

Learning Inverse Depth Regression for Multi-View Stereo with Correlation Cost Volume

Qingshan Xu, Wenbing Tao*

National Key Laboratory of Science and Technology on Multispectral Information Processing
School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, China
{qingshanxu, wenbingtao}@hust.edu.cn

Abstract

Deep learning has shown to be effective for depth inference in multi-view stereo (MVS). However, the scalability and accuracy still remain an open problem in this domain. This can be attributed to the memory-consuming cost volume representation and inappropriate depth inference. Inspired by the group-wise correlation in stereo matching, we propose an average group-wise correlation similarity measure to construct a lightweight cost volume. This can not only reduce the memory consumption but also reduce the computational burden in the cost volume filtering. Based on our effective cost volume representation, we propose a cascade 3D U-Net module to regularize the cost volume to further boost the performance. Unlike the 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. Through extensive experiments on DTU dataset and Tanks and Temples dataset, we show that our proposed network with Correlation cost volume and Inverse DEpth Regression (CIDER¹), achieves state-of-the-art results, demonstrating its superior performance on scalability and accuracy.

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 always follows a two-stage pipeline: depth map estimation and fusion. Of these two stages, depth map estimation plays an important role in the whole pipeline and many MVS methods (Zheng et al. 2014; Galliani, Lasinger, and Schindler 2015; Schönberger et al. 2016; Xu and Tao 2019; Huang et al. 2018; Yao et al. 2018; 2019) 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 mea-

suring the similarity between these pixels. Traditional methods (Galliani, Lasinger, and Schindler 2015; Schönberger et al. 2016; Xu and Tao 2019) 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 (Kolmogorov and Zabih 2002; Hosni et al. 2013) resort to regularization technologies, such as graph-cuts and cost filtering. However, these engineered regularization methods still struggle in the above challenging areas.

To overcome the above difficulties, recent works (Yao et al. 2018; 2019; Huang et al. 2018) leverage deep convolutional neural networks (DCNN) to learn multi-view depth inference. Thanks to their powerful cost volume representation and filtering, these works achieve comparable results to the traditional state-of-the-arts. However, because the cost volume representation is proportional to the model resolution, it always requires taking up a lot of memory (Huang et al. 2018; Yao et al. 2018). This greatly limits the use of these methods in large-scale and high-resolution scenarios. To alleviate this problem, R-MVSNet (Yao et al. 2019) utilizes the gated recurrent unit (GRU) to sequentially regularize 2D cost maps along the depth direction. Although this dramatically reduces the memory consumption, it cannot incorporate enough context information over the cost volume like the 3D U-Net in MVSNet (Yao et al. 2018). As a result, the performance of the network self degrades and a traditional variational refinement is required.

Inspired by the group-wise correlation in stereo matching (Guo et al. 2019), we propose an average group-wise correlation similarity measure to construct a lightweight cost volume representation. Specifically, we first extract the deep features for the reference image and source images. With a differential warping module, we compute compact similarity scores between every source image and the reference image using the group-wise correlation. To aggregate an arbitrary number of neighboring image information, we average the similarity scores of all source images to produce a unified cost feature in the cost volume. In this way, our network can greatly reduce the memory consumption and simultaneously fit in an arbitrary number of neighboring images. Based on

*Corresponding author

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹Code will be available at <https://github.com/GhiXu/CIDER>.

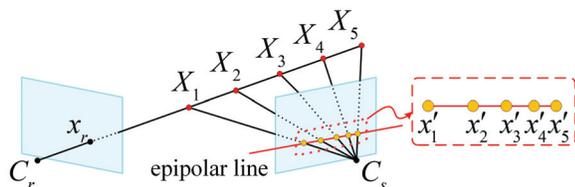


Figure 1: Depth sample and regression. Depth hypotheses are uniformly sampled in depth space. When these uniformly distributed depth hypotheses are projected to a source image, their corresponding 2D points are not distributed uniformly along the epipolar line.

this effective cost volume representation, we present a cascade 3D U-Net to learn more context information. This further boosts the performance of our network.

Additionally, existing multi-view depth inference networks (Huang et al. 2018; Yao et al. 2018; 2019) usually cast this task as a depth regression problem or an inverse depth classification problem. It is obvious that the inverse depth classification always introduce stair effects. Thus, DeepMVS (Huang et al. 2018) applies the Fully-Connected Conditional Random Field (DenseCRF) (Krähenbühl and Koltun 2011) and R-MVSNet (Yao et al. 2019) utilizes variational refinement to refine their raw predictions. As for the depth regression, this strategy uniformly samples depth hypotheses in depth space and achieves sub-pixel estimation. However, it is not robust. As shown in Figure 1, although the sampled depth hypotheses are uniformly distributed in depth space, their projected 2D points in a source image are not distributed uniformly along the epipolar line. Consequently, the true depth hypothesis near the camera center may not be captured by the deep features in a source image. On the other hand, when the true depth hypothesis is far away from the camera center, some depth hypotheses around it will correspond to multiple similar deep features in a source image because these deep features may be sampled from almost the same position. This confuses the true depth hypothesis.

To achieve robust sub-pixel estimation, we 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. The inverse depth hypotheses are used to obtain the cost map slice in the cost volume while the ordinals are employed to regress the sub-pixel ordinal. The obtained sub-pixel ordinal is further converted to the final depth value, which is utilized to guide the training of our network. The inverse depth regression enables our network to be applied in large-scale scenes.

With the above proposed strategies, our network achieves promising reconstruction results on DTU dataset (Aanaes et al. 2016) and Tanks and Temples dataset (Knapitsch et al. 2017). Our contributions are three-fold.

- We propose an average group-wise correlation similarity measure to construct a lightweight cost volume. This greatly eases the memory burden of our network.
- We present a cascade 3D U-Net to incorporate more context information to boost the performance of our network.

- 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.

Related Work

Our proposed method is closely related to some learning-based works in stereo matching and multi-view stereo. We briefly review these works in the following.

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 (Žbontar and LeCun 2015) 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. (Luo, Schwing, and Urtasun 2016) propose an 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 (Mayer et al. 2016) is proposed with an encoder-decoder architecture. Kendall et al. (Kendall et al. 2017) 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 (Chang and Chen 2018) employ a spatial pyramid pooling module to incorporate global context information and use a stacked hourglass architecture to learn more context information. Tulyakov et al. (Tulyakov, Ivanov, and Fleuret 2018) compress the concatenated left-right image descriptors into compact matching signatures to decrease the memory footprint. Guo et al. (Guo et al. 2019) 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. (Hartmann et al. 2017) utilize an n-way Siamese network architecture and the mean operation to learn a multi-patch similarity metric. (Ji et al. 2017) proposes to encode camera parameters together with images in a 3D voxel representation to learn the 3D model of a scene in an end-to-end manner. Kar et al. (Kar, Häne, and Malik 2017) leverage perspective geometry to construct feature grids for each image and fuse these feature grids into a single grid with a grid recurrent fusion module. Through the 3D grid reasoning, a voxel in the 3D grid can be judged whether it belongs to the surface. DeepMVS (Huang et al. 2018) adopts plane-sweeping to sample image patches and constructs a cost volume for one source image. Then an intra-volume feature aggregation is utilized to perceive non-local information. To tackle an arbitrary number of input images, max-pooling is used to gather the information from neighboring images. DeepMVS poses depth estimation as a multi-label classification problem and thus the DenseCRF

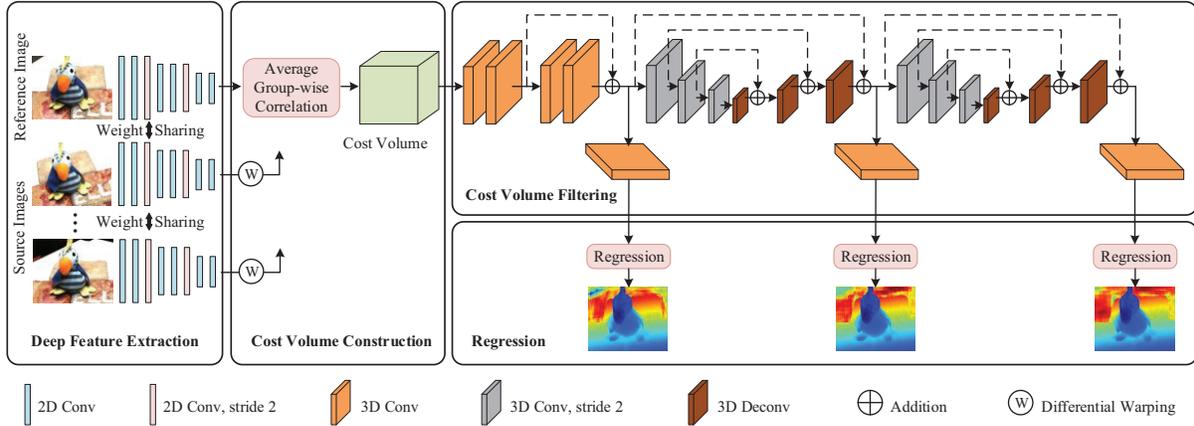


Figure 2: Network architecture of our proposed CIDER. Feature maps are extracted via a weigh-sharing deep feature extraction module for a reference image and source images. The feature maps of source images are warped into the coordinate of the reference image by differential warping. All features are fed to an average group-wise correlation module to construct a lightweight cost volume. The predicted depth maps are obtained by imposing cost volume filtering and regression on the cost volume.

is needed to refine the raw predictions. MVSNet (Yao et al. 2018) employs a differentiable homography warping to explicitly encode camera geometries in the network and builds the cost volume upon the reference camera frustum. The predictions are regressed from the depth cost volume. To resolve the scalability issue in MVSNet, R-MVSNet (Yao et al. 2019) leverages a gated recurrent unit to sequentially regularize the cost volume. As R-MVSNet also casts the problem as an inverse depth classification task, a variational refinement is applied to gain more accurate estimation.

Method

Our proposed network takes as input a reference image $I_{\text{ref}} = I_0$ and source images $I_{\text{src}} = \{I_i | i = 1 \dots N - 1\}$ 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 Figure 2, our proposed network, CIDER, consists of four modules: deep feature extraction, cost volume construction, cost volume filtering and regression. The detailed parameters of each module are listed in Table 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, making the depth estimation in these regions failed. Instead, we adopt the multi-scale deep feature extraction network used in MVSNet (Yao et al. 2018) to incorporate context information. In this way, for each input image $3 \times H \times W$, a multi-scale deep feature $32 \times \frac{H}{4} \times \frac{W}{4}$ can be obtained, where H and W are the input image height and width.

Correlation Cost Volume Construction

After getting the deep features for all input images, we hope to encode these features together with the camera pa-

rameters into the network to enable its geometry awareness. Inspired by the traditional plane sweep stereo (Collins 1996), 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

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

where D is the total sample number of depth values, \mathbf{K}_{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 $\mathcal{F}_{\text{src}} = \{\mathcal{F}_i | i = 1 \dots N - 1\}$ can be warped into the coordinate of the deep feature of the reference image \mathcal{F}_{ref} . The warped deep features of all source images at depth d_j are denoted as $\tilde{\mathcal{F}}_{\text{src},j} = \{\tilde{\mathcal{F}}_{i,j} | i = 1 \dots N - 1\}$.

In order to measure the multi-view feature similarity, MVSNet (Yao et al. 2018) 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 (Yao et al. 2018), 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 (Tulyakov, Ivanov, and Fleuret 2018) 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 (Guo et al. 2019), we propose an average group-wise correlation similarity measure to construct a lightweight cost volume. Specifically, for the deep reference image feature \mathcal{F}_{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}_{ref} and $\tilde{\mathcal{F}}_{i,j}$ is computed as

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

where $g = 0 \dots G-1$, $\mathcal{F}_{\text{ref}}^g$ is the g -th feature of \mathcal{F}_{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}_{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 cost volume V_i of size $G \times \frac{H}{4} \times \frac{W}{4} \times D$. In order to adapt an arbitrary number of input source images, the individual cost volumes for different source images are averaged to compute the following final multi-view cost volume:

$$V = \frac{1}{N-1} \sum_{i=1}^{N-1} V_i. \quad (3)$$

Note that, the size of this multi-view cost volume is also $G \times \frac{H}{4} \times \frac{W}{4} \times D$. As aforementioned, a lightweight raw 8-channel cost volume can be obtained by setting $G = 8$. This can greatly reduce the memory consumption. Also, this cost volume representation can ease the computation burden of the subsequent cost volume filtering module.

Cost Volume Filtering

As pointed out in (Kendall et al. 2017), 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 (Chang and Chen 2018; Guo et al. 2019; Zhang et al. 2019) 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 (Guo et al. 2019) 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 Figure 2 and Table 1. Note that, the previous MVS networks

Table 1: The detailed parameters of the proposed CIDER network. If not specified, each 2D/3D convolution/deconvolution layer is followed by a batch normalization (BN) and a rectified linear unit(ReLU). S1/2 denotes the convolution stride. * denotes no BN and ReLU. * denotes no ReLU.

Index	Layer Description	Output Size
Input Images		H×W×3
Deep Feature Extraction		
1	Conv2D, 3×3, S1, 8	H×W×8
2	Conv2D, 3×3, S1, 8	H×W×8
3	Conv2D, 5×5, S2, 16	½H×½W×16
4	Conv2D, 3×3, S1, 16	½H×½W×16
5	Conv2D, 3×3, S1, 16	½H×½W×16
6	Conv2D, 5×5, S2, 32	¼H×¼W×32
7	Conv2D, 3×3, S1, 32	¼H×¼W×32
8	Conv2D*, 3×3, S1, 32	¼H×¼W×32
Correlation Cost Volume Construction		
Differential Warping and Average Group-wise Correlation		D×¼H×¼W×8
Cost Volume Filtering		
9	Conv3D 3×3×3, S1, 8	D×¼H×¼W×8
10	Conv3D 3×3×3, S1, 8	D×¼H×¼W×8
11	Conv3D 3×3×3, S1, 8	D×¼H×¼W×8
12	Conv3D* 3×3×3, S1, 8 Add the output of 10	D×¼H×¼W×8
13	Conv3D 3×3×3, S2, 16	½D×½H×½W×16
14	Conv3D 3×3×3, S2, 32	¼D×¼H×¼W×32
15	Conv3D 3×3×3, S2, 64	¼D×½H×½W×64
16	Deconv3D 3×3×3, S2, 32 Add the output of 14	¼D×¼H×¼W×32
17	Deconv3D 3×3×3, S2, 16 Add the output of 13	½D×½H×½W×16
18	Deconv3D 3×3×3, S2, 8 Add the output of 12	D×¼H×¼W×8
19-24	Repeat 13-18	D×¼H×¼W×8
Regression		
25	From the output of 12 Conv3D* 3×3×3, S1, 1 Regression	D×¼H×¼W×1 ¼H×¼W
26	From the output of 18 Conv3D* 3×3×3, S1, 1 Regression	D×¼H×¼W×1 ¼H×¼W
27	From the output of 24 Conv3D* 3×3×3, S1, 1 Regression	D×¼H×¼W×1 ¼H×¼W

never employ this structure due to their large memory consumption caused by the huge cost volume representation, e.g., MVSNet (Yao et al. 2018) and R-MVSNet (Yao et al. 2019). 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.

Inverse Depth Regression

In order to achieve the sub-pixel estimation, (Kendall et al. 2017) 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 (Figure 1).

As described in the Correlation Cost Volume Construction, 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,

$$d_j = \left(\left(\frac{1}{d_{min}} - \frac{1}{d_{max}} \right) \frac{j}{D-1} + \frac{1}{d_{max}} \right)^{-1}, j = 0 \dots D-1, \quad (4)$$

where d_{min} and d_{max} are the minimal depth value and the maximal depth value of the reference image. 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 Figure 2, there are three output branches in our network. In each branch, the filtered cost volume is converted to a 1-channel cost volume via a 3D convolution operation. To obtain the final continuous depth map, we first regress the sub-pixel ordinal k from the cost volume as follows,

$$k = \sum_{j=0}^{D-1} j \times p_j, \quad (5)$$

where p_j is the probability at depth value d_j , which is computed from the predicted cost volume via the softmax function. The final predicted depth value for each pixel is computed as

$$\hat{d} = \left(\left(\frac{1}{d_{min}} - \frac{1}{d_{max}} \right) \frac{k}{D-1} + \frac{1}{d_{max}} \right)^{-1}. \quad (6)$$

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

$$L = \sum_{q=0}^2 \lambda_q l(\mathbf{d}, \hat{\mathbf{d}}_q), \quad (7)$$

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

Experiments

In this section, we evaluate our proposed network on DTU dataset (Aanæs et al. 2016) and Tanks and Temples dataset (Knapitsch et al. 2017). First, we describe the datasets and evaluation metrics followed by implementation details. Then, we perform ablation studies using Tanks and Temple dataset. Last, we show the benchmarking results on the above datasets.

Datasets and Evaluation Metrics

DTU Dataset (Aanæs et al. 2016) This dataset contains more than 100 object-centric scenes. 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 (Knapitsch et al. 2017) 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.

Evaluation Metrics As suggested in different datasets, the accuracy and the completeness of the distance metric are used for DTU dataset while the accuracy and the completeness of the percentage metric for Tanks and Temple dataset. In order to obtain a summary measure for the accuracy and the completeness, the mean value of them is employed for the distance metric and the F_1 score is utilized for the percentage metric.

Implementation Details

Training Following (Ji et al. 2017), 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 (Yao et al. 2018) to generate the depth maps at a resolution of 160×128 by leveraging the screened Poisson surface reconstruction (Kazhdan and Hoppe 2013). During the training, the image size is scaled and cropped to 640×512 and the total number of input image is set to $N = 3$. d_{min} and d_{max} are fixed to $425mm$ and $935mm$ respectively. The total sample number of depth values is set to $D = 192$. The weights for three outputs are set to $\lambda_0 = 0.5$, $\lambda_1 = 0.5$ and $\lambda_2 = 0.7$. We implement our network by using PyTorch. The network is trained for 10 epoch in total on a TITAN X GPU. 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.

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 (Galliani, Lasinger, and Schindler 2015; Schönberger et al. 2016; Yao et al. 2018; Xu and Tao 2019). 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

Table 2: Ablation study results of proposed networks on Tanks and Temples training datasets (Knapitsch et al. 2017) 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	Caterp.	Church	Court.	Ignatius	Meeting.	Truck	Mean	GPU Mem.	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

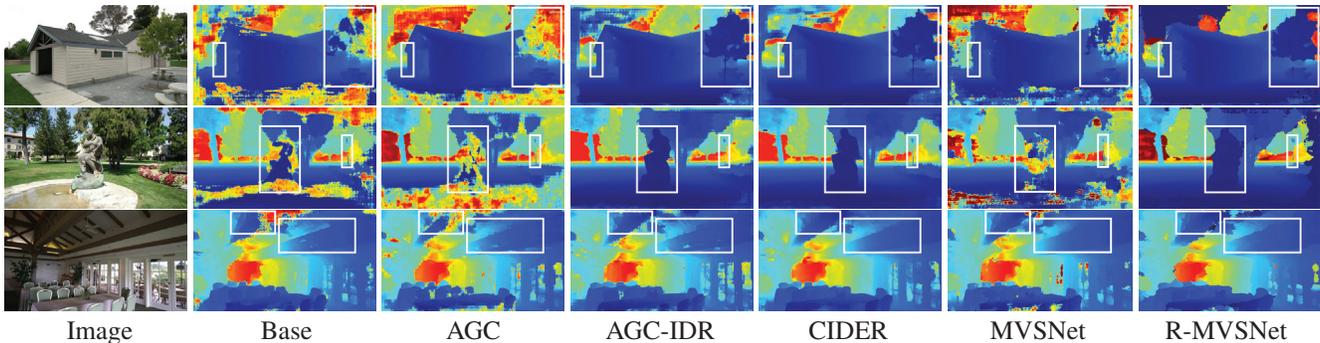


Figure 3: Depth map reconstructions of Barn, Ignatius and Meetingroom, Tanks and Temples training datasets (Knapitsch et al. 2017) using different settings of the proposed networks.

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.

Ablation Studies

In this section, we explore the effectiveness of our proposed strategies, 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 (Yao et al. 2018). In order to simultaneously show the generalization of different models on unseen datasets, we use Tanks and Temples training datasets here to conduct experiments. The camera poses are obtained by COLMAP (Schönberger and Frahm 2016). The image resolution is resized to 1920×1056 as (Yao et al. 2018; 2019) does. The depth sampling number is set to $D = 192$ and the input view number is $N = 5^2$ for all models.

²The discussion of N is shown in supplementary materials.

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 $32 \times \frac{H}{4} \times \frac{W}{4} \times D$ to $8 \times \frac{H}{4} \times \frac{W}{4} \times D$. The results are shown in Table 2 and Figure 3. 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 information well but also achieve a compact cost volume representation.

Inverse Depth Regression It can be seen from Table 2 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. As shown in Table 2, this model outperforms the previous two models with a significant margin. Moreover, according to the visualization of the reconstructed depth maps in Figure 3, 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

Table 3: Quantitative results on the DTU evaluation test (Aanæs et al. 2016) using the distance metric (*mm*). R-MVSNet\Ref. means R-MVSNet without variational refinement.

Method	Mean Acc.	Mean Comp.	Overall
SurfaceNet	0.450	1.04	0.745
MVSNet	0.396	0.527	0.462
R-MVSNet\Ref.	0.444	0.486	0.465
CIDER	0.417	0.437	0.427
R-MVSNet	0.385	0.459	0.422

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 Figure 3, 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 Figure 3, 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 2. 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 2, 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 our network with other state of the art learning-based MVS methods on different benchmarks.

Comparison with Existing Methods We also compare our method with MVSNet (Yao et al. 2018) and R-MVSNet without variational refinement (R-MVSNet\Ref.) (Yao et al. 2019). Table 2 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 Figure 3. Thus, our method is also better than R-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.

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, including SurfaceNet (Ji et al. 2017), MVSNet (Yao et al. 2018) and R-MVSNet (Yao et al. 2019).

DTU Dataset (Aanæs et al. 2016) As illustrated in Table 3, our method produces the best mean completeness and overall score among all methods without post-processing. Figure 4 shows the qualitative results of our reconstructions.

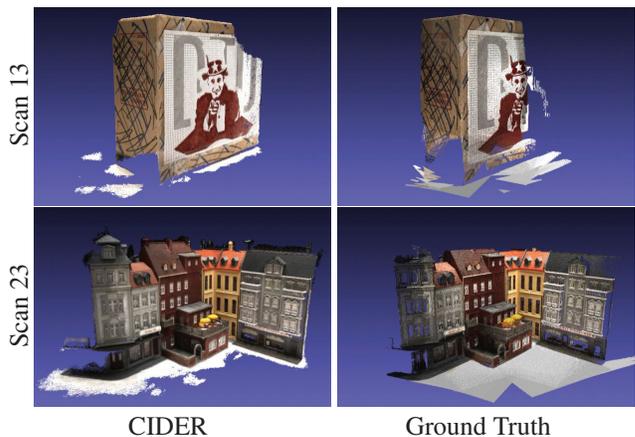


Figure 4: Our reconstructed point clouds and ground truth on scan 13 and scan 23 of DTU evaluation set (Aanæs et al. 2016).

Table 4: Quantitative results on the Tanks and Temples Intermediate dataset and Advanced dataset (Aanæs et al. 2016) using the percentage metric (%).

Dataset	Method	Acc.	Comp.	F_1
Intermediate	MVSNet	40.23	49.70	43.48
	R-MVSNet	43.74	57.60	48.40
	CIDER	42.79	55.21	46.76
Advanced	MVSNet	-	-	-
	R-MVSNet	31.47	22.05	24.91
	CIDER	26.64	21.27	23.12

Note that, the performance of R-MVSNet gains a lot with variational refinement.

Tanks and Temple Dataset (Knapitsch et al. 2017) In Intermediate dataset, our method 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 our method, we think that its performance advantage comes from its variation refinement post-processing instead of the network self, which can be seen from the evaluation of DTU dataset. See supplementary materials for reconstruction results.

Conclusion

In this paper, we propose a learning-based multi-view stereo method with correlation cost volume. The correlation cost volume is lightweight due to our proposed average group-wise correlation similarity measure. This reduces the memory consumption and makes our method be scalable on high-resolution images. Moreover, we treat the multi-view depth inference as an inverse depth regression problem. This greatly enhances the generation of our method on unseen large-scale scenarios. We also present a cascade 3D U-Net filtering to improve the accuracy on ambiguous areas. Combined with the above strategies, extensive experiments demonstrate the good applicability of our method, CIDER, on different datasets.

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

References

- 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* 120(2):153–168.
- Chang, J., and Chen, Y. 2018. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5410–5418.
- Collins, R. T. 1996. A space-sweep approach to true multi-image matching. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 358–363.
- Galliani, S.; Lasinger, K.; and Schindler, K. 2015. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, 873–881.
- Guo, X.; Yang, K.; Yang, W.; Wang, X.; and Li, H. 2019. Group-wise correlation stereo network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3273–3282.
- Hartmann, W.; Galliani, S.; Havlena, M.; Gool, L. V.; and Schindler, K. 2017. Learned multi-patch similarity. In *Proceedings of the IEEE International Conference on Computer Vision*, 1595–1603.
- Hosni, A.; Rhemann, C.; Bleyer, M.; Rother, C.; and Gelautz, M. 2013. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35(2):504–511.
- Huang, P.; Matzen, K.; Kopf, J.; Ahuja, N.; and Huang, J. 2018. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2821–2830.
- Ji, M.; Gall, J.; Zheng, H.; Liu, Y.; and Fang, L. 2017. Surfacer-net: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision*, 2307–2315.
- Kar, A.; Häne, C.; and Malik, J. 2017. Learning a multi-view stereo machine. In *Advances in Neural Information Processing Systems*. 365–376.
- Kazhdan, M., and Hoppe, H. 2013. Screened poisson surface reconstruction. *ACM Trans. Graph.* 32(3):29:1–29:13.
- Kendall, A.; Martirosyan, H.; Dasgupta, S.; and Henry, P. 2017. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision*, 66–75.
- Knapitsch, A.; Park, J.; Zhou, Q.-Y.; and Koltun, V. 2017. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Trans. Graph.* 36(4):78:1–78:13.
- Kolmogorov, V., and Zabih, R. 2002. Multi-camera scene reconstruction via graph cuts. In *Proceedings of the European Conference on Computer Vision*, 82–96.
- Krähenbühl, P., and Koltun, V. 2011. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in Neural Information Processing Systems*. 109–117.
- Luo, W.; Schwing, A. G.; and Urtasun, R. 2016. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5695–5703.
- Mayer, N.; Ilg, E.; Häusser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; and Brox, T. 2016. 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*, 4040–4048.
- Schönberger, J. L., and Frahm, J. 2016. Structure-from-motion revisited. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 4104–4113.
- Schönberger, J. L.; Zheng, E.; Frahm, J.-M.; and Pollefeys, M. 2016. Pixelwise view selection for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision*, 501–518.
- Tulyakov, S.; Ivanov, A.; and Fleuret, F. 2018. Practical deep stereo (pds): Toward applications-friendly deep stereo matching. In *Advances in Neural Information Processing Systems*. 5871–5881.
- Žbontar, J., and LeCun, Y. 2015. Computing the stereo matching cost with a convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1592–1599.
- Xu, Q., and Tao, W. 2019. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5483–5492.
- Yao, Y.; Luo, Z.; Li, S.; Fang, T.; and Quan, L. 2018. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision*, 767–783.
- Yao, Y.; Luo, Z.; Li, S.; Shen, T.; Fang, T.; and Quan, L. 2019. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5525–5534.
- Zhang, F.; Prisacariu, V.; Yang, R.; and Torr, P. H. 2019. Ganet: Guided aggregation net for end-to-end stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 185–194.
- Zheng, E.; Dunn, E.; Jovic, V.; and Frahm, J. M. 2014. Patch-match based joint view selection and depthmap estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1510–1517.