Unstructured Multi-view Depth Estimation Using Mask-Based Multiplane Representation

Yuxin Hou$^{(\boxtimes)}$ [iD], Arno Solin [iD], and Juho Kannala [iD]

Department of Computer Science, Aalto University, Espoo, Finland
{yuxin.hou,arno.solin,juho.kannala}@aalto.fi

Abstract. This paper presents a novel method, MaskMVS, to solve depth estimation for unstructured multi-view image-pose pairs. In the plane-sweep procedure, the depth planes are sampled by histogram matching that ensures covering the depth range of interest. Unlike other plane-sweep methods, we do not rely on a cost metric to explicitly build the cost volume, but instead infer a multiplane mask representation which regularizes the learning. Compared to many previous approaches, we show that our method is lightweight and generalizes well without requiring excessive training. We outperform the current state-of-the-art and show results on the SUN3D, SCENES11, MVS, and RGBD test data sets.

Keywords: Computer vision · Depth estimation · Multi-view stereo

1 Introduction

Multi-view stereo (MVS) aims at reconstructing depth (or disparity) maps from a collection of overlapping images, which is a fundamental problem in computer vision. Any progress in the field will have a direct impact on applications like augmented reality and self-driving cars. Conventional methods often use hand-crafted features and compute similarity between patches. However, these approaches may suffer from limitations of the features, especially regarding poorly textured or reflective regions. As deep convolutional neural networks (CNNs) have shown great success in many vision tasks such as image classification, it has triggered the interest to overcome the weakness of traditional methods and improve 3D reconstruction using deep models.

There are already several works that approach two-view stereo using deep models with successful results (*e.g.*, [10,13]). However, rigid two-view stereo is a simpler problem than unstructured multi-view stereo, where camera motion can be arbitrary and varying. Yet, unstructured multi-view stereo is a highly relevant problem that appears in the context of depth estimation from moving monocular cameras. In fact, there are already robust and accurate approaches for real-time tracking of motion (*e.g.*, commercially deployed solutions such as
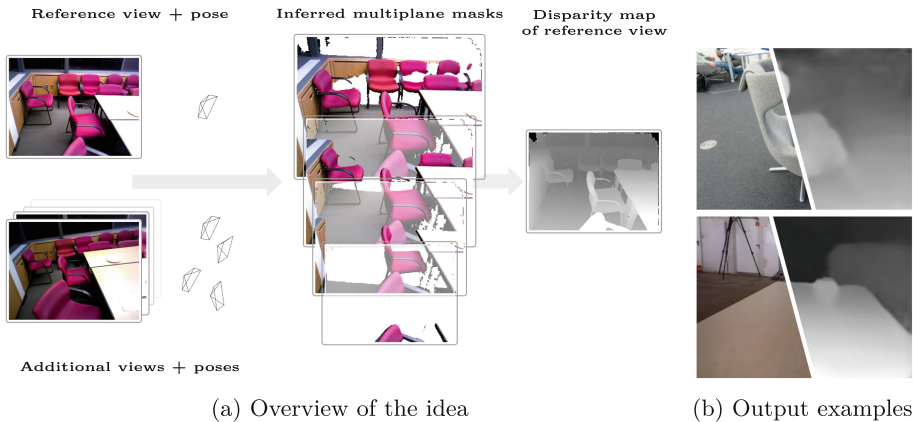
Reference view + pose   Inferred multiplane masks   Disparity map of reference view

Additional views + poses

(a) Overview of the idea   (b) Output examples

Fig. 1. (a) Overall idea of our method: To estimate a depth map with an arbitrary number of images and known camera poses, we back-project images onto a set of planes to generate the multiplane masks representation via a convolutional neural network. The inferred masks are then passed through a second network to reconstruct the final disparity map. (b) Our method performs well in textureless and varying depth cases.

ARCore by Google and ARKit by Apple) but depth estimation remains a challenge. For example, some smartphones nowadays contain stereo cameras but the small baseline due to the size restriction of the device is a limitation for long-range depth estimation. Thus, multi-view depth estimation is helpful also in such cases since the additional baseline arising from motion can alleviate some of the problems.

Recently, various deep learning based multi-view depth estimation methods have been proposed, *e.g.* [6,17,21]. They typically discretize the depth space and utilize a plane-sweep approach to compute a matching cost volume from which the disparity map is inferred via CNNs. The benefit is that the cost volume based approach force the network to learn disparity estimation via matching instead of just learning the single-view cues, which is beneficial for generalization. However, these methods have also problems: the depth range must be approximately known in advance and discretization poses an inherent trade-off between depth resolution and computational complexity. In addition, manually specified features and metrics are often used in the construction of the cost volume [17,21] or the used networks are large and complex hampering computation speed [6].

In this paper, we propose our own plane-sweep based approach, which aims at avoiding some of the shortcomings of the previous methods. In particular, our method does not use manually specified features or cost metrics but instead infers a set of masking planes to regularize the learning of features (Fig. 1). In addition, we propose selecting intermediate depth planes by depth histogram matching if the depth range of interest is approximately known *a priori*. In comparison to recent approaches like [6], our architecture is relatively simple, lightweight, and more accurate.

In summary, the contributions of this paper are: *(i)* We propose a CNN-based approach for multi-view stereo depth estimation that does not require constructing an explicit cost-volume metric; *(ii)* We propose a way of selecting intermediate depth-planes by depth histogram matching; *(iii)* We demonstrate that the current state-of-the-art in CNN-based MVS can be matched with a relatively simple and lightweight architecture. This paper is structured as follows. Sect. 2 goes through the background and covers related approaches. Sect. 3 presents our MaskMVS method in detail. Experiments and ablation studies are presented in Sect. 4, and this paper is concluded with a discussion in Sect. 5.

2   Related Work

Multi-view stereo reconstruction has been under active research for long and only recently it has gotten a boost from CNN-based methods. Conventional MVS algorithms typically seek to design photometric error measures and solve an optimization task subject to penalizing visual inconsistency (see review in [3]). The most prominent traditional method is COLMAP [14] which jointly estimates depths and surface normals by leveraging photometric and geometric priors. While conventional MVS methods deliver impressive results in the best case, they fail in poorly textured regions where the photometric consistency is not reliable. They also cannot use visual cues for depth such as shadows or lighting, and typically require many frames as input.

The success of convolutional neural networks in computer vision has spawned a number of methods that leverage the capability of CNNs to learn visual cues. The extreme case is purely monocular depth estimation [1,12], while left–right stereo reconstruction [10,13] relies on the 1D correlation layer along the disparity line. Other approaches use nearby images as the supervision by warping and computing image reconstruction error [4,20,22], but the CNN-based prediction still only utilize single view information.

Unlike left–right two-view stereo, images collected from monocular videos are more unstructured and they also suffer from dynamic moving objects, which makes the task more challenging. DeMoN [16] can learn depth and motion for unconstrained image pairs, but it cannot handle multiple images as input. Current attempts on learned MVS mainly employ plane-sweeping approaches and regard the depth estimation problem as a multi-class classification problem [6,19,21]. In practice, these methods employ classical plane-sweep stereo approaches with a defined cost metric to build a cost volume, and the CNN is used to infer depth from the cost volume and refine the depth map. For example, DeepTAM [21] computes the sum of absolute difference (SAD) of 3×3 patches between warped image pairs. To increase the density of sampled planes, an adaptive narrow band strategy is used. DeepMVS [6] proposed a patch matching network to extract features that can aid in the comparison of patches. To do the feature aggregation, it considers both an intra-volume feature aggregation network and inter-volume aggregation network. Semantic features from pre-trained VGG-19 on ImageNet also aids in intra-volume feature aggregation. The Dense-CRF is used to refine the final depth map. MVDepthNet [17] computes the
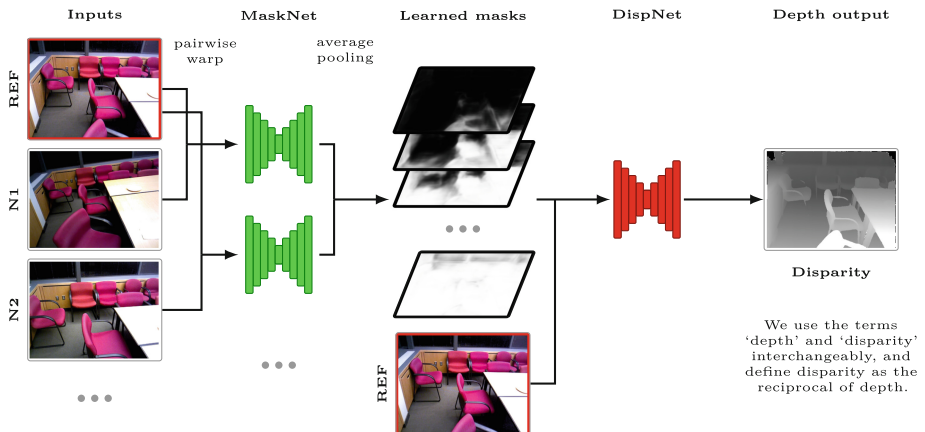
Fig. 2. Overview of our MaskMVS architecture. The MaskNet (left, green) will generate multi-plane mask maps to represent the probability of real surfaces being hit by a ray before each plane. Given the mask-based representation and the reference image, the DispNet (right, red) will learn to predict the disparity map for the reference image. (Color figure online)

absolute difference directly without a supporting window to generate the cost volume, as the pixel-wise cost matching enable the volume to preserve detail information. MVSNet [19] proposes a variance-based cost metric and employ four-scale 3D CNN to obtain smooth cost volume automatically. DPSNet [7] concatenate warped image features firstly and use a series of 3D convolutions to learn the cost volume generation.

Attempts on learned MVS without cost metrics have shown promise in reconstructing 3D objects. [5] propose a CNN to learn multi-patch similarity directly, but it still matches patches explicitly. SurfaceNet [8], and LSM [9] use 3D grid to fuse information. However, due to the volumetric structure, SurfaceNet and LSM are limited to small-scale reconstructions (see discussion in [19]).

3   Methods

The overall architecture of MaskMVS is shown in Fig. 2. The estimation scheme consists of two parts, MaskNet and DispNet. Given (an arbitrary number of) image pairs and known camera poses, we back-project images onto virtual planes to construct the warped volume to feed the MaskNet. The MaskNet will generate multi-plane mask maps to represent the probability of a surface being hit by each ray before each plane. Given the mask-based representation and the reference image, the DispNet learns to predict the disparity map for the reference image. Our methods will be introduced as follows. Sect. 3.1 presents our novel idea of depth plane sampling. Sects. 3.2 and 3.3 explain the details of MaskNet and DispNet, respectively.

3.1   Histogram-Based Depth Plane Sampling

The selection of virtual planes is important for plane-sweeping methods. Current methods generally uniformly sample planes in the depth domain [19,21] or inverse-depth domain [7,17] between predefined minimum and maximum values. One principle of plane selection is to achieve higher sampling density for close by depths and lower density for distant depths, so uniform sampling in the inverse-depth domain generally produces more accurate predictions (see [7]). However, both of these sampling methods rely on a fixed depth range, and the ideal depth ranges for indoor scenes and outdoor scenes are typically different. Some methods, such as [6], pre-run traditional methods like COLMAP [14] to obtain the depth range for each input firstly to deal with different scenes.

   We propose the idea of selecting planes according to the cumulative histogram of depth. This allows us to sample reasonable numbers of depth planes in both nearby and far away depths when the training set is a mixed data set. There will be more pixels covering areas closer to the camera in general, so the histogram of depth distribution is naturally in accordance with the selection principle mentioned above. We define the depth density and cumulative depth density functions as

$$p(d_i) = \frac{n_i}{N} \qquad \text{and} \qquad P(d_i) = \sum_{j=1}^{i} p(d_i), \qquad (1)$$

where n_i is the frequency of the depth value d_i, and N is the total number of pixels in the training data set. These discrete density functions, $p : [0, d_{\max}] \to [0,1]$ and $P : [0, d_{\max}] \to [0,1]$, can be seen as the normalized histogram and normalized cumulative histograms of the depths (we use a binning of 200 points in the experiments). Based on the cumulative density function, we can choose a set of depths covering the entire range by considering the inverse cumulative density function $P^{-1} : [0,1] \to [0, d_{\max}]$. By choosing to cover the quantiles $\theta_1, \theta_2, \ldots, \theta_D$ of P uniformly, we find a set of depths $\{d_i\}_{i=1}^{D}$ such that $d_i = P^{-1}(\theta_i)$, where D is the number of planes.

3.2   MaskNet: Mask-Based Multiplane Representation

Similar to traditional plane-sweep stereo, we firstly construct a warped volume from each image pair by warping the neighbour image via the fronto-parallel planes at fixed depths to the reference frame using the planar homography:

$$\mathbf{H} = \mathbf{K} \left(\mathbf{R} + \mathbf{t} \left(0 \ 0 \ \tfrac{1}{d_i} \right) \right) \mathbf{K}^{-1}, \qquad (2)$$

where $\mathbf{K}$ is the known intrinsics matrix and the relative pose $(\mathbf{R}, \mathbf{t})$ is given in terms of a rotation matrix and translation vector with respect to the neighbour frame. d_i denotes the depth value of the ith virtual plane. The input of the MaskNet has size $3(1 + D) \times H \times W$ consisting of the reference image and concatenated warped neighbour images from D planes (D+1 RGB images with

height H and width W). The output of the MaskNet is a set of mask maps of size $D \times H \times W$.

Traditional plane-sweep based methods need to design a cost metric based on photo-consistency of warped images to select an optimal depth plane for each pixel. In that case, the predicted depth maps will be noisy and the accuracy will be limited by the density of the chosen planes. Instead of using a distance metric, we propose a novel mask-based multiplane representation to roughly encode the near–far relationship. In our method, the intuition is that given a reference image and warped neighbour image on two successive planes, if the relative position of a warped pixel flips, it tells that the surface will be hit by the ray between the two planes. To enable the network to learn this, we assign a supervised binary classification task to the MaskNet, where the ground truth for masks can be obtained from ground truth depth maps directly. The MaskNet will predict whether the ray will hit a surface in front of the plane (including on the plane) or behind the plane for each pixel.

For this purpose, the MaskNet consists of an encoder–decoder architecture that shares a similar architecture with [17]. The encoder includes five convolutional layers, and convolutional filter sizes decrease towards deeper layers: 7×7 for the first layer, 5×5 for the second layer, and 3×3 for the following three layers. There are four skip connections between the encoder and the decoder, and mask maps are predicted in four scales. All layers are followed by batch normalization and ReLU except for the mask prediction layers that are followed by a sigmoid function. Each pixel on each plane has a value in range $[0, 1]$, representing the predicted probability that the true surface is located in front of the sampled plane. To support arbitrary length of sequence, we deal with neighbour images separately and then use average pooling for predicted masks with the finest resolution to fuse information from different neighbours. We compute the pixel-wise cross-entropy between the averaged mask map and the ground truth mask map as loss function to train the MaskNet.

3.3 DispNet: Continuous-Disparity Prediction

To regress continuous depth values, given prediction results from MaskNet and the reference image, we concatenate them into an input of size $(3 + D) \times H \times W$ to feed the DispNet. The encoder–decoder architecture of our DispNet is similar to the DispNet in [13]. There are six convolutional layers for the encoder (3×3 filters except for the first two layers, which have a 7×7 filter and a 5×5 filter, respectively) to extract features, and the decoder will gradually upsample the feature maps, considering also features from the encoder with six skip connections. All convolutional layers are followed by batch normalization and LeakyReLU, and all deconvolution layers are followed by LeakyReLU. Unlike [17,22] that use scale and sigmoid functions to constrain the range of the output, our inverse depth prediction layer consists of a convolutional layer and a ReLU layer to only ensure that predicted inverse depth maps have values larger than zero for all pixels. The DispNet will generate depth maps in six resolutions and
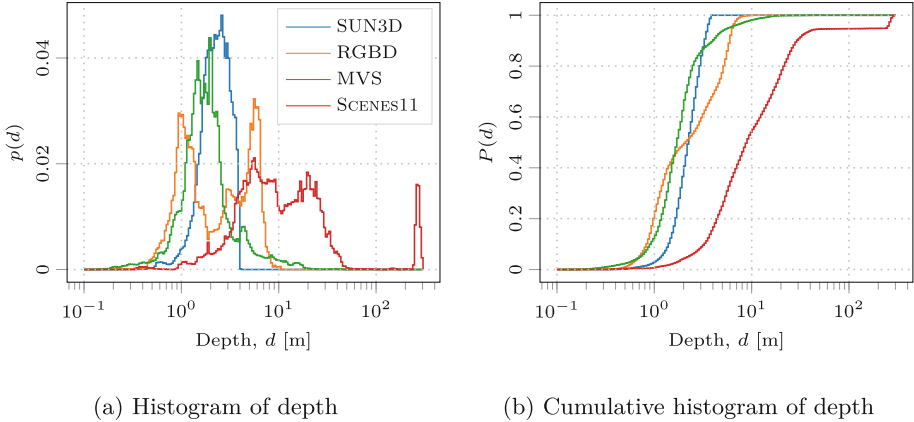
(a) Histogram of depth   (b) Cumulative histogram of depth

Fig. 3. Depth distributions (log-scale) for the four data sets used in training and evaluation. The typical depths in the data are highly dependent on the type of environments covered (*e.g.*, SUN3D covering mostly indoor scenes and SCENES11 having high variability in the type of scenes covered.)

the finest resolution is the same as in the reference image. During training, the loss function is defined as the sum of the average L1 loss at different resolutions:

$$L = \sum_{s=0}^{6} w_s \frac{1}{n_s} \sum_i |\hat{d}_{s,i} - d_{s,i}|, \tag{3}$$

where $\hat{d}_{s,i}$ is the estimated inverse depth at scale s and $d_{s,i}$ is the corresponding ground truth inverse depth. n_s is the number of valid pixels and w_s is the loss weight for scale s. We assign the highest weight for the loss with the finest resolution as 0.5 and others 0.1.

4   Experiments

Similar to DeepMVS [6], we train our networks with the same data sets as used in DeMoN [16]. The training data set includes short sequences from real-world data sets SUN3D [18], RGBD [15], MVS (includes CITYWALL and ACHTECK-TURM [2]), and a synthesized data set SCENES11 [16]. SUN3D consists of a variety of indoors scenes, RGBD provides scenes of an office and an industrial hall. MVS data sets include both indoor and outdoor scenes, and the ground truth depth maps of outdoor scenes are often sparse. SCENES11 provides perfect ground truth but lack realism. Combined, there are 92,558 training samples and each training sample consists of a sequence of three frames, the ground truth depth map for the reference frame, and provided ground truth camera poses. The resolution of the input images is 320 × 240.

Our training procedure consists of two stages as we pre-train the MaskNet first and then employed the pre-trained parameters of MaskNet to predict masks

to train the DispNet. For both networks we use the Adam solver [11] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate for MaskNet and DispNet is $2 \cdot 10^{-4}$ and 10^{-4}, respectively. Our framework is trained using only $D = 16$ sampled depth planes as sparse sampling can present the effect of plane selection more clearly and provide speed-up. The whole framework is implemented on PyTorch for 500k iterations with a mini-batch size of four.

Error Metrics. In our evaluations, we use three common measures: *(i)* L1-rel, *(ii)* L1-inv, and *(iii)* sc-inv (see, *e.g.*, [1]). The two L1 metrics are the mean absolute relative difference and mean absolute difference in inverse depth, respectively. They are given in terms of

$$\text{L1-rel} = \frac{1}{n} \sum_i \frac{|d_i - \hat{d}_i|}{\hat{d}_i} \qquad \text{and} \qquad \text{L1-inv} = \frac{1}{n} \sum_i \left| \frac{1}{d_i} - \frac{1}{\hat{d}_i} \right|, \qquad (4)$$

where d_i [meters] is the predicted depth value, $\hat{d}_i$ [meters] is the ground truth value, n is the number of pixels for which the depth is available. The third, scale-invariant metric, is given as:

$$\text{sc-inv} = \sqrt{\frac{1}{n} \sum_i z_i^2 - \frac{1}{n^2} \left(\sum_i z_i \right)^2} \qquad (5)$$

where $z_i = \log d_i - \log \hat{d}_i$. The L1-rel metric normalizes the error, L1-inv metric gives more importance to close depth values and sc-inv is a scale-invariant metric.

### 4.1   Ablation Study: The Effect of Plane Selection

As the training set is a mixed data set that consists of both indoor scenes, outdoor scenes, and synthesized scenes, the depth ranges of image samples vary with type of data set. Figure 3 shows the distribution of depth for different sets separately. To examine the effects of plane selections, we conducted an ablation study for our method. We compare two options for plane selection: uniform sampling in the inverse-depth space and uniform sampling in the distribution space. Namely, to uniformly sample $D = 16$ planes in the inverse-depth space from $d_{\min} = 0.5\,\text{m}$ to $d_{\max} = 50\,\text{m}$, the ith depth plane is given by:

$$\frac{1}{d_i} = \left(\frac{1}{d_{\min}} - \frac{1}{d_{\max}} \right) \frac{i}{D - 1} + \frac{1}{d_{\max}}. \qquad (6)$$

To uniformly sample planes in the distribution space of the whole data set, we set $\theta_{\min} = 0.1$ and $\theta_{\max} = 1$, then the ith depth plane is given by:

$$\theta_i = \theta_{\min} + (\theta_{\max} - \theta_{\min}) \frac{i}{D}, \quad \text{such that} \quad d_i = P^{-1}(\theta_i), \qquad (7)$$

where $i \in \{0, 1, ..., D - 1\}$. Figure 4a shows the two sampling schemes. The curve is the cumulative histogram of depth of the whole mixed data set and the
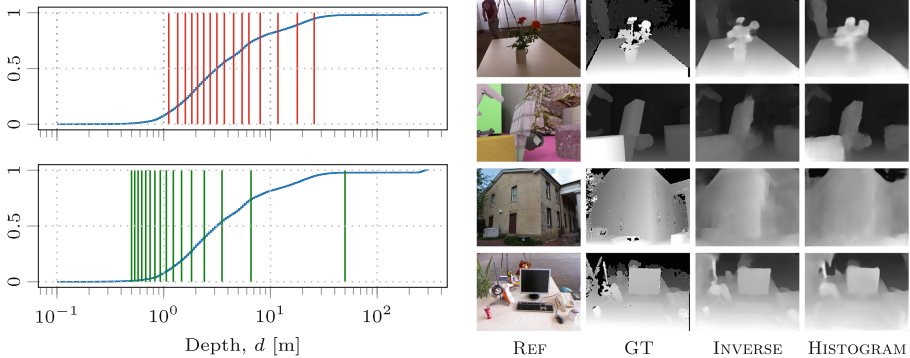
(a) Two ways of depth plane sampling   (b) Comparison of disparity maps

Fig. 4. Setup and results for the ablation study. In (a), the upper figure shows the chosen depth plane depths for the histogram-based sampling and the bottom shows the depths for inverse depth based sampling. In (b), we show qualitative comparison between the two depth plane selection methods used in combination with our MaskMVS.

vertical lines present sampled depths. The higher slope of the curve corresponds to denser distribution of objects. It shows that using histogram-based sampling successfully gives planes within the range with higher slope, and the density is also higher in the closer areas than distant areas.

The evaluation results in Table 1 shows that selecting histogram-based planes perform much better in outdoor and synthesized scenes as it has more planes put in for distant depths. The performance of indoor scenes remains comparable with using inverse-depth sampled planes as it still samples densely-enough in close by depths. Figure 4b shows comparison of the disparity maps from the two sampling approaches. Generally, using histogram-based sampling can provide good quality of prediction in both small-scale and large-scale depth scenes even with sparse depth planes. The last row in the figure shows that our methods with both sampling strategies give good predictions, but using histogram-based sampling failed to capture small objects like cans on the table in the close areas. It is mainly because our sampled planes started from the value farther than 1 m, while the office scenes in RGBD contain many objects within 1 m (see the first bump of RGBD in Fig. 3a); conversely, inverse-space sampled planes are very dense within the range, so its prediction captures these details well.

4.2   Comparisons

We provide both qualitative and quantitative comparisons to the state-of-the-art by evaluating using unstructured view pairs from the test sets in MVS, SUN3D, RGBD, and SCENES11. We compare our methods with two CNN-based multi-view stereo methods (DeepMVS [6] and MVDepthNet [17]) and one traditional

Table 1. Comparison results between MVDepthNet, DeepMVS, COLMAP, and our method. We outperform other methods in most of the data sets and error metrics (smaller better).

	MVDepth-16	DeepMVS	COLMAP	Ours (hist)	Ours (inv.)
scenes11					
L1-rel	0.2352	0.3755	0.7205	**0.1475**	0.2144
L1-inv	0.0292	0.0495	0.0936	**0.0231**	0.0308
sc-inv	0.3207	0.5810	0.7814	**0.2483**	0.2985
MVS					
L1-rel	0.3835	0.8217	0.9921	**0.2669**	0.4030
L1-inv	0.1384	**0.1065**	0.1812	0.1377	0.1600
sc-inv	0.3427	0.5325	0.6892	**0.3001**	0.3100
sun3d					
L1-rel	0.1840	0.8604	1.8499	0.1797	**0.1611**
L1-inv	0.0865	0.1317	0.4511	**0.0818**	0.0808
sc-inv	0.2013	0.4992	1.1219	0.1916	**0.1769**
RGBD					
L1-rel	0.1628	0.5066	2.2992	0.1748	**0.1572**
L1-inv	**0.0789**	0.1717	0.5593	0.0846	0.0802
sc-inv	0.2360	0.5238	1.2970	0.2304	**0.2062**

multiview stereo method (COLMAP [14]). The original MVDepthNet is trained with 64 planes and a larger data set (covering also the standard test samples in the sets). In order to make a fair comparison, we retrained the MVDepthNet with our training data set and 16 planes. The results are reported in Table 1. Our predictions have significantly lower errors in SCENES11 and MVS, and comparable performance in SUN3D and RGBD. The improvement of the outdoor scenes and synthesized scenes can be explained by the consideration of far depth planes. As mentioned in Sect. 4.1, almost half of the scenes of RGBD include objects closer than 1 m that is below our histogram-based sampling range, so the performance is slightly worse. We also evaluated the inference time for CNN-based models on a desktop workstation (NVIDIA GTX 1080 Ti, i7-7820X CPU and 63 GB memory; average over 100 predictions): DeepMVS 5.81 s, MVDepth-16 0.063 s, ours 0.089 s. The running time of our model is comparable to MVdepth-16 but our accuracy is better. Both models are significantly faster than DeepMVS.

Figure 5 shows qualitative comparisons between MVDepthNet, DeepMVS, COLMAP, and our MaskMVS approach. It should be noted that our method is the only method that can capture the small objects in the top left of the third row and the bottom left of the last row. Moreover, our method provide more accurate prediction for close areas (see the brightest parts of ground truths) in the first row and the fourth row.
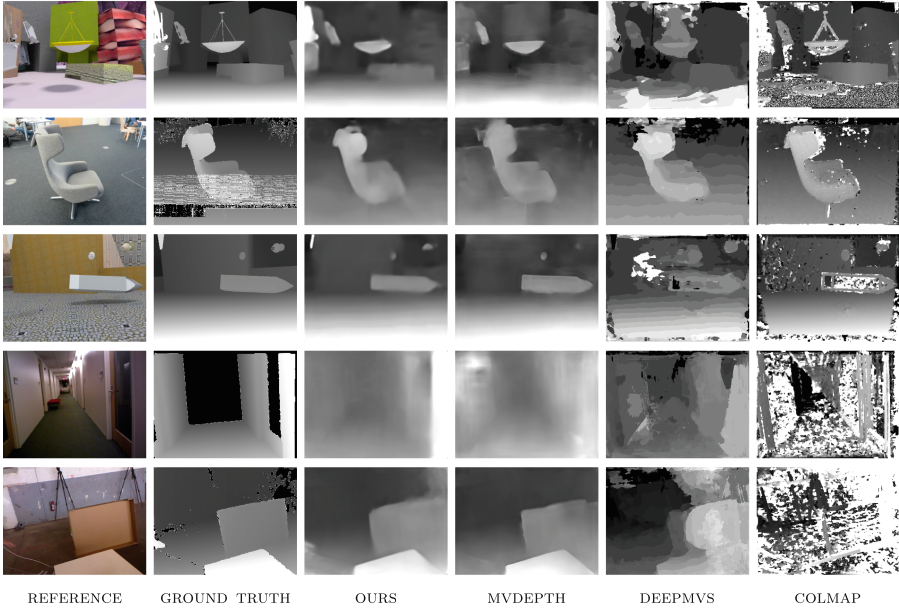
REFERENCE   GROUND TRUTH   OURS   MVDEPTH   DEEPMVS   COLMAP

Fig. 5. Qualitative comparisons between different algorithms on the MVS, Scenes11, Sun3d, and RGBD test sets. The traditional COLMAP method fails in low-texture environments. Our methods successfully captures small objects in close areas and provides better shape estimates for objects in far areas at the same time. Missing values in ground truth are shown in black.

5   Discussion and Conclusions

We have proposed a novel CNN-based architecture for multi-view stereo depth estimation that is inspired by traditional plane-sweep algorithms without the need of constructing an explicit cost-volume metric. Instead, we designed a binary classification task for our MaskNet and used the mask-based multiplane representations to aggregate information from multiple views and exploit geometric relationships. Moreover, we discussed the effect of depth selection and proposed a novel way of sampling depth planes based on histogram matching. Our ablation study showed that uniformly sampling in the distribution domain can deal with both small depths such as indoor scenes and large depths such as in outdoor scenes, even with sparsely sampled planes. As the running time will drop when reducing the number of planes, our proposed sampling method can be beneficial for real-time systems that have restrictions on computation time and memory. Moreover, compared to traditional multi-view stereo methods, our approach can handle low-texture inputs and does not need iterative refinement; compared to other CNN-based methods that also employ a plane-sweep scheme, our method do not need to compute any distance metrics, which makes our method time-efficient.

As ideal plane selection can lead to better prediction, one direction to improve our architecture might be adjusting depth planes according to inputs automatically. It should be noted that using predicted mask maps from our MaskNet, the depth distribution can be roughly estimated. Then it is possible to obtain uniform samples in distribution domain by just 1D linear interpolation. This might offer the possibility for varying sampled depth planes with different scenes in the future. Codes are available at https://github.com/AaltoVision/MaskMVS.

References

1. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: NIPS (2014)
2. Fuhrmann, S., Langguth, F., Goesele, M.: MVE - a multi-view reconstruction environment. In: GCH (2014)
3. Furukawa, Y., Hernández, C.: Multi-view stereo: a tutorial. Found. Trends® Comput. Graph. Vis. **9**(1–2), 1–148 (2015)
4. Godard, C., Mac Aodha, O., Brostow, G.J.: Unsupervised monocular depth estimation with left-right consistency. In: CVPR (2017)
5. Hartmann, W., Galliani, S., Havlena, M., Van Gool, L., Schindler, K.: Learned multi-patch similarity. In: ICCV, pp. 1595–1603 (2017)
6. Huang, P.H., Matzen, K., Kopf, J., Ahuja, N., Huang, J.B.: DeepMVS: learning multi-view stereopsis. In: CVPR (2018)
7. Im, S., Jeon, H.G., Lin, S., Kweon, I.S.: DPSNet: end-to-end deep plane sweep stereo. In: ICLR (2019)
8. Ji, M., Gall, J., Zheng, H., Liu, Y., Fang, L.: SurfaceNet: an end-to-end 3D neural network for multiview stereopsis. In: ICCV (2017)
9. Kar, A., Häne, C., Malik, J.: Learning a multi-view stereo machine. In: NIPS (2017)
10. Kendall, A., et al.: End-to-end learning of geometry and context for deep stereo regression. In: CVPR (2017)
11. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
12. Liu, F., Shen, C., Lin, G., Reid, I.D.: Learning depth from single monocular images using deep convolutional neural fields. IEEE Trans. Pattern Anal. Mach. Intell. **38**(10), 2024–2039 (2016)
13. Mayer, N., et al.: A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In: CVPR (2016)
14. Schönberger, J.L., Zheng, E., Frahm, J.-M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 501–518. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_31
15. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of RGB-D SLAM systems. In: IROS (2012)
16. Ummenhofer, B., et al.: DeMoN: depth and motion network for learning monocular stereo. In: CVPR (2017)
17. Wang, K., Shen, S.: MVDepthNet: real-time multiview depth estimation neural network. In: International Conference on 3D Vision (3DV) (2018)

18. Xiao, J., Owens, A., Torralba, A.: SUN3D: a database of big spaces reconstructed using SFM and object labels. In: ICCV (2013)
19. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: MVSNet: depth inference for unstructured multi-view stereo. In: ECCV (2018)
20. Yin, Z., Shi, J.: GeoNet: unsupervised learning of dense depth, optical flow and camera pose. In: CVPR (2018)
21. Zhou, H., Ummenhofer, B., Brox, T.: DeepTAM: deep tracking and mapping. In: ECCV (2018)
22. Zhou, T., Brown, M., Snavely, N., Lowe, D.G.: Unsupervised learning of depth and ego-motion from video. In: CVPR (2017)