to the probability volume obtained from the previous stage. The ordinal-based uncertainty estimation is utilized to generate adaptive inverse depth hypotheses. Then we still leverage the group-wise correlation and visibility maps obtained at the coarsest scale to construct cost volumes. The higher-resolution estimation is obtained by cost filtering and inverse depth regression. This process is iterated until we obtain the depth map with the same resolution as the reference image. Moreover, to improve the robustness of visibility learning, a new anti-noise training strategy is used during the training.

3.1 Deep Feature Extraction

In traditional methods, the original image representations are directly used to construct the cost volume. This may result in the lack of context information in some ambiguous regions, e.g., low-textured surface, repetitive patterns and reflective regions, making the depth estimation in these regions failed. To incorporate context information, we adopt a 2D U-Net [39] module to extract multi-scale deep features. In this way, for each input image with resolution $3 \times H \times W$, a multi-scale deep image features with size $32 \times \frac{H}{4} \times \frac{W}{4}$, $16 \times \frac{H}{2} \times \frac{W}{2}$ and $8 \times H \times W$ can be obtained.

3.2 Pixelwsie Visibility-aware Correlation Cost Volume Construction

After getting the deep features for all input images, we hope to encode these features together with the camera parameters into the network to enable its geometry awareness. To this end, we first construct the two-view cost volume between each source image and the reference image via group-wise correlation similarity measurement. We then regress the visibility map of each source image based on each cost volume. With these visibility maps, multiple two-view cost volumes are further aggregated into a unified cost volume.

3.2.1 Two-view Cost Volume Construction via Group-wise Correlation

Inspired by the traditional plane sweep stereo [10], recent learning-based MVS methods, e.g., MVSNet, DeepMVS and R-MVSNet, sample depth hypotheses in 3D space. Based on the sampled depth hypotheses, the feature representations of source images can be warped into the coordinate of the reference camera to construct a cost volume. Our network also leverages this idea to construct our cost volume. For a pixel $\mathbf{p}$ in the reference images I_{ref}, given the j-th sampled depth value d_j $(j = 0 \dots D-1)$, its corresponding pixel $\mathbf{p}_{i,j}$ in the source image I_i is computed as

$$\mathbf{p}_{i,j} = \mathbf{K}_i(\mathbf{R}_{\text{ref},i}(\mathbf{K}_{\text{ref}}^{-1}\mathbf{p}d_j) + \mathbf{t}_{\text{ref},i}), \tag{1}$$

where D is the total sample number of depth values, $\mathbf{K}_{\text{ref}}$ and $\mathbf{K}_i$ are the intrinsic parameters for the reference image I_{ref} and the source image I_i, $\mathbf{R}_{\text{ref},i}$ is the relative rotation and $\mathbf{t}_{\text{ref},i}$ is the relative translation. With the above transformation, the deep features of all source images $\boldsymbol{\mathcal{F}}_{\text{src}} = \{\mathcal{F}_i | i = 1 \dots N-1\}$ can be warped into the coordinate of the deep feature of the reference image $\mathcal{F}_{\text{ref}}$. The warped deep features of all source images at depth d_j are denoted as $\tilde{\boldsymbol{\mathcal{F}}}_{\text{src},j} = \{\tilde{\mathcal{F}}_{i,j} | i = 1 \dots N-1\}$.

In order to measure the multi-view feature similarity, MVSNet [53] employs a variance-based metric to generate a raw 32-channel cost volume. As the cost volume representation is proportional to the model resolution, it always makes the network have a huge memory footprint. As pointed out in [53], before feeding the cost volume into the subsequent cost volume regularization module, MVSNet first reduces the 32-channel cost volume to an 8-channel one. Also, the authors of [46] demonstrate that feeding an 8-channel cost volume which is compressed from a 32-channel cost volume into the regularization module can reach a similar accuracy. This makes us believe that the raw 32-channel cost volume representation may be redundant. Although the above works take the 8-channel cost volume as the input of the cost volume regularization module, they require an extra module to compress the raw 32-channel. This not only increases the computational requirement but also the memory consumption. Thus, we intend to construct a raw 8-channel cost volume to simultaneously reduce the computational requirement and the memory consumption.

Inspired by the group-wise correlation in [16], we first propose an average group-wise correlation similarity measure to construct a lightweight indiscriminate cost volume. Specifically, for the deep reference image feature $\mathcal{F}_{\text{ref}}$ and the i-th warped deep source image feature at depth d_j, $\tilde{\mathcal{F}}_{i,j}$, their feature channels are evenly divided into G groups along the channel dimension. Then, the g-th group similarity between $\mathcal{F}_{\text{ref}}$ and $\tilde{\mathcal{F}}_{i,j}$ is computed as

$$S_{i,j}^{g} = \frac{1}{32/G} \left\langle \mathcal{F}_{\text{ref}}^{g}, \tilde{\mathcal{F}}_{i,j}^{g} \right\rangle, \tag{2}$$

where $g = 0 \dots G-1$, $\mathcal{F}_{\text{ref}}^{g}$ is the g-th feature of $\mathcal{F}_{\text{ref}}$, $\tilde{\mathcal{F}}_{i,j}^{g}$ is the g-th feature of $\tilde{\mathcal{F}}_{i,j}$ and $\langle \cdot, \cdot \rangle$ is the inner product. When the feature similarities of all G groups are computed for $\mathcal{F}_{\text{ref}}$ and $\tilde{\mathcal{F}}_{i,j}$, they are packed into

a G-channel feature similarity map $S_{i,j}$. As there are D sampled depth values, the D feature similarity maps between the reference image and the i-th source image are further packed into the two-view cost volume $C_{\text{ref},i}$ of size $G \times D \times \frac{H}{4} \times \frac{W}{4}$. By setting $G = 8$, we can directly obtain raw 8-channel cost volumes.

In order to adapt an arbitrary number of input source images, CIDER employs an average group-wise correlation metric to compute the following final multi-view cost volume:

$$C = \frac{1}{N-1} \sum_{i=1}^{N-1} C_{\text{ref},i}. \tag{3}$$

This metric leads an indiscriminate multi-view similarity definition, making learning-based MVS methods unable to tackle scenarios with strong viewpoint variations. In next section, we will further introduce our pixelwise visibility-aware cost volume aggregation.

3.2.2 Pixelwise Visibility-Aware Cost Volume Aggregation.

After obtaining the two-view cost volume $C_{\text{ref},i}$, we hope to leverage it to regress the visibility map for the source image I_i. This is possible because the two-view cost volume $C_{\text{ref},i}$ encodes the confidence of different sampling depths [22,38,30,29]. The network architecture of our pixelwise visibility network is shown in the red boxes of Fig. 4 (a). Since the distribution of two-view cost volumes is often non-discriminative [44,11], we first apply the pixelwise visibility network which consists of a 3D U-Net [39] to modulate cost volumes. Our 3D U-Net uses a three-scale encoder-decoder structure to increase the receptive field. Except for the last convolution layer that produces one-channel feature followed by the sigmoid activation function, other convolution layers are followed by a batch-normalization (BN) layer and a rectified linear unit (ReLU). Then, we apply the max-pooling along the depth dimension to obtain the visibility map V_i for the source image I_i

$$V_i(\mathbf{p}) = \max\{\mathbf{P}_v(j, \mathbf{p}) | j = 0, \cdots, D-1\}, \tag{4}$$

where $\mathbf{P}_v(j, \mathbf{p})$ is the probability estimation for pixel $\mathbf{p}$ at the j-th sampling depth value. Intuitively, if a pixel is visible in a source image, there will exist an obvious peak among multiple matching scores. Otherwise, all its corresponding matching scores will be relatively low. Thus, it is possible to apply max-pooling to find the visibility proxy for different source images. To further eliminate the influence of unrelated source images, we deactivate the images whose visibility probability is below a certain threshold. Then, the visibility map of each source image is modified as

$$V_i'(\mathbf{p}) = \begin{cases} V_i(\mathbf{p}), & \text{if } V_i(\mathbf{p}) > \tau; \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

where $\tau = 0.05$ is a threshold that controls the activation of source images. The above equation is similar to ReLU and our overall network can be trained end-to-end with back-propagation. In this way, the final aggregated cost volume is computed by

$$C_{\text{agg}}(\mathbf{p}) = \frac{\sum_{i=1}^{N-1} V_i'(\mathbf{p}) \cdot C_{\text{ref},i}(\mathbf{p})}{\sum_{i=1}^{N-1} V_i'(\mathbf{p})}. \tag{6}$$

The final aggregated cost volume is the sum of each two-view cost volume weighted by its visibility map. This definition differs from all previous multi-view similarity measures [18,26,53,23] because it considers the pixelwise visibility information of each neighboring view. This not only reduces the influence of noise but also lessens the regularization burden of cost volume filtering.

3.3 Cost Volume Filtering

As pointed out in [28], in order to regress the final sub-pixel estimation, it is import to keep the probability distribution along the depth dimension at each pixel location uni-model. To this end, many works [6,16,56] repeat the same cost volume regularization module to filter cost volumes. Inspired by this idea, we design a cascade 3D U-Net to regularize the above raw 8-channel cost volume.

Before the cascade 3D U-Net, we set up a residual module and a regression module to let our network learn a better feature representation as [16] does. Then, to handle the depth estimation in some ambiguous regions, two 3D U-Nets are cascaded to filter the cost volume. Due to the repeated top-down/bottom-up processing structure, our network can learn more context information. The detailed structure of our cost volume filtering module is shown in Fig. 4. Note that, the previous MVS networks never employ this structure due to their large memory consumption caused by the huge cost volume representation, e.g., MVSNet [53] and R-MVSNet [54]. This makes their incorporated context information limited. Thanks to our lightweight cost volume representation, a progressive cost volume filtering can be conducted in our network.

3.4 Inverse Depth Regression

In order to achieve the sub-pixel estimation, [28] first uses disparity regression to estimate the continuous disparity map in stereo matching. As the images are rectified in stereo matching, the uniform disparity sampling in the disparity space results in a uniformly distributed 1D correspondence search problem. Differently, as the images in the multi-view setup are not rectified, the direct depth sampling in the depth space will not lead to the similar distribution in the epipolar line of neighboring images (Fig. 3 (a)).

As described in the Section 3.2, the sampled depth hypotheses will be projected to neighboring images to obtain a series of 2D points. To make these 2D points that lie in the same epipolar line distribute as uniformly as possible, discrete depth hypotheses are uniformly sampled in inverse depth space as follows,

$$d_j = ((\frac{1}{d_{min}} - \frac{1}{d_{max}})\frac{j}{D-1} + \frac{1}{d_{max}})^{-1}, j = 0 \dots D-1, \tag{7}$$

where d_{min} and d_{max} are the minimal depth value and the maximal depth value of the reference image (Fig. 3 (b)). With the above depth value sampling scheme, we can construct a more discriminative cost volume to be sent to the subsequent cost volume filtering module.

As shown in Fig. 4, there are three output branches in our network. In each branch, the filtered cost volume is converted to a 1-channel probability volume via a 3D convolution operation and the softmax function. To obtain the final continuous depth estimate, we first regress the sub-pixel ordinal k from the probability volume as follows,

$$k = \sum_{j=0}^{D-1} j \times p_j, \tag{8}$$

where p_j is the probability at depth value d_j. The final predicted depth value for each pixel is computed as

$$\hat{d} = ((\frac{1}{d_{min}} - \frac{1}{d_{max}})\frac{k}{D-1} + \frac{1}{d_{max}})^{-1}. \tag{9}$$

To train our network, we use the ground truth depth map as our supervised signal. We denote the ground truth depth map as $\boldsymbol{d}$ and the three predicted depth maps as $\hat{\boldsymbol{d}}_0$, $\hat{\boldsymbol{d}}_1$ and $\hat{\boldsymbol{d}}_2$. Our final loss function of the baseline model is defined as

$$L = \sum_{q=0}^{2} \lambda_q l(\boldsymbol{d}, \hat{\boldsymbol{d}}_q), \tag{10}$$

where λ_q denotes the weight for the q-th predicted depth map and $l(\cdot, \cdot)$ is the mean absolute difference.

3.5 Ordinal-Based High-Resolution Refinement

It has been shown that high-resolution depth maps are beneficial for improving the completeness of 3D models [54,34,8]. Directly upsampling depth maps inevitably introduces artifacts. Therefore, R-MVSNet [54] employs an individual post-precessing step, variational refinement, to optimize upsampled depth maps. To make full use of the potential of DCNN, P-MVSNet [34] concatenate upsampled depth values and high-resolution texture information to regress high-resolution depth maps. Point-MVSNet [8] defines a sequence of equidistant depth hypotheses for the current upsampled depth values. Then a point-based inference network is utilized to refine the upsamled depth maps. Some cascade methods [15,9,52] adopt a coarse-to-fine strategy to sample depth planes for high-resolution depth map refinement. All these methods perform high-resolution depth map refinement in the depth domain. This makes them not scalable to large-scale scenes. Based on our inverse depth regression, we further propose an ordinal-based uncertainty estimation strategy for high-resolution depth map refinement. This strategy also uses the pixelwise visibility-aware correlation cost volume to construct lightweight adaptive thin cost volumes.

Our ordinal-based high-resolution depth map refinement is conducted in a coarse-to-fine manner. In our refinement, our whole algorithm contains L scales ($L = 3$). Our baseline method Vis-CIDER constitutes the first stage of the refinement progress. It produces the lowest-resolution depth map prediction of the coarsest scale ($l = 0$) and its corresponding probability volume of size $D \times \frac{H}{4} \times \frac{W}{4}$. As mentioned before, for each pixel of the coarsest depth map, its corresponding column in the probability volume stands for the probability distribution of sampled ordinal values. According to the probability distribution of the regressed sub-pixel ordinal k, its variance is computed as

$$\sigma^2 = \sum_{j=0}^{D-1} (j-k)^2 \times p_j. \tag{11}$$

Then, the uncertainty interval of the regressed sub-pixel ordinal k is measured as

$$u = [k - \alpha\sigma, k + \alpha\sigma], \tag{12}$$

where σ is the corresponding standard deviation and α is a scalar determining how large the uncertainty interval is. For each pixel, its uncertainty interval is adaptively computed. Thus, we can adaptively generate finer sampling depth values by uniformly sampling in ordinal-based uncertainty interval. Specifically, in order to obtain higher-resolution depth map prediction at

scale l, we first produce the ordinal map $\widetilde{O}_l$ and uncertainty length $\widetilde{U}_l$ of the current scale l by upsampling the ordinal map O_{l-1} and uncertainty length U_{l-1} of the previous scale $l-1$. According to these two estimates, we generate new finer ordinal hypotheses at the scale l as follows,

$$H_l = \{\widetilde{O}_l - \frac{\widetilde{U}_l}{2} + \frac{0}{s_l - 1} \cdot \widetilde{U}_l, \ldots, \widetilde{O}_l - \frac{\widetilde{U}_l}{2} + \frac{s_l - 1}{s_l - 1} \cdot \widetilde{U}_l\}, \tag{13}$$

where s_l is sampling number of depth hypotheses at the scale l. With these finer ordinal hypotheses, we can get finer depth values according to Eq. (9), and then apply our proposed pixelwise visibility-aware correlation cost volume to construct the adaptive thin cost volume at the scale l. Note that, the visibility maps at the scale l are directly upsampled from the previous scale. Afterwards, we regularize the cost volume with a 3D U-Net and compute depth map predictions with inverse depth regression. The predicted depth maps at scales $l = 1$ and $l = 2$ are denoted as $\hat{\boldsymbol{d}}_3$ and $\hat{\boldsymbol{d}}_4$ respectively. The loss function to train our ordinal-based refinement network is defined as

$$L_r = L + \lambda_3 l(\boldsymbol{d}_{\uparrow}, \hat{\boldsymbol{d}}_3) + \lambda_4 l(\boldsymbol{d}_{\uparrow\uparrow}, \hat{\boldsymbol{d}}_4), \tag{14}$$

where $\boldsymbol{d}_{\uparrow}$ and $\boldsymbol{d}_{\uparrow\uparrow}$ are the ground truth depth maps at scales $l = 1$ and $= 2$ and λ_3 and λ_4 are the weights for the higher-resolution depth map predictions.

3.6 Anti-noise training strategy

Ideally, we hope to supervise the training of our proposed pixelwise visibility network directly with the ground truth visibility information. However, due to occlusions, baseline angles and scale changes, the ground truth visibility map is different for different visible source images, making it hard to be labeled. Thus, we turn to leveraging the ground truth depth maps to indirectly supervise the training of the network. This implicitly makes the network discriminate the different importance of source images. Then, we encounter another dilemma.

Without considering pixelwise visibility information, all the previous learning-based methods [54,8,49,9] follow MVSNet [53] to select the best two neighboring views via global view selection for model training. However, the two selected views are too close to the reference image, so that there are the vast majority of visible pixels participating in the model training while only a few invisible pixels participate in model training. The extreme imbalance between positives and negatives prevents our method from fully exploiting the potential of the pixelwise visibility network. To alleviate this problem, we propose an anti-noise training strategy that introduces disturbed views. Specifically, we adopt the method in MVSNet to compute global view selection scores of neighboring views based on the number and baseline angles of sparse points between each source image and the reference image. Then, we choose the best two views and the worst two views to train our model. This training strategy introduces more negative samples, making our network more robust to unrelated views.

4 Experiments

In this section, we evaluate our proposed network on DTU dataset [1], Tanks and Temples dataset [32] and ETH3D high-res benchmark [43]. First, we describe the datasets and evaluation metrics followed by implementation details. Then, we perform ablation studies for CIDER and PVSNet. Last, we show the benchmarking results on the above datasets.

4.1 Datasets and Evaluation Metrics

DTU Dataset [1] This dataset contains more than 100 object-centric scenes. Each scene is captured at 49 or 64 fixed camera positions with 7 lighting conditions. The ground truth point clouds are scanned in the indoor controlled environments. Thus, The viewpoints and lighting conditions are all deliberately designed. The ground truth camera poses and ground truth point clouds are all publicly available. The image resolution is 1600×1200.

Tanks and Temples Dataset [32] This dataset provides both indoor and outdoor scenes. The dataset is further divided into Intermediate datasets and Advanced datasets. Compared to the Intermediate datasets, the Advanced datasets contain larger scale and more complex scenes. Their ground truth camera poses and ground truth point clouds are withheld by the evaluation website. Additionally, this dataset also provides training datasets with their ground truth 3D models available.

ETH3D High-res Benchmark [43] This benchmark is a dataset with strong viewpoint variations, which includes both indoor and outdoor scenes. Moreover, the image resolution of this benchmark is very high, reaching about $H \times W = 4000 \times 6000$. This benchmark contains training dataset and test dataset. Both of them provide ground truth camera parameters. The ground truth point clouds are only available for training dataset.

Evaluation Metrics The quality of depth prediction is evaluated by the commonly used mean absolute depth error (MAE). For point cloud evaluation, the accuracy and completeness of the distance metric are adopted for DTU dataset while the accuracy and completeness of the percentage metric for the other two datasets. The overall score of the distance metric is the mean of the accuracy and completeness while their F_1 score measures the overall score of the percentage metric.

4.2 Implementation Details

Training Following [25], we divide the DTU dataset into training set, validation set and test set. We train our network on DTU training set. As DTU dataset does not provide ground truth depth maps, we follow the idea in [53] to generate the depth maps at a resolution of 640×512 by leveraging the screened Poisson surface reconstruction [27]. During the training, the image size is scaled and cropped to 640×512. d_{min} and d_{max} are fixed to $425mm$ and $935mm$ respectively. For CIDER and Vis-CIDER, the total sample number of depth values is set to $D = 192$, the weights for loss function (10) are set to $\lambda_0 = 0.5$, $\lambda_1 = 0.5$ and $\lambda_2 = 0.7$. The total number of input image is set to $N = 3$ for CIDER, and its visibility map of each source image to fill with the scalar value 1. The total number of input image is set to $N = 5$ for Vis-CIDER and PVSNet, the global view selection includes 20 neighboring views for anti-noise training strategy. For PVSNet, the total sample number of depth values for three stage are set to $D_1 = 192$, $D_2 = 32$ and $D_3 = 8$, the weights for the loss function (14) are set to $\lambda_0 = 0.5$, $\lambda_1 = 0.5$, $\lambda_2 = 0.7$, $\lambda_3 = 0.7$ and $\lambda_4 = 0.7$. We implement our network by using PyTorch. Both CIDER and Vis-CIDER are trained for 10 epoch in total, and PVSNet is trained for 15 epoch. We use RMSprop as the optimizer and the initial learning rate is set to 0.001. The learning rate is decayed every $10,000$ iterations with a base of 0.9. The TITAN X GPU is used for training and test.

Filtering and Fusion In order to generate the final single 3D point cloud, we filter and fuse depth maps like other depth map based MVS methods [14,41,53, 48]. Specifically, a probability volume is generated in the regression part of our network. After obtaining the regressed sub-pixel ordinal for each pixel, we locate its corresponding 4-neighboring ordinals and accumulate the probabilities of these ordinals to obtain the final probability representation. This measures the reliability of the depth estimation for each pixel. Then, we filter out the pixels with probability lower than a threshold of 0.8 to produce a cleaner depth map. In our fusion step, we treat each input image as the reference image in turn. For each pixel in the reference image, we calculate its projected depth and coordinate in neighboring views according to its depth in the reference image. Further, we know the estimated depth in the projected coordinate. Then, we compute the relative depth difference between the projected depth and the estimated depth. With the corresponding depth in neighboring views known, we can compute the reprojected coordinate in the reference image in the same way. We define the distance between the reprojected coordinate and the original coordinate in the reference image as the reprojection error. A pixel will be deemed two-view consistent if its relative depth difference is lower than 0.01 and its reprojection error is smaller than 1 pixel. In our experiments, all pixels should be at least three-view consistent and their corresponding 3D points are averaged to produce the final point cloud.

4.3 Ablation Studies for CIDER

In this section, we explore the effectiveness of our proposed strategies in CIDER, including average group-wise correlation similarity, cascade 3D U-Net filtering and inverse depth regression. To this end, we define a Base model to prove the effectiveness of the above strategies. This Base model replaces the above strategies with variance-based similarity, 3D U-Net filtering and depth regression that are employed in MVSNet [53]. In order to simultaneously show the generalization of different models on unseen datasets, we use Tanks and Temples training datasets here to conduct experiments. These datasets contain 7 scenes. These scenes not only contains object-centric ones but also large-scale ones. The camera poses are obtained by COLMAP [40]. The image resolution is resized to 1920×1056 as [53,54] does. The depth sampling number is set to $D = 192$ and the input view number is $N = 5$ for all models.

Average Group-wise Correlation Similarity In order to validate the effectiveness of average group-wise correlation similarity, we replace the variance-based similarity in the Base model with the average group-wise correlation similarity and denote this model as AGC. This makes the cost volume size be reduced from $32 \times D \times \frac{H}{4} \times \frac{W}{4}$ to $8 \times D \times \frac{H}{4} \times \frac{W}{4}$. The results are shown in Table 1 and Fig. 5. We see that the total memory consumption is reduced by nearly half. Moreover, the reconstruction results of the AGC model remain almost the same as the Base model. This is because our proposed metric also explicitly measures the multi-view feature difference as the variance-based similarity does. As a result, the proposed average group-wise correlation similarity can not only aggregate the multi-view

Table 1 Ablation study results of proposed networks on Tanks and Temples training datasets [32] using the percentage metric (%). Due to the GPU memory limitation, image resolution is resized to 1536×832 for MVSNet. R-MVSNet\Ref. means R-MVSNet without variational refinement.

Model	Barn	Caterpllar	Church	Courthouse	Ignatius	Meetingroom	Truck	Mean	GPU Memory	Time
Base	24.82	6.34	39.03	37.26	11.92	22.90	12.85	22.16	11.1 GB	2.44 s
AGC	23.76	5.98	41.20	35.45	13.98	24.76	13.57	22.67	**6.5 GB**	**1.90 s**
AGC-IDR	54.00	48.39	37.48	35.25	64.26	27.89	61.48	46.96	**6.5 GB**	2.29 s
CIDER	56.44	49.38	40.53	36.28	64.95	**29.94**	63.09	48.66	7.4 GB	3.11 s
CIDER (D=256)	**56.97**	52.62	39.47	37.38	**67.71**	28.52	**64.56**	**49.60**	9.6 GB	4.24 s
MVSNet	24.87	6.97	37.69	35.50	11.36	21.75	17.12	22.18	11.7 GB	2.59 s
R-MVSNet\Ref.	51.42	**53.55**	**45.03**	**40.65**	67.26	23.06	62.13	49.01	**6.5 GB**	8.57 s

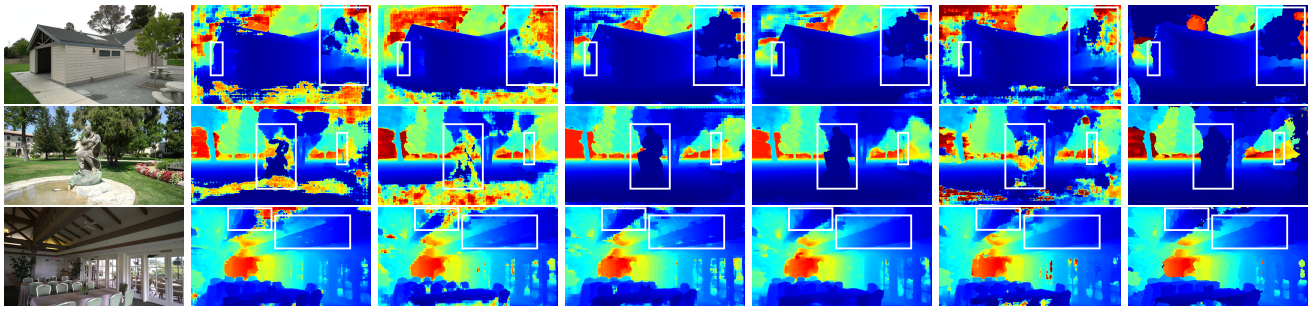


Fig. 5 Depth map reconstructions of Barn, Ignatius and Meetingroom, Tanks and Temples training datasets [32] using different settings of the proposed networks.

information well but also achieve a compact cost volume representation. Note that, our proposed average group-wise correlation similarity measure does not increase any model parameters. In fact, since the cost volume size is reduced, the module parameters of 3D U-Net are also reduced.

Inverse Depth Regression It can be seen from Table 1 that the Base model and the AGC model do not generalize well on Tanks and Temples training datasets. As mentioned before, we think that these two models do not carefully consider the epipolar geometry. To prove this, we replace the regression in the AGC model with the inverse depth regression and name this model as AGC-IDR.

We retrain the AGC-IDR model and show its test results on Tanks and Temples training dataset in Table 1 and Fig. 5. As shown in Table 1, this model outperforms the previous two models with a significant margin. Moreover, according to the visualization of the reconstructed depth maps in Fig. 5, the network with inverse depth regression can estimate depth maps more accurately than the networks with depth regression. This demonstrates that the inverse depth regression can better depict the distribution of the depth hypotheses in the epipolar line of neighboring images. Therefore, the network can accurately capture the true depth hypothesis.

Cascade 3D U-Net Filtering In our proposed overall network, CIDER, cascade 3D U-Net filtering is utilized to regularize the cost volume. In Fig. 5, the depth maps reconstructed by AGC-IDR still contain much noise in some ambiguous areas. We suppose that this can be attributed to its limited 3D U-Net regularization and further improve it with cascade 3D U-Net filtering. As shown in Fig. 5, CIDER can better suppress the noise in ambiguous areas than AGC-IDR. Therefore, it achieves better 3D reconstruction results than AGC-IDR, which can be seen from Table 1. It is noteworthy that although two 3D U-Nets are cascaded, the memory consumption is only slightly increased.

In addition, we increase the total depth sampling number from 192 to 256 over the same depth range. As illustrated in Table 1, the F_1 score on Tanks and Temples training datasets is increased from 48.66% to 49.60% and the memory consumption is still acceptable. Thus, we will fix the total depth sampling number to be 256 when comparing CIDER with other state of the art learning-based MVS methods on different benchmarks.

Comparison with Existing Methods We also compare our method with MVSNet [53] and R-MVSNet without variational refinement (R-MVSNet\Ref.) [54]. Table 1 shows that our method is much better than MVSNet due to our proposed strategies. Although R-MVSNet\Ref. employs the gated recurrent unit to reduce the memory consumption, it cannot incorporate enough context information to tackle the depth estimation in edges and ambiguous regions, e.g., white boxes shown in Fig. 5. Thus, our method is also better than R-

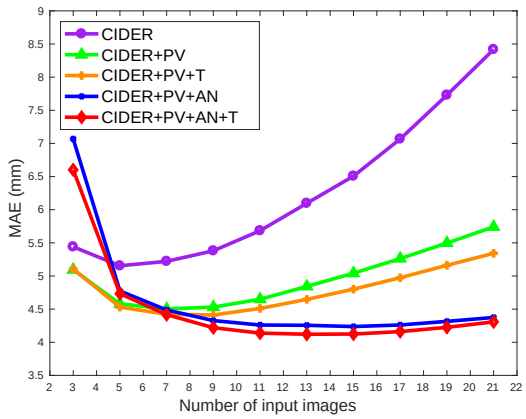


Fig. 6 The prediction error of different models for varying input numbers.

Table 2 Ablation study on the training set of ETH3D high-res benchmark using F_1 score (in %) (higher is better).

MVSNet	CIDER	ATV	PV+AN	$F_1 \uparrow$
✓				25.23
✓		✓		40.46
	✓			41.56
	✓		✓	53.20
	✓	✓		57.61
	✓	✓	✓	**67.48**

MVSNet\Ref. As for the running time, due to our proposed lightweight cost volume, the methods with correlation cost volume are faster than R-MVSNet, which acquires larger depth sampling number.

4.4 Ablation Studies for PVSNet

This section gives the ablation study of the PVSNet. Fig. 6 shows the experimental results of our low-resolution model (only using Eq. 10 to train our model) on DTU validation set. Table 2 demonstrates the effectiveness of different components of our method, where the model is trained on the training set of DTU dataset and test on the training set of ETH3D high-res benchmark without any fine-tuning.

Pixelwise Visibility Network (PV) We first test the PV by setting the visibility map of each source image to fill with the scalar value 1. This makes our method turn into CIDER [49], where each source image is treated equally in the multi-view cost volume aggregation. For a fair comparison, we only use the best two neighboring views to train CIDER and CIDER with PV. Here Eq. 5 will not be used. As shown in Fig. 6, CIDER achieves the best performance at 5 input views with 5.154 MAE (the purple curve). But its performance degrades dramatically when the input views increase. It shows that if not considering visibility estimation, more input views will bring more noise from unrelated views, which will seriously degrade the performance of the methods. Therefore, existing learning-based methods cannot use more input views like traditional geometric methods, which prevents them from being applied in practice. Comparatively, CIDER+PV (the green curve) gets 4.580 MAE at 5 input views and reaches the best at 7 input views (4.503). This shows that even for the datasets with slight viewpoint changes, visibility estimation is still crucial. With the increase of input views, the results of CIDER+PV also deteriorate, but the degradation is much less severe than that of CIDER. This is because the manifest imbalance between positives and negatives in the original training strategy makes PV less discriminative to unrelated views. To confirm this, we directly apply the Eq. 5 to test CIDER+PV (the orange curve). Its MAE is further reduced and the minimal MAE is 4.411. This shows it is useful to further lower the visibility probability of unrelated views. Since the original training strategy uses the best two neighboring views, their pixelwise visibility probability is relatively high in most cases. This makes PV tend to only fit the visibility of high probability during the training, resulting in its low discrimination to unrelated views. Therefore, with the increase of input views, the weighted sum of unrelated views will dominate in the aggregated cost volume, making the performance degrade.

Anti-noise Training Strategy (AN) To make the pixelwise visibility network (PV) more discriminative to unrelated views, we retrain CIDER+PV by anti-noise training strategy (AN). In contrast to the original training strategy, AN introduces the worst two neighboring views. With the AN, CIDER+PV gets better and better performance with the increase of input views and reaches the best with 4.237 MAE at 15 input views (cf. the blue curve in Fig. 6). Moreover, even if the input views are increased to 21, CIDER+PV+AN is still not degraded obviously. This indicates that CIDER+PV+AN can not only choose good views by assigning them higher probability, but also remove bad views by lowering their probability. When we further introduce the probability truncation (T) to train CIDER+PV+AN (note that this makes the model turn into Vis-CIDER), the results (the red curve) show its performance improves slightly and reach the best with 4.121 MAE at 13 input views. This also demonstrates our proposed AN can distinguish the unrelated views well. Note that, when the input views are less than 7, CIDER+PV+AN is not better than CIDER+PV. But after that CIDER+PV+AN becomes better and better while CIDER+PV is worse and worse. This means that the previous training strategy makes networks depend too much on the cost vol-

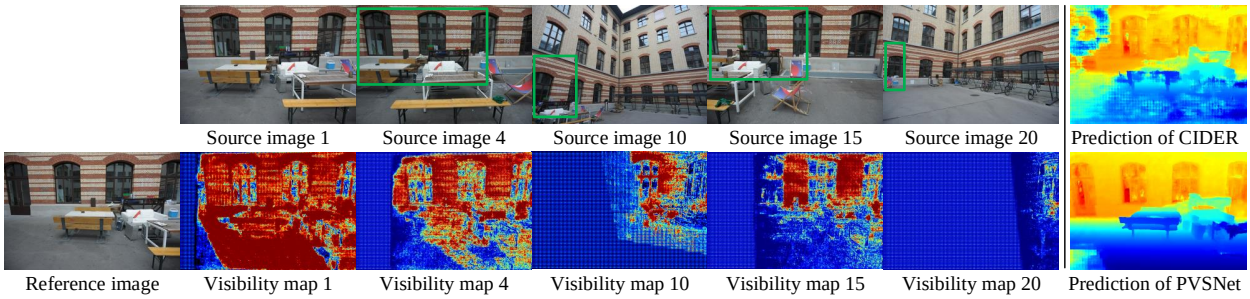


Fig. 7 Left: Visibility maps show the visible areas in the reference image with respect to source images. Right: Predicted depth map comparison between CIDER and our proposed PVSNet.

ume filtering while AN can make good use of multi-view information. Thus, when the input views are more than 7, relying on the cost volume filtering alone is not enough to counter the noise from unrelated views. In contrast, our proposed AN allows PV to distinguish unrelated views very well and eliminate their influence before the cost volume filtering, which greatly improves the performance. In addition, when the input view number is less than 5, the accurate geometric clues from the neighboring views are actually limited. The models with AN will cast the pixels without sufficient geometric clues as noise, leading to their worse performance than the models without AN. This also shows that when the effective geometric cues are greatly increased, the performance of the models with AN will also increase rapidly. This explains why their performance will increase rapidly when the number of input views increases from 3 to 5.

Anti-noise Training Strategy and Pixelwise Visibility Network To further verify the effectiveness of AN+PV, the components of PVSNet are analyzed. As shown in Table 2, compared with the combination with MVSNet [53], our method combined with CIDER can benefit more. Note that, MVSNet+adaptive thin volumes (ATV) [9] is the UCSNet [9], where ATV is constructed based on the depth distribution. It can be seen that our model can generalize well to this challenging dataset with wide baseline with the help of AN+PV, no matter in the case of low-resolution or high-resolution. Moreover, combined with the ATV generated by our proposed ordinal-based high-resolution refinement, our model can infer high-resolution depth maps, and the performance of our method on this dataset can further boost.

Visualization Analysis of the Visibility Map So far, we have demonstrated that our overall PVSNet is able to eliminate the influence of unrelated views to improve the performance on DTU dataset and ETH3D high-res benchmark. We further present the visualization analysis of visibility map estimation on ETH3D high-res benchmark in Fig. 7. It can be seen that these source images contain strong variations in viewpoint. If they are treated equally in the cost volume aggregation, the depth prediction of the reference image will be quite inaccurate (cf. the depth prediction of CIDER in Fig. 7). With the help of the pixelwise visibility network, our method can reasonably identify the visible areas in the reference image with respect to different source images. For example, our proposed pixelwise visibility network indicates that most areas of the reference image are visible in source image 1 and source image 4. As for visibility map 10, 15 and 20, they indicate that only partial areas of the reference image are visible in source image 10, 15 and 20, which are denoted as the green boxes in Fig. 7. By only including these visible areas in the cost volume aggregation, our method can accurately recover the depth information of the reference image (cf. the depth prediction of PVSNet in Fig. 7).

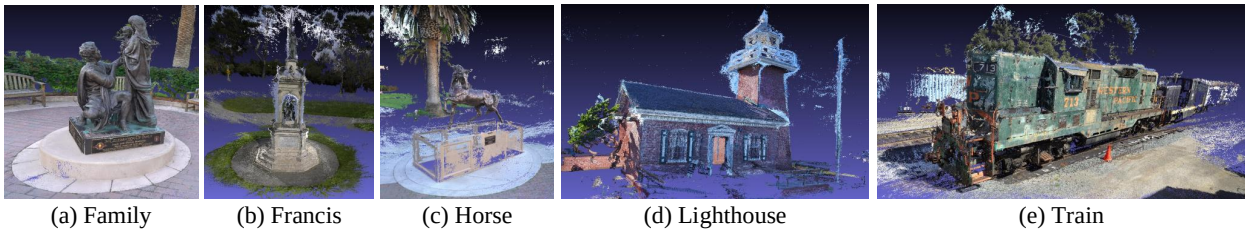
4.5 Benchmarking

For the benchmark evaluations, we use our model trained on the DTU training set without fine-tuning. We compare our method with other state-of-the-art learning-based MVS methods.

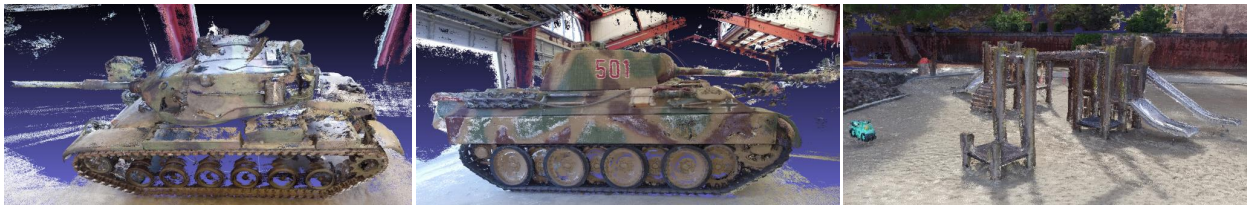
DTU Dataset [1] As illustrated in Table 3, our method PVSNet produces the best mean completeness and overall score among all methods. Besides, among the given learning-based methods with low resolution output, VisCIDER achieves the best completeness and almost the best overall score while CIDER is competitive with previous studies. Fig. 8 shows the qualitative reconstruction results of PVSNet.

Tanks and Temple Dataset [32] In Intermediate dataset, CIDER surpasses MVSNet by 3.28% and can be applied to large-scale scenes, Advanced datasets while MVSNet cannot. Although R-MVSNet is a little better than CIDER, we think that its performance advantage comes from its variation refinement post-processing instead of the network self, which can be seen from the

Fig. 8 Point cloud reconstruction on the DTU dataset.



(a) Family (b) Francis (c) Horse (d) Lighthouse (e) Train



(f) M60 (g) Panther (g) Playground

Fig. 9 Point cloud reconstruction on the Tanks and Temple Intermediate dataset.

evaluation of DTU dataset. PVSNet achieves the best results on both Intermediate dataset and Advanced dataset. This demonstrates that our PV network also improves the performance on datasets with video sequences as input. Note that, since the Advanced dataset contains stronger viewpoint variations, PVSNet has a greater performance improvement on this dataset. Fig. 9 shows the qualitative reconstruction results of PVSNet on Intermediate set.

ETH3D High-res Benchmark [43] We set image size $H \times W = 1280 \times 1920$ for this benchmark due to the GPU memory limitation. Table 5 summarizes the F_1 scores of state-of-the-art methods based on geometric and learning. Note that, the existing learning-based MVS methods have not been evaluated on this benchmark, so we evaluate these methods on the training set of ETH3D high-res benchmark. Without considering visibility estimation, the performance of existing learning-based methods [53,49,52,9,15] is very limited on this benchmark with strong viewpoint variations. Although visibility information is also introduced in VA-Point-MVSNet [7] and Vis-MVSNet [57], their performance on this benchmark is much lower than that of PVSNet. The performance of VA-Point-MVSNet [7] failed on this benchmark mainly because it uses MLP to estimate the visibility, which makes it unable to fully exploit the relationship between neighboring pixels to facilitate better visibility learning. The Vis-MVSNet [57] integrates occlusion information via the matching uncertainty estimation, which still aims to handle the MVS

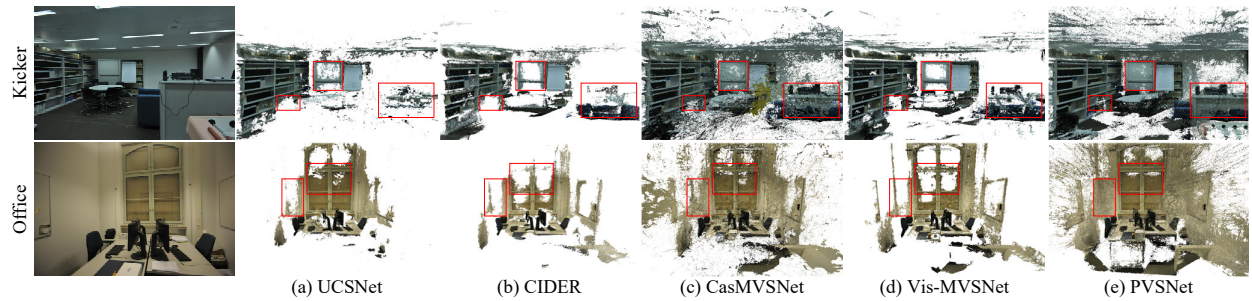


Fig. 10 Qualitative comparisons of Kicker and Office of ETH3D high-res benchmark [43].

Table 3 Quantitative results on the DTU evaluation set using the distance metric (mm) (lower is better). R-MVSNet\Ref. means R-MVSNet without variational refinement.

	Method	Acc.↓	Comp.↓	Overall↓
Geometric	Camp [5]	0.835	0.554	0.695
	Furu [13]	0.613	0.941	0.777
	Tola [45]	0.342	1.190	0.766
	Gipuma [14]	**0.283**	0.873	0.578
	Colmap [41]	0.400	0.664	0.532
LR Learning	SurfaceNet [25]	0.450	1.040	0.745
	MVSNet [53]	0.396	0.527	0.462
	R-MVSNet [54]	0.383	0.452	0.417
	R-MVSNet\Ref. [54]	0.444	0.486	0.465
	CIDER [49]	0.417	0.437	0.427
	MVSCRF [51]	0.371	0.426	0.398
	Vis-CIDER	0.408	0.393	0.400
HR Learning	P-MVSNet [34]	0.406	0.434	0.420
	Point-MVSNet [8]	0.342	0.411	0.376
	VA-Point-MVSNet [7]	0.359	0.358	0.359
	CasMVSNet [15]	0.325	0.385	0.355
	CVP-MVSNet [52]	0.296	0.406	0.351
	UCSNet [9]	0.338	0.349	0.344
	PVSNet (ours)	0.337	**0.315**	**0.326**

problems with narrow baselines, so that it can not deal with the MVS problems with wide baselines well. Fig. 10 illustrates the qualitative comparisons of reconstructions between UCSNet [9], CIDER [49], CasMVSNet [15], Vis-MVSNet [57] and ours. We can see that our method has also surpassed some traditional methods, such as MVE [12], Gipuma [14] and PMVS [13]. Moreover, our method is on par with Colmap [41]. It is worth noting that the resolution of the images tested in our network, 1280×1920, is much lower than that of Colmap, 2130×3200[1]. When we resize the input high-res images to 1280×1920 and rerun Colmap, the results in Table 5 show that the F_1 scores of our method are better than that of Colmap with low resolution input. This further demonstrates that our method can eliminate the gap between geometric-based and learning-based MVS methods on the MVS datasets with wide baselines.

[1] In fact, Colmap's test results on ETH3D high-res benchmark are obtained by resizing the high-res images to 2130×3200.

Table 4 Quantitative results on Tanks and Temples dataset using percentage metric (%) (higher is better). All the values including ours are from the website [31].

	Method	Acc.↑	Comp.↑	F_1 ↑
Intermediate	MVSNet [53]	40.23	49.70	43.48
	R-MVSNet [54]	43.74	57.60	48.60
	MVSCRF [51]	41.99	52.58	45.73
	CIDER [49]	42.79	55.21	46.76
	P-MVSNet [34]	49.93	63.82	55.62
	Point-MVSNet [8]	41.27	60.13	48.27
	VA-Point-MVSNet [7]	41.83	60.10	48.70
	CasMVSNet [15]	47.62	**74.01**	56.84
	CVP-MVSNet [52]	51.41	60.19	54.03
	UCSNet [9]	46.66	70.34	54.83
	PVSNet	**53.71**	63.88	**56.88**
Advanced	R-MVSNet [54]	**31.47**	22.05	24.91
	CIDER [49]	26.64	21.27	23.12
	CasMVSNet [15]	29.68	35.24	31.12
	PVSNet	29.43	**41.17**	**33.46**

Table 5 F_1 score (in %) comparisons of point clouds on ETH3D high-res benchmark at evaluation threshold $2cm$ (higher is better). Colmap (low) means that input images are resized to $H \times W = 1280 \times 1920$. All the values are from the website [42].

	Method	Training	Test
Geometric	MVE [12]	20.47	30.37
	Gipuma [14]	36.38	45.18
	PMVS [13]	46.06	44.16
	Colmap (low) [41]	60.54	-
	Colmap [41]	**67.66**	**73.01**
Learning	VA-Point-MVSNet [7]	18.13	-
	MVSNet [53]	25.23	-
	CIDER [49]	41.56	-
	CVP-MVSNet [52]	32.66	-
	UCSNet [9]	40.46	-
	CasMVSNet [15]	51.18	-
	Vis-MVSNet [57]	60.85	-
	Ours	**67.48**	**72.08**

5 Conclusion

In this paper, we propose a pixelwise visibility-aware multi-view stereo networks with inverse depth regression. A pixelwsie visibility-aware group-wise correlation similarity measure is designed to construct a light weight cost volume, which not only allows our network to be truly applied to datasets with strong view point changes, but also reduces the memory consumption. With the introduced anti-noise training strategy, the proposed pixelwise visibility network can be more discriminative to unrelated views. Moreover, we treat the multi-view depth inference as an inverse depth regression problem, which greatly enhances the generation of our method on unseen large-scale scenarios. The proposed ordinal-based uncertainty estimation strategy for high-resolution depth map refinement makes our method be scalable on high-resolution images and large-scale scenes. Combined with the above strategies, extensive experiments demonstrate the good applicability of our method, PVSNet, on different datasets.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grants 61772213 and 91748204.

References

1. Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120(2):153–168, 2016.
2. Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. In *ACM SIGGRAPH*, pages 24:1–24:11, 2009.
3. Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, volume 11, pages 1–11, 2011.
4. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, Nov 2001.
5. Neill D. F. Campbell, George Vogiatzis, Carlos Hernández, and Roberto Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *Proceedings of the European Conference on Computer Vision*, pages 766–779, 2008.
6. J. Chang and Y. Chen. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5418, 2018.
7. R. Chen, S. Han, J. Xu, and h. su. Visibility-aware point-based multi-view stereo network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2020.
8. Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1538–1547, 2019.
9. Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.
10. R. T. Collins. A space-sweep approach to true multi-image matching. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 358–363, 1996.
11. Z. Fu and M. Ardabilian Fard. Learning confidence measures by multi-modal convolutional neural networks. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision*, pages 1321–1330, 2018.
12. Simon Fuhrmann, Fabian Langguth, and Michael Goesele. Mve: A multi-view reconstruction environment. In *Proceedings of the Eurographics Workshop on Graphics and Cultural Heritage*, page 11–18, 2014.
13. Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010.
14. S. Galliani, K. Lasinger, and K. Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 873–881, 2015.
15. Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.
16. Xiaoyang Guo, Kai Yang, Wukui Yang, Xiaogang Wang, and Hongsheng Li. Group-wise correlation stereo network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3273–3282, 2019.
17. Norbert Haala and Mathias Rothermel. Dense multi-stereo matching for high quality digital elevation models. *Photogrammetrie-Fernerkundung-Geoinformation*, 2012(4):331–343, 2012.
18. W. Hartmann, S. Galliani, M. Havlena, L. V. Gool, and K. Schindler. Learned multi-patch similarity. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1595–1603, 2017.
19. Philipp Heise, Brian Jensen, Sebastian Klose, and Alois Knoll. Variational patchmatch multiview reconstruction and refinement. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 882–890, 2015.
20. H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2008.
21. A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511, 2013.
22. X. Hu and P. Mordohai. A quantitative evaluation of confidence measures for stereo vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2121–2133, 2012.
23. P. Huang, K. Matzen, J. Kopf, N. Ahuja, and J. Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2821–2830, June 2018.

24. Sunghoon Im, Hae-Gon Jeon, Stephen Lin, and In So Kweon. Dpsnet: End-to-end deep plane sweep stereo. *arXiv preprint arXiv:1905.00538*, 2019.
25. Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2307–2315, 2017.
26. Abhishek Kar, Christian Häne, and Jitendra Malik. Learning a multi-view stereo machine. In *Advances in Neural Information Processing Systems*, pages 365–376. 2017.
27. Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Trans. Graph.*, 32(3):29:1–29:13, July 2013.
28. A. Kendall, H. Martirosyan, S. Dasgupta, and P. Henry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 66–75, 2017.
29. Sunok Kim, Seungryong Kim, Dongbo Min, and Kwanghoon Sohn. Laf-net: Locally adaptive fusion networks for stereo confidence estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 205–214, 2019.
30. S. Kim, D. Min, S. Kim, and K. Sohn. Unified confidence estimation networks for robust stereo matching. *IEEE Transactions on Image Processing*, 28(3):1299–1313, 2019.
31. Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and Temples Benchmark. https://www.tanksandtemples.org, 2017.
32. Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Trans. Graph.*, 36(4):78:1–78:13, 2017.
33. Vladimir Kolmogorov and Ramin Zabih. Multi-camera scene reconstruction via graph cuts. In *Proceedings of the European Conference on Computer Vision*, pages 82–96, 2002.
34. Keyang Luo, Tao Guan, Lili Ju, Haipeng Huang, and Yawei Luo. P-mvsnet: Learning patch-wise matching confidence aggregation for multi-view stereo. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 10452–10461, 2019.
35. Keyang Luo, Tao Guan, Lili Ju, Yuesong Wang, Zhuo Chen, and Yawei Luo. Attention-aware multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1590–1599, 2020.
36. W. Luo, A. G. Schwing, and R. Urtasun. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5695–5703, 2016.
37. N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016.
38. M. Poggi, F. Tosi, and S. Mattoccia. Quantitative evaluation of confidence measures in a machine learning world. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5238–5247, 2017.
39. Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pages 234–241, 2015.
40. J. L. Schönberger and J. Frahm. Structure-from-motion revisited. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.
41. Johannes L. Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision*, pages 501–518, 2016.
42. Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. ETH3D Benchmark. https://www.eth3d.net.
43. T. Schöps, J. L. Schönberger, S. Galliani, T. Sattler, K. Schindler, M. Pollefeys, and A. Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2538–2547, 2017.
44. Akihito Seki and Marc Pollefeys. Patch based confidence prediction for dense disparity map. In *Proceedings of the British Machine Vision Conference*, pages 23.1–23.13, 2016.
45. Engin Tola, Christoph Strecha, and Pascal Fua. Efficient large-scale multi-view stereo for ultra high-resolution image sets. *Machine Vision and Applications*, 23(5):903–920, 2012.
46. Stepan Tulyakov, Anton Ivanov, and François Fleuret. Practical deep stereo (pds): Toward applications-friendly deep stereo matching. In *Advances in Neural Information Processing Systems*, pages 5871–5881. 2018.
47. J. Žbontar and Y. LeCun. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1592–1599, 2015.
48. Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5483–5492, 2019.
49. Qingshan Xu and Wenbing Tao. Learning inverse depth regression for multi-view stereo with correlation cost volume. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
50. Qingshan Xu and Wenbing Tao. Planar prior assisted patchmatch multi-view stereo. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
51. Youze Xue, Jiansheng Chen, Weitao Wan, Yiqing Huang, Cheng Yu, Tianpeng Li, and Jiayu Bao. Mvscrf: Learning multi-view stereo with conditional random fields. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4312–4321, 2019.
52. Jiayu Yang, Wei Mao, Jose M. Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020.
53. Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision*, pages 767–783, 2018.
54. Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5525–5534, 2019.
55. Zehao Yu and Shenghua Gao. Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1949–1958, 2020.

56. Feihu Zhang, Victor Prisacariu, Ruigang Yang, and Philip HS Torr. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 185–194, 2019.
57. Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. *arXiv preprint arXiv:2008.07928*, 2020.
58. E. Zheng, E. Dunn, V. Jojic, and J. M. Frahm. Patchmatch based joint view selection and depthmap estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1517, 2014.