RayMVSNet: Learning Ray-based 1D Implicit Fields for Accurate Multi-View Stereo

Junhua Xi* Yifei Shi* Yijie Wang Yulan Guo Kai Xu†
National University of Defense Technology

Abstract

Learning-based multi-view stereo (MVS) has by far centered around 3D convolution on cost volumes. Due to the high computation and memory consumption of 3D CNN, the resolution of output depth is often considerably limited. Different from most existing works dedicated to adaptive refinement of cost volumes, we opt to directly optimize the depth value along each camera ray, mimicking the range (depth) finding of a laser scanner. This reduces the MVS problem to ray-based depth optimization which is much more light-weight than full cost volume optimization. In particular, we propose RayMVSNet which learns sequential prediction of a 1D implicit field along each camera ray with the zero-crossing point indicating scene depth. This sequential modeling, conducted based on transformer features, essentially learns the epipolar line search in traditional multi-view stereo. We also devise a multi-task learning for better optimization convergence and depth accuracy. Our method ranks top on both the DTU and the Tanks & Temples datasets over all previous learning-based methods, achieving overall reconstruction score of $0.33mm$ *on DTU and f-score of* 59.48% *on Tanks & Temples.*

1. Introduction

Learning-based multi-view stereo has gained a surge of attention recently since the seminal work of MVSNet [42]. The core idea of MVSNet and many followup works is to construct a 3D cost volume in the frustum of the reference view through warping the image features of several source views onto a set of fronto-parallel sweeping planes at hypothesized depths. 3D convolutions are then conducted on the cost volume to extract 3D geometric features and regress the final depth map of the reference view.

Most existing methods are limited to low-resolution cost volume since 3D CNN is generally computation and memory consuming. Several recent works proposed to upsample or refine cost volume aiming at increasing the resolution of output depth maps [8, 13, 41]. Such refinement, however, still needs to trade off between depth and spatial (image) resolutions. For example, CasMVSNet [13] opts to narrow down the range of depth hypothesis to allow high-res depth estimation, matching the spatial resolution of input RGB. 3D convolution is then naturally confined within the narrow band, thus degrading the efficacy of 3D feature learning.

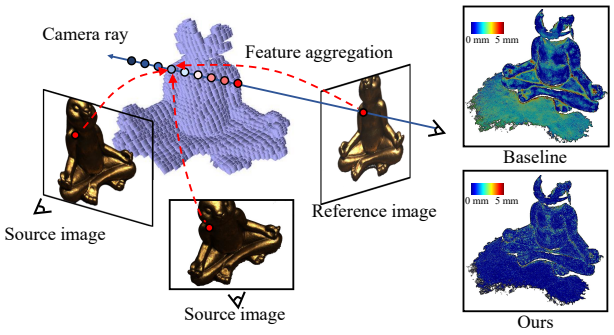


Figure 1. RayMVSNet performs multi-view stereo via predicting 1D implicit fields on a camera ray basis. The sequential prediction of 1D field is light-weight and the monotonicity of ray-based distance field around surface-crossing points facilitates robust learning, leading to more accurate depth estimation than the purely cost-volume-based baselines such as MVSNet [42].

In fact, depth map is view-dependent although cost volume is not. Since the target is depth map, refining the cost volume seems neither economic nor necessary. There could be a large portion of the cost volume invisible to the view point. In this work, we advocate direct optimizing the depth value along each camera ray, mimicking the range (depth) finding of a laser scanner. This allows us to convert the MVS problem into ray-based depth optimization which is, individually, a much more light-weight task than full cost volume optimization. We formulate the "range finding" of each camera ray as learning a 1D implicit field along the ray whose zero-crossing point indicates the scene depth along that ray (Figure 1). To do so, we propose RayMVSNet which learns sequential modeling of multi-view features along camera rays based on recurrent neural networks.

Technically, we propose two critical designs to facilitate learning accurate ray-based 1D implicit fields. *Firstly*, the sequential prediction of 1D implicit field along a camera ray is essentially conducting an epipolar line search [2] for

*Joint first authors
†Corresponding author: kevin.kai.xu@gmail.com

cross-view feature matching whose optimum corresponds to the point of ray-surface intersection. To learn this line search, we propose *Epipolar Transformer*. Given a camera ray of the reference view, it learns the matching correlation of the pixel-wise 2D features of each source view based on attention mechanism. The transformer features of all views, together with (low-res) cost volume features, are then concatenated and fed into an LSTM [15] for implicit field regression. Figure 3 visualizes how epipolar transformer selects reliable matching features from different views.

Secondly, we confine the sequential modeling for each camera ray within a fixed-length range centered around the hypothesized surface-crossing point given by the vanilla MVSNet. This makes the output 1D implicit field along each ray monotonous, which is normalized to $[-1, 1]$. Such restriction and normalization lead to significant reduction of learning complexity and improvement of result quality. We devise two learning tasks: 1) sequential prediction of signed distance at a sequence of points sampled in the fixed-length range and 2) regression of the zero-crossing position on the ray. A carefully designed loss function correlates the two tasks. Such multi-task learning approach yields highly accurate estimation of per-ray surface-crossing points.

Learning view-dependent implicit fields has been well-exploited in neural radiance fields (NeRF) [25] with great success. Recently, NeRF was combined with MVSNet for better generality [4]. Albeit sharing conceptual similarity, our work is a completely different from NeRF. *First*, NeRF (including MVSNeRF [4]) is designed for novel view synthesis, a different task from MVS. *Second*, the radiance field in NeRF is defined and learned in continuous 3D space and camera rays are used only in the volume rendering stage. In our RayMVSNet, on the other hand, we explicitly learn 1D implicit fields on a camera ray basis.

RayMVSNet ranks top on both the DTU and the Tanks & Temples datasets over all learning-based methods. It achieves overall reconstruction score of 0.33mm on DTU, and f-score of 59.48% on Tanks & Temples. Notably, since all rays share weights for the LSTM and the epipolar transformer, the RayMVSNet model is light weight. Moreover, the computation for each ray is highly parallelizable.

Our work makes the following contributions:

- A novel formulation of deep MVS as learning ray-based 1D implicit fields.
- An epipolar transformer designed to learn cross-view feature correlation with attention mechanism.
- A multi-task learning approach to sequential modeling and prediction of 1D implicit fields based on LSTM.
- A challenging test set focusing on regions with specular reflection, shadow or occlusion based on the DTU dataset [1] and associated extensive evaluations.

2. Related Work

Learning-based MVS. Recent advances have made remarkable progress on learning-based MVS. Hartmann et al. [14] first propose to learn the multi-patch similarity from two views by a Siamese convolutional network. SurfaceNet [18] and DeepMVS [16] warp the multi-view images into the 3D cost volume and adopt 3D neural networks to estimate the geometry. MVSNet [42] proposes a differentiable homography and leverages 3D cost volume in a learning pipeline. MVSNet aggregates contextual information by a 3D convolutional network. However, the high computation and memory consumption restrict the output depth resolution, limiting its scalability in large scenes.

To reduce the requirements, many follow-up works have been developed. R-MVSNet [43] proposes to regularize the 2D cost maps along the depth direction so the memory consumption could be greatly reduced. Point-MVSNet [5] first computes the coarse depth with a low-resolution cost volume and then uses a point-based refinement network to generate the high-resolution depth map. CasMVSNet [13] adopts a cascade cost volume to gradually narrow the depth range and increase the cost volume resolution. Similar ideas are later explored to reduce the memory cost of 3D convolutions and/or increase the depth quality, such as coarse-to-fine depth optimization [8, 23, 37, 38, 40, 41, 47], attention-based feature aggregation [22, 36, 46, 50], and patch matching-based method [21, 35]. Unlike these works, RayMVSNet optimizes the depth on each camera viewing ray instead of the 3D volume, which is more light-weight.

Multi-view feature aggregation is one of the most crucial components in learning-based MVS. Previous works adopted various solutions to learn mutual correlations [51], avoiding the influences of incorrect matches caused by occlusion. Popular solutions include the visibility-based aggregation [6, 48], the attention-based aggregation [36, 45], etc. RayMVSNet follows the attention-based aggregation route. Nevertheless, it learns feature aggregation at each 3D point, instead of the entire image or volume, thus greatly reducing the memory consumption.

Learning Implicit Representation. Many works have attempted learning shape representation based on implicit fields. Implicit field shows promising results on facilitating a variety number of problems, such as shape reconstruction [9, 26, 49, 52] and rendering [25, 32]. DeepSDF [28] proposes to predict the magnitude of 3D point to indicate the distance to the surface boundary and a sign to determine whether the point is inside or outside of the shape. IM-Net [7] and Occupancy Network [24] learn the implicit fields to estimate the point-wise occupancy probability with a binary classifier. To improve the effectiveness and generalization on complex scenes, latest studies propose to enhance implicit field by introducing extra inputs [29, 39],

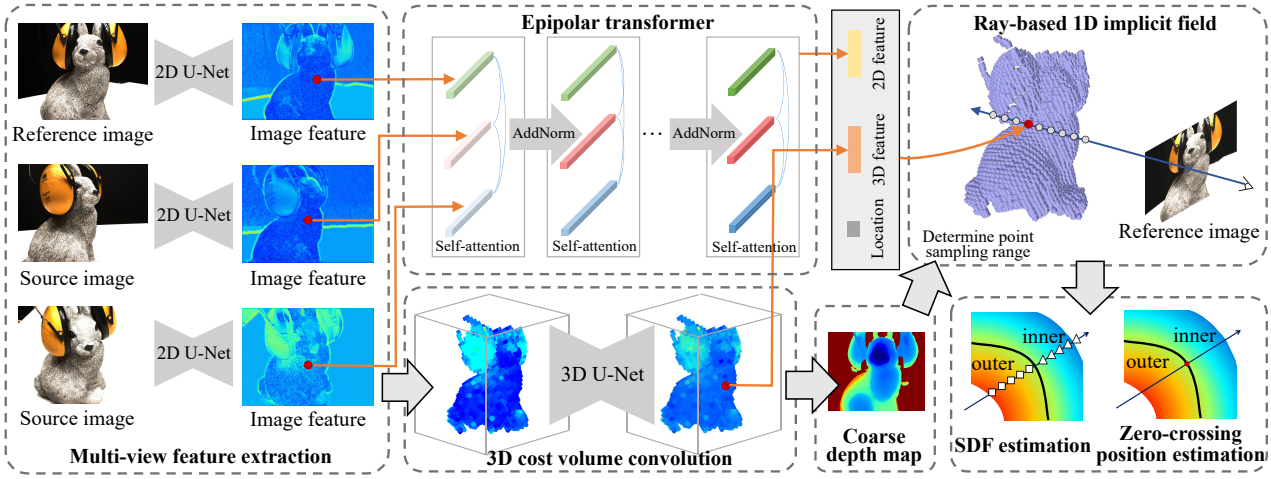


Figure 2. Method overview. Given multiple overlapping RGB images, the multi-view image features are extracted by a 2D U-Net. The coarse depth map is then estimated by a coarse 3D cost volume. 2D multi-view image features are then correlated and aggregated by epipolar transformer. At last, the 1D implicit field is learnt on each camera viewing ray to simultaneously estimate the SDF of the sampled points and the location of the zero-crossing point.

adopting advanced learning techniques [10, 27, 31, 33] and decomposing the scene into local regions [3, 12, 19, 32]. NeRF [25] represents complex scene by learning a view-dependent implicit neural radiance field, achieving high-resolution realistic novel view synthesis.

3. Method

Overview. RayMVSNet estimates the depth maps from multiple overlapping RGB images. Similar to [42], at each time, it takes one reference image I_1 and $N-1$ source images $\{I_i\}_2^N$ as input, and infers the depth map of the reference image. RayMVSNet starts from building a light-weight 3D cost volume and estimating a coarse depth map (Sec. 3.1). Then, epipolar transformer is proposed to learn the matching correlation of the pixel-wise 2D features of each view using attention mechanism (Sec. 3.2). The transformed features are fed into the 1D implicit field, implemented by an LSTM, along each camera viewing ray to estimate the signed distance functions (SDFs) of the hypothesized points as well as the zero-crossing position (Sec. 3.3). The method overview is illustrated in Figure 2.

3.1. 3D Cost Volume and Coarse Depth Prediction

We first feed the multi-view images $\{I_i\}_1^N$ to a 2D U-Net to extract image features $\{\mathbf{F}_i^I\}_1^N$. The width and height of the image features are the same to those of the input images. Hence, $\{\mathbf{F}_i^I\}_1^N$ preserve the fine appearance feature of local details, facilitating the high-resolution depth estimation. By leveraging the 2D multi-view image features and the camera parameters, we build a variance-based 3D cost volume V, and extract the 3D volumetric features $\mathbf{F}^V$ via a 3D U-Net [42]. Since 3D convolution is memory-consuming, the resolution of V in our work is set to be smaller than that in the previous works [8, 13, 41]. The coarse depth maps are estimated from the 3D volumetric features, which are then used for determining the modeling range of the ray-based 1D implicit fields.

3.2. Epipolar Transformer

We cast a set of rays $\mathbf{R} = \{\mathbf{r}_i\}_1^M$ from the camera's viewing direction of the reference image, where M is the number of pixels in the reference image. Our goal is to estimate the location of the zero-crossing point on each ray, so we can obtain the depth map of the reference view. Compared to methods that estimate depth on the 3D cost volume, the ray-based method maintains the following advantages. *First*, since the depth map is view-dependent, ray-based depth optimization is more straightforward and light-weight. *Second*, all the ray-based 1D implicit fields share an identical spatial property, i.e. the monotonicity of the SDFs along the ray direction. As a result, the learning would be simplified and well regularized, leading to efficient network training and more accurate results.

Zero-crossing hypothesis sampling. We perform a point sampling to generate the zero-crossing point hypothesis on each ray. Ideally, one could generate as many points as possible on each ray. However, most of the points are far from the surface, providing less informative information for the depth estimation. To facilitate efficient training, as shown in Figure 4 (a), we adopt the coarse depth map predicted in sec. 3.1 and uniformly sample K points $P = \{p_k\}_1^K$ on the ray in the range of $\pm\delta$ around the estimated coarse depth.

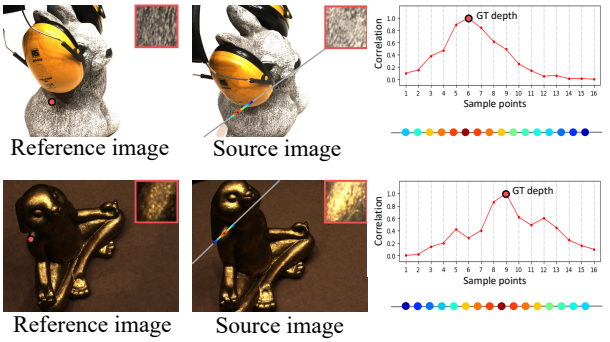


Figure 3. Effects of epipolar transformer. Given a point in the reference image, epipolar transformer automatically selects reliable matching feature on the epipolar line of the source image. Note that it finds the matching feature correctly despite the influences of light changing (top row) and specular reflection (bottom row). The visualized point-pair correlations are deduced from the Softmax($\mathbf{QK}^T$) in Formulation 1.

Attention-aware cross-view feature correlation. The next step is to aggregate feature for the hypothesized points based on the multi-view image features. A naive way to achieve this is to fetch the features from multi-view images based on the view projection, and take the variance. However, image feature could be easily influenced by image defects, such as specular reflection and light changing. Naive variance considers all image features equally, which might incur unreliable features and provide incorrect cross-view feature correlation. To alleviate this problem, we propose *Epipolar Transformer* to learn cross-view feature correlation with attention mechanism (Figure 4 b).

To be specific, the network architecture of epipolar transformer contains four self-attention layers, each followed by two AddNorm layers and one feed-forward layer. Suppose $\mathbf{X} = \text{Concat}(\mathbf{F}^I_{1,p}, ..., \mathbf{F}^I_{N,p})$, where Concat$(\cdot)$ is the concatenation operation, $\{\mathbf{F}^I_{i,p}\}^N_1$ are the fetched multi-view image features at 3D point p. The self-attention layer of epipolar transformer is:

$$\mathbf{S} = \text{SelfAttention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Softmax}(\mathbf{QK}^T)\mathbf{V}, \quad (1)$$

where $\mathbf{Q} = \mathbf{XW^Q}$, $\mathbf{K} = \mathbf{XW^K}$, $\mathbf{V} = \mathbf{XW^V}$ are the query vector, the key vector and the value vector respectively. $\mathbf{W^Q}$, $\mathbf{W^K}$, $\mathbf{W^V}$ are the learned weights. Examples to demonstrate the effects of first self-attention layer in epipolar transformer are visualized in Figure 3. The AddNorm layer of epipolar transformer is:

$$\mathbf{Z} = \text{AddNorm}(\mathbf{X}) = \text{LayerNorm}(\mathbf{X} + \mathbf{S}), \quad (2)$$

where LayerNorm$(\cdot)$ is the layer normalization operation. The output of epipolar transformer is the attention-aware denoised multi-view feature $\mathbf{F}^A_p = \{\mathbf{F}^A_{1,p}, ..., \mathbf{F}^A_{N,p}\}$.

To further improve the feature quality, we concatenate the attention-aware feature with the 3D volume feature $\mathbf{F}^V_p$ fetched from the 3D cost volume processed in Sec. 3.1:

$$\mathbf{F}_p = \text{Concat}(\mathbf{F}^A_{\mu,p}, \mathbf{F}^A_{\sigma,p}, \mathbf{F}^A_{1,p}, \mathbf{F}^V_p). \quad (3)$$

where $\mathbf{F}^A_{\mu,p}$ and $\mathbf{F}^A_{\sigma,p}$ are the mean and variation of the elements in $\mathbf{F}^A_p$ [17, 42]. $\mathbf{F}^A_{1,p}$ is the attention-aware feature at 3D point p in the reference image.

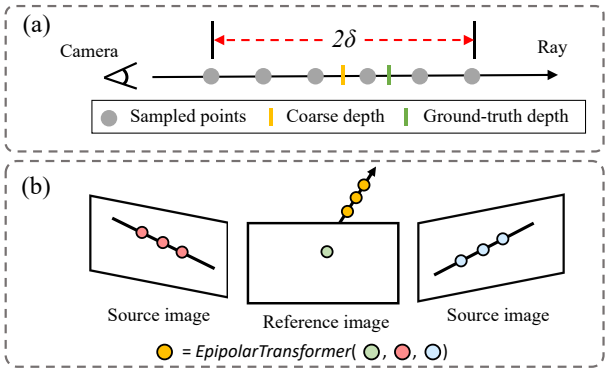


Figure 4. (a) The hypothesized points are sampled around the predicted coarse depth to narrow down the search space of the zero-crossing position. (b) Epipolar transformer learns the matching correlation of the pixel-wise 2D features and aggregates these features using an attention mechanism.

3.3. Ray-based 1D Implicit Field

LSTM vs. alternative. Given the features of the hypothesized points, the ray-based 1D implicit fields are learned with an LSTM [15]. Crucially, we leverage two attributes of LSTM. *First*, the mechanism of sequential processing inherently facilitates the learning of the SDF monotonicity along the ray direction. *Second*, the property of time invariance increases the network robustness by allowing the zero-crossing position to appear at any place (time-step) on the ray. An alternative to performing sequential inference is to use transformer [34]. However, we experimentally found that replacing LSTM with transformer would not make the performance improve (see Table 3).

Network architecture. The network architecture of the 1D implicit field is shown in Figure 5. The LSTM first aggregates the hypothesized points sequentially, and generates the ray feature $\mathbf{c}_K$. Specifically, the formulations of an LSTM unit at time-step k are:

$$\begin{aligned}
\mathbf{z} &= \tanh(\mathbf{W}[\mathbf{F}_k, \mathbf{h}_{k-1}] + b), \\
\mathbf{z}^f &= \sigma(\mathbf{W}^f[\mathbf{F}_k, \mathbf{h}_{k-1}] + b^f), \\
\mathbf{z}^u &= \sigma(\mathbf{W}^u[\mathbf{F}_k, \mathbf{h}_{k-1}] + b^u), \\
\mathbf{z}^o &= \sigma(\mathbf{W}^o[\mathbf{F}_k, \mathbf{h}_{k-1}] + b^o), \\
\mathbf{c}_k &= \mathbf{z}^f \circ \mathbf{c}_{k-1} + \mathbf{z}^u \circ \mathbf{z}, \\
\mathbf{h}_k &= \mathbf{z}^o \circ \tanh(\mathbf{c}_k),
\end{aligned} \quad (4)$$

where $\mathbf{F}_k$ is the feature of point p_k, $\mathbf{h}_k$ and $\mathbf{h}_{k-1}$ are the hidden state of point p_k and p_{k-1} respectively, $\mathbf{z}$ is the cell

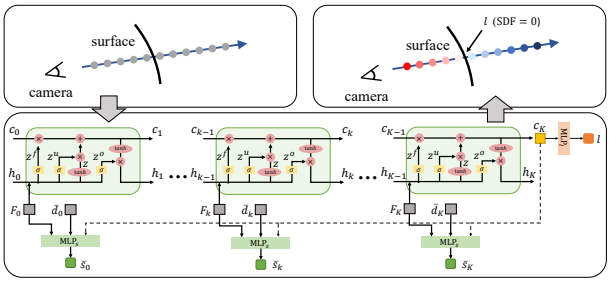


Figure 5. Network architecture of the ray-based 1D implicit field. The hypothesized points are fed into an LSTM sequentially, to estimate the position of the zero-crossing point as well as the SDFs.

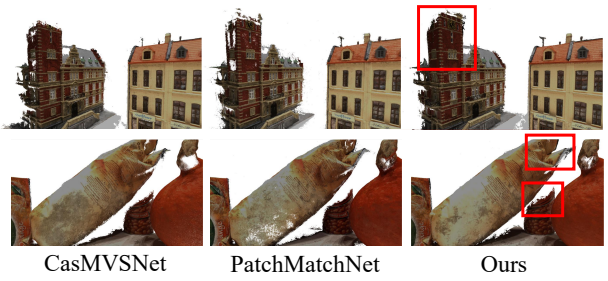


Figure 6. Visual comparison of the reconstructed point cloud by RayMVSNet and the baselines. Please pay attention to the results of the challenging areas highlighted in the figures.

input activation vector, $\mathbf{z}^f$ is the activation vector of the forget gate, $\mathbf{z}^u$ is the activation vector of the update gate, $\mathbf{z}^o$ is the activation vector of the output gate, $\mathbf{c}_k$ is the cell state vector, $\mathbf{W}$, $\mathbf{W}^f$, $\mathbf{W}^u$, $\mathbf{W}^o$ are the weight matrices, b, b^f, b^u, b^o are the weight vectors, $\circ$ is the element-wise multiplication. The LSTM is initialized with $\mathbf{c}_0 = 0$ and $\mathbf{h}_0 = 0$.

For each hypothesized point p_k, we use the ray feature $\mathbf{c}_K$, the point-wise feature $\mathbf{F}_k$ and its depth value d_k (indicating the location on the ray) to estimate its SDF s_k using an MLP. Instead of using the true depth value d_k and estimating the true SDF s_k, we use the normalized depth value $\bar{d}_k = k/K \in [0, 1]$ and the normalized SDF $\bar{s}_k = s_k/s_{max} \in [-1, 1]$, where s_{max} is the maximal absolute SDF value on the ray. Such normalization leads to a significant reduction of learning complexity and improvement of the result quality. The formulation of the SDF prediction is:

$$\bar{s}_k = \text{MLP}_s([\mathbf{c}_K, \mathbf{F}_k, \bar{d}_k]). \tag{5}$$

The above network predicts the SDFs of the hypothesized points on the ray. However, post-processing, e.g. ray casting, is still needed to find the zero-cross position. We extend our method to estimate the zero-cross position explicitly with another MLP. Taking the ray feature $\mathbf{c}_K$ as input, the MLP predicts the zero-crossing location $l \in [0, 1]$ on the ray in the normalized 1D coordinate:

$$l = \text{MLP}_l(\mathbf{c}_K). \tag{6}$$

Loss functions. We adopt a multi-task learning strategy to optimize RayMVSNet. The two tasks, i.e. SDF estimation and zero-crossing position estimation, are inherently relevant and could reinforce each other by optimizing the following loss:

$$\mathcal{L} = w_s\mathcal{L}_s + w_l\mathcal{L}_l + w_{sl}\mathcal{L}_{sl}, \tag{7}$$

where $\mathcal{L}_s$ and $\mathcal{L}_l$ are the loss of the SDF estimation and the zero-crossing location estimation, respectively:

$$\begin{aligned} \mathcal{L}_s &= \sum_{k=1}^{K} L_1(s_k, \hat{s_k}), \\ \mathcal{L}_l &= L_1(l, \hat{l}), \end{aligned} \tag{8}$$

where $\hat{s_k}$ and $\hat{l}$ are the ground-truth, $L_1(\cdot)$ denotes the L1 loss function. $\mathcal{L}_{sl}$ is a relational loss that penalizes the inconsistency between the predicted SDFs and the predicted zero-crossing position:

$$\mathcal{L}_{sl} = \begin{cases} 1, & s_l^a \times s_l^b > 0 \\ 0, & s_l^a \times s_l^b \leq 0, \end{cases} \tag{9}$$

where s_l^a and s_l^b are the predicted SDF of the closest two sampled points around the predicted zero-crossing position on the ray. w_s, w_l, w_{sl} are the pre-defined weights.

3.4. Implementations

We provide implementation details of the training and inference. At each time, RayMVSNet takes several images with input image size 640×512. The output feature size is $640 \times 512 \times 8$. The 2D U-Net consists of 6 convolutional layers and 6 deconvolutional layers, each followed by a batch normalization layer and a ReLU layer, except for the last ones. The 3D cost volume is fed into a 3D U-Net which consists of three 3D convolutional layers and three 3D deconvolutional layers. On each ray, the number of hypothesized points K is 16. Range of point sampling δ is $20mm$ for DTU and $100mm$ for Tanks & Temples. The feature fetching from images and volume are achieved by using bilinear interpolation and trilinear interpolation, respectively. The hidden dimension of $\mathbf{z}, \mathbf{z}^f, \mathbf{z}^u, \mathbf{z}^o, \mathbf{c}_k, \mathbf{h}_k$ are 50. MLP_l and MLP_s both contain 4 fully-convolutional layers. The weights w_s, w_l, w_{sl} of multi-task learning loss function are 0.1, 0.8, 0.1, respectively. Epipolar transformer and the LSTM are jointly trained. We use Adam optimizer with initial learning rate 0.0005 which is decreased by 0.9 for every 2 epochs. The training takes 48 hours. The inference time is about 2 seconds. We filter and fuse the depth maps to produce 3D point cloud like previous work [42].

4. Results and Evaluation

Datasets. We train and test RayMVSNet on the *DTU* dataset [1]. The *DTU* dataset contains 79 training scans and 22 testing scans, all captured under changing lighting conditions. Since *DTU* did not provide SDF annotations, we densely generate the point-wise SDFs using the reconstructed surfaces [28, 42]. Besides, three challenging test subsets focusing on regions with *Specular reflection*, *Shadow* and *Occlusion* are created from the *DTU* test set. These regions are manually annotated and are designed for evaluating the method performance on challenging cases. Please refer to the supplemental material for the subsets details. To evaluate the generality, we test RayMVSNet on *Tanks & Temples* [20] which contains large-scale complex scenes, using the trained model on *DTU* without any fine-tuning. *BlendedMVS* [44] is another large-scale dataset consists of various complex scenes. We provide qualitative results on *BlendedMVS* to show the scalability of our method.

4.1. Performance on DTU

Evaluation on point cloud. To evaluate the proposed method on *DTU*, we compare *Accuracy* and *Completeness* of the reconstructed point cloud using the distance metric in [1]. The quantitative results are shown in Table 1. It shows that our method not only produces competitive results in terms of *Accuracy* and *Completeness*, but also achieves the state-of-the-art *Overall* performance. This demonstrates the effectiveness of RayMVSNet, especially on balancing the trade-off between *Accuracy* and *Completeness*. The qualitative comparisons are visualized in Figure 6. It is shown that our method achieves high-quality reconstruction in various scenarios. In particular, our method outperforms the baselines in scenes with textureless regions, heavy occlusion, and complex geometry.

Table 1. Quantitative results on the *DTU* dataset. We compare all methods using the distance metric [1]. The numbers are reported in mm (lower is better).

Method	Accuracy	Completeness	Overall
Gipuma [11]	0.283	0.873	0.578
MVSNet [42]	0.396	0.527	0.462
R-MVSNet [43]	0.383	0.452	0.417
CIDER [38]	0.417	0.437	0.427
P-MVSNet [21]	0.406	0.434	0.420
Point-MVSNet [5]	0.342	0.411	0.376
Fast-MVSNet [47]	0.336	0.403	0.370
Att-MVSNet [22]	0.383	0.329	0.356
CasMVSNet [13]	0.325	0.385	0.355
CVP-MVSNet [41]	0.296	0.406	0.351
PatchmatchNet [35]	0.427	0.277	0.352
UCS-Net [8]	0.338	0.349	0.344
AACVP-MVSNet [46]	0.357	0.326	0.341
U-MVS [37]	0.354	0.353	0.354
Ours	0.341 (6th)	0.319 (2nd)	**0.330 (1st)**

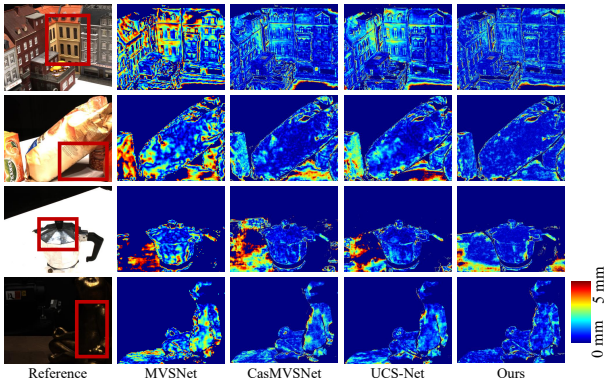


Figure 7. Visual comparison of the estimated depth map by RayMVSNet and the baselines.

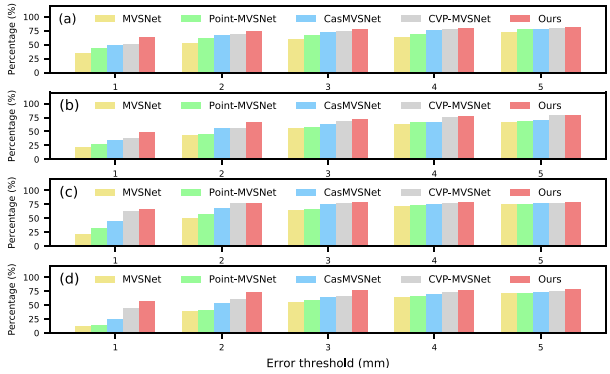


Figure 8. Quantitative comparisons on the depth map prediction of the whole *DTU* test set (a) and the challenging test subsets: *Specular reflection* (b), *Shadow* (c) and *Occlusion* (d). The percentage (Y-axis) represents the ratio of the pixels whose depth prediction error is smaller than the specific error thresholds (X-axis).

Evaluation on depth map. To further demonstrate our advantage, we compare RayMVSNet with existing works, in terms of the predicted depth map. The quantitative comparisons on the whole *DTU* test set (Figure 8 a) and the challenging subsets (Figure 8 b-d) are reported. The percentage (Y-axis) represents the ratio of the pixels whose depth prediction error is smaller than the specific error thresholds (X-axis). Higher percentages represent better performances. It is clear that our method outperforms all the baselines in all error thresholds. Crucially, our method is more general and robust in challenging cases as shown in Figure 7, thanks to the prior learnt from the ray-based 1D implicit field.

4.2. Performance on Tanks & Temples

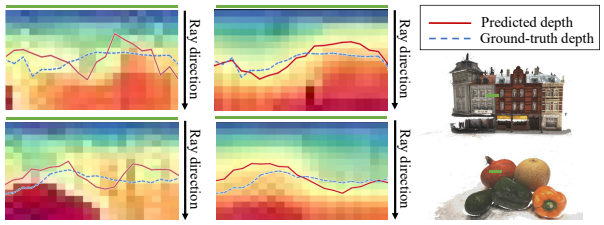
We compare our method with the baselines on *Tanks & Temples*. Following the protocol of previous work [13], we use the network trained on *DTU*. *F-score* is the evaluation metric. The quantitative results are shown in Table 2. Our method achieves the best performance, demonstrating the generality of epipolar transformer and ray-based 1D implicit field on large-scale scenes.

Table 2. Quantitative results on the *Tanks & temples* dataset. We use the f-score as the evaluation metric (higher is better).

Method	Family	Francis	Horse	Light house	M60	Panther	Playground	Train	Mean
MVSNet [42]	55.99	28.55	25.07	50.79	53.96	50.86	47.90	34.69	43.48
R-MVSNet [43]	69.96	46.65	32.59	42.95	51.88	48.80	52.00	42.38	48.40
PVA-MVSNet [45]	69.36	46.80	46.01	55.74	57.23	54.75	56.70	49.06	54.46
CVP-MVSNet [41]	76.50	47.74	36.34	55.12	57.28	54.28	57.43	47.54	54.03
CasMVSNet [13]	76.37	58.45	46.26	55.81	56.11	54.06	58.18	49.51	56.84
UCS-Net [8]	76.09	53.16	43.03	54.00	55.60	51.49	57.38	47.89	54.83
D2HC-RMVSNet [40]	74.69	56.04	**49.42**	**60.08**	59.81	59.61	**60.04**	**53.92**	59.20
U-MVS [37]	76.49	60.04	49.20	55.52	55.33	51.22	56.77	52.63	57.15
Ours	**78.55**	**61.93**	45.48	57.59	**61.00**	**59.78**	59.19	52.32	**59.48**

Table 3. Ablation studies. The performance under distance metric is reported (lower is better).

Method	Accuracy	Completeness	Overall
w/o epipolar transformer	0.347	0.339	0.343
w/o 2D image feature	0.345	0.352	0.348
w/o 3D volume feature	0.434	0.322	0.378
vis-max feature aggregation	0.345	0.331	0.338
Global implicit field	0.573	0.642	0.608
Ray with Transformer	**0.339**	0.343	0.341
Ray with average pooling	0.356	0.406	0.381
Ray with max pooling	0.466	0.383	0.424
w/o SDF prediction	0.354	0.330	0.342
Ours	0.341	**0.319**	**0.330**



(a) No SDF prediction (b) The full method (c) Scene

Figure 9. Mid-layer feature map t-SNE visualization of the w/o SDF prediction baseline (a) and the full method (b) for the green segment marked in the scenes in (c).

4.3. Ablation Study

In Table 3, we conduct ablation studies to quantify the efficacy of several crucial components in RayMVSNet.

Feature aggregation. The cross-view feature aggregation is a key component of RayMVSNet. To evaluate the importance, we compare the full method to several baselines without some specific component: *w/o epipolar transformer*, *w/o 2D image feature* and *w/o 3D volume feature*. It clearly shows that all these baselines make the performance decline. *It is worth noting that w/o epipolar transformer* achieves a lower completeness score, indicating epipolar transformer could make the reconstruction complete by providing more reliable cross-view correlations. We also compare our epipolar transformer to other multi-view feature aggregation method. In the experiment of *vis-max feature aggregation*, we replace the epipolar transformer with the visibility-aware max-pooling feature aggregation [6]. The result indicates epipolar transformer is a better solution.

Global implicit field. Our method learns the local 1D implicit field by splitting the scene into a bunch of rays. To show its necessity, a straightforward baseline is to learn a global implicit field in the reference frustum directly, such that there is no ray-based representation. This baseline adopts the same cross-view feature aggregation as the full method, and predicts the SDF of all points in the reference frustum by using an MLP. The depth map is then generated by a ray-casting algorithm from the predicted SDFs. Unsurprisingly, experiments show this network is hard to converge and leads to low quantitative performance, which suggests that the ray-based 1D implicit field indeed simplifies the learning and is suitable to the MVS problem.

Other ray-based implicit field models. In order to reveal the need of the proposed LSTM, we compare our method against several baselines with alternative models of processing sequential data. To be specific, we study the effects of replacing the LSTM with average pooling, max pooling, and Transformer [34], respectively. The *Ray with average pooling* and the *Ray with max pooling* baselines aggregate ray feature by average pooling and max pooling over all sampled points, respectively. The aggregated features are then used to predict the zero-crossing location. The point-wise SDF predictions are also performed as an auxiliary task. The result shows that our method outperforms all the baselines. In particular, the performance drops significantly with the *Ray with average pooling* and the *Ray with max pooling*, implying that the modeling of ray-based 1D implicit field is a non-trivial task. The *Ray with Transformer* is inferior to the full method, in terms of the *Overall score*, confirming that LSTM is more appropriate to our problem.

No SDF prediction. The SDF prediction is an auxiliary task in RayMVSNet. We demonstrate its influence by turning it off and comparing to the full method. The perfor-

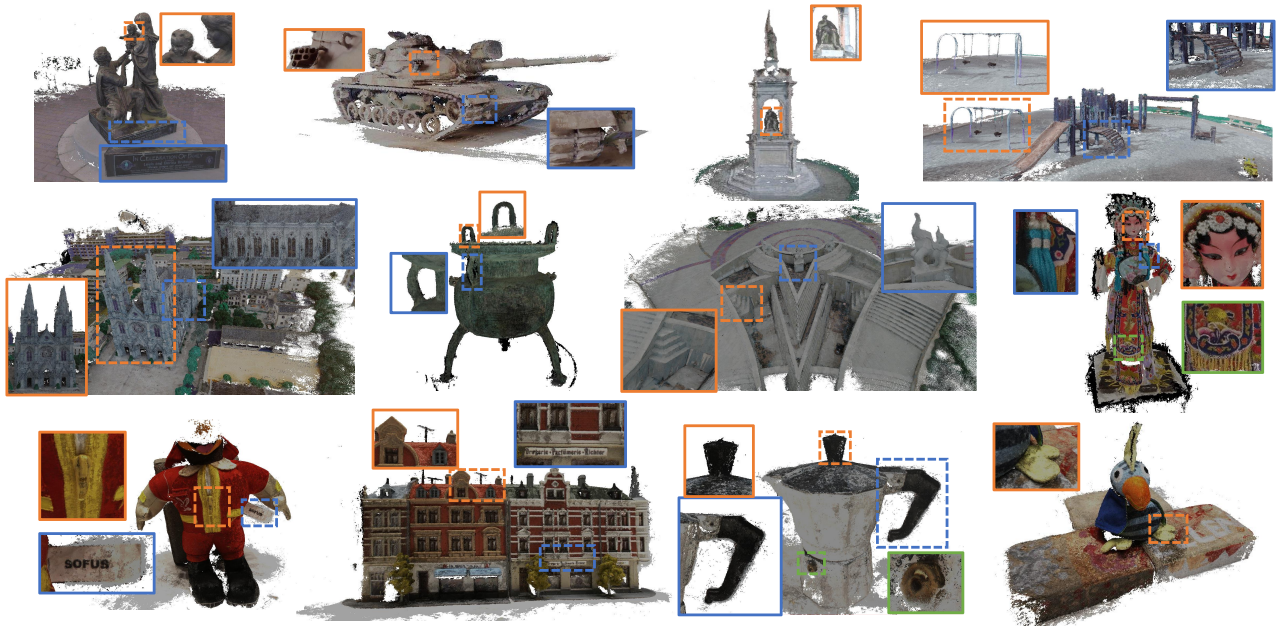


Figure 10. Gallery of the reconstructed point cloud on *Tanks & temples* (top row), *BlendedMVS* (middle row) and *DTU* (bottom row).

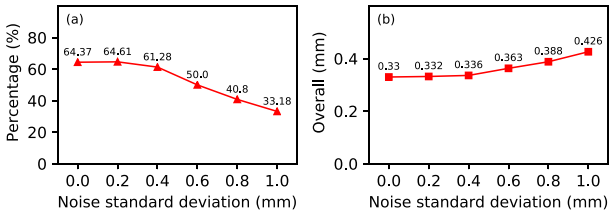


Figure 11. Sensitivity to coarse depth quality. The percentage of pixel-wise depth predictions whose error is smaller than $1mm$ (a) and the overall score of point cloud reconstruction (b) are reported.

mance of *w/o SDF prediction* baseline is inferior to the full method, demonstrating the joint training of SDF prediction and zero-crossing position prediction is indeed helpful, due to the extra supervision of SDF. Examples are visualized in Figure 9 which compares the mid-layer features of the full model and the baseline without SDF prediction. We can see that the mid-layer features of the full method, with SDF supervision, maintain a better monotonicity along the ray direction, resulting in more accurate predictions.

4.4. Sensitivity to coarse depth quality

We show our RayMVSNet is robust to the incorrectness of coarse depth prediction with a pressure test. In the experiment, we add Gaussian noise to the predicted coarse depth maps, during both the training and testing phases. We report the performance of the depth map prediction and the point cloud reconstruction on *DTU*. Figure 11 shows RayMVSNet is robust to moderate perturbation (noise standard deviation $\leq 0.4mm$). It is interesting to see that the quality of depth map prediction slightly increases when moderate noise is added. This demonstrates that data augmentation such as modest perturbation to coarse depth is helpful for training a more generalizable RayMVSNet.

4.5. Qualitative results

We visualize the qualitative results of RayMVSNet on several datasets in Figure 10. Note that RayMVSNet is able to reconstruct large-scale scenes with fine-grained geometry details, such as the highlighted regions.

5. Conclusion

We have presented RayMVSNet, which learns to directly optimize the depth value along each camera ray. An epipolar transformer is designed to enable sequential modeling of 1D ray-based implicit fields, which essentially mimics the epipolar line search in traditional MVS. The ray-based approach demonstrates significant performance boost with only a low-res cost volume. An interesting future direction is to further enhance the ray-based deep MVS approach so that cost volume convolution could be completely saved. In most deep MVS works, 3D point cloud is recovered from the estimated depth map as a post-processing. We would like to study end-to-end optimization of 3D point clouds [30].

Acknowledgements

We thank the anonymous reviewers for their valuable comments. This work was supported in part by NSFC (62132021, 62102435, 62002379, U20A20185, 61972435), National Key Research and Development Program of China (2018AAA0102200) and the Zhejiang Lab's International Talent Fund for Young Professionals.

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, 6

[2] Alex M Andrew. Multiple view geometry in computer vision. *Kybernetes*, 2001. 1

[3] Rohan Chabra, Jan E Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *European Conference on Computer Vision*, pages 608–625. Springer, 2020. 3

[4] Anpei Chen, Zexiang Xu, Fuqiang Zhao, Xiaoshuai Zhang, Fanbo Xiang, Jingyi Yu, and Hao Su. Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. *arXiv preprint arXiv:2103.15595*, 2021. 2

[5] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1538–1547, 2019. 2, 6

[6] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Visibility-aware point-based multi-view stereo network. *IEEE transactions on pattern analysis and machine intelligence*, 43(10):3695–3708, 2020. 2, 7

[7] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2

[8] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 2524–2534, 2020. 1, 2, 3, 6, 7

[9] Yu Deng, Jiaolong Yang, and Xin Tong. Deformed implicit field: Modeling 3d shapes with learned dense correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10286–10296, 2021. 2

[10] Yueqi Duan, Haidong Zhu, He Wang, Li Yi, Ram Nevatia, and Leonidas J Guibas. Curriculum deepsdf. In *European Conference on Computer Vision*, pages 51–67. Springer, 2020. 3

[11] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 873–881, 2015. 6

[12] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3d shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4857–4866, 2020. 3

[13] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020. 1, 2, 3, 6, 7

[14] Wilfried Hartmann, Silvano Galliani, Michal Havlena, Luc Van Gool, and Konrad Schindler. Learned multi-patch similarity. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1586–1594, 2017. 2

[15] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 2, 4

[16] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2821–2830, 2018. 2

[17] 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. 4

[18] 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. 2

[19] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, Thomas Funkhouser, et al. Local implicit grid representations for 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6001–6010, 2020. 3

[20] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017. 6

[21] 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/CVF International Conference on Computer Vision*, pages 10452–10461, 2019. 2, 6

[22] Keyang Luo, Tao Guan, Lili Ju, Yuesong Wang, Zhuo Chen, and Yawei Luo. Attention-aware multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1590–1599, 2020. 2, 6

[23] Xinjun Ma, Yue Gong, Qirui Wang, Jingwei Huang, Lei Chen, and Fan Yu. Epp-mvsnet: Epipolar-assembling based depth prediction for multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5732–5740, 2021. 2

[24] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4460–4470, 2019. 2

[25] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020. 2, 3

[26] Henry J Nelson and Nikolaos Papanikolopoulos. Learning continuous object representations from point cloud data. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2446–2451. IEEE, 2020. 2

[27] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 3

[28] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019. 2, 6

[29] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, pages 523–540. Springer, 2020. 2

[30] Yifei Shi, Junwen Huang, Hongjia Zhang, Xin Xu, Szymon Rusinkiewicz, and Kai Xu. Symmetrynet: learning to predict reflectional and rotational symmetries of 3d shapes from single-view rgb-d images. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020. 8

[31] Vincent Sitzmann, Eric R Chan, Richard Tucker, Noah Snavely, and Gordon Wetzstein. Metasdf: Meta-learning signed distance functions. *arXiv preprint arXiv:2006.09662*, 2020. 3

[32] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson, Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11358–11367, 2021. 2, 3

[33] Edgar Tretschk, Ayush Tewari, Vladislav Golyanik, Michael Zollhöfer, Carsten Stoll, and Christian Theobalt. Patchnets: Patch-based generalizable deep implicit 3d shape representations. In *European Conference on Computer Vision*, pages 293–309. Springer, 2020. 3

[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 4, 7

[35] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14203, 2021. 2, 6

[36] Zizhuang Wei, Qingtian Zhu, Chen Min, Yisong Chen, and Guoping Wang. Aa-rmvsnet: Adaptive aggregation recurrent multi-view stereo network. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6187–6196, 2021. 2

[37] Hongbin Xu, Zhipeng Zhou, Yali Wang, Wenxiong Kang, Baigui Sun, Hao Li, and Yu Qiao. Digging into uncertainty in self-supervised multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6078–6087, 2021. 2, 6, 7

[38] 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*, volume 34, pages 12508–12515, 2020. 2, 6

[39] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. *arXiv preprint arXiv:1905.10711*, 2019. 2

[40] Jianfeng Yan, Zizhuang Wei, Hongwei Yi, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Dense hybrid recurrent multi-view stereo net with dynamic consistency checking. In *European Conference on Computer Vision*, pages 674–689. Springer, 2020. 2, 7

[41] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 4877–4886, 2020. 1, 2, 3, 6, 7

[42] 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 (ECCV)*, pages 767–783, 2018. 1, 2, 3, 4, 5, 6, 7

[43] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 5525–5534, 2019. 2, 6, 7

[44] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020. 6

[45] Hongwei Yi, Zizhuang Wei, Mingyu Ding, Runze Zhang, Yisong Chen, Guoping Wang, and Yu-Wing Tai. Pyramid multi-view stereo net with self-adaptive view aggregation. In *European Conference on Computer Vision*, pages 766–782. Springer, 2020. 2, 7

[46] Anzhu Yu, Wenyue Guo, Bing Liu, Xin Chen, Xin Wang, Xuefeng Cao, and Bingchuan Jiang. Attention aware cost volume pyramid based multi-view stereo network for 3d reconstruction. *ISPRS Journal of Photogrammetry and Remote Sensing*, 175:448–460, 2021. 2, 6

[47] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 1949–1958, 2020. 2, 6

[48] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network. *arXiv preprint arXiv:2008.07928*, 2020. 2

[49] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6525–6534, 2021. 2

[50] Xudong Zhang, Yutao Hu, Haochen Wang, Xianbin Cao, and Baochang Zhang. Long-range attention network for multi-view stereo. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3782–3791, 2021. 2

[51] Yawei Zhao, Kai Xu, En Zhu, Xinwang Liu, Xinzhong Zhu, and Jianping Yin. Triangle lasso for simultaneous clustering and optimization in graph datasets. *IEEE Transactions on Knowledge and Data Engineering*, 31(8):1610–1623, 2018. 2

[52] Zerong Zheng, Tao Yu, Qionghai Dai, and Yebin Liu. Deep implicit templates for 3d shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1429–1439, 2021. 2