

VolRecon: Volume Rendering of Signed Ray Distance Functions for Generalizable Multi-View Reconstruction

Yufan Ren¹ *Fangjinhua Wang² *Tong Zhang¹Marc Pollefeys^{2,3}Sabine Süsstrunk¹¹School of Computer and Communication Sciences, EPFL²Department of Computer Science, ETH Zurich ³ Microsoft

Abstract

With the success of neural volume rendering in novel view synthesis, neural implicit reconstruction with volume rendering has become popular. However, most methods optimize per-scene functions and are unable to generalize to novel scenes. We introduce VolRecon, a generalizable implicit reconstruction method with Signed Ray Distance Function (SRDF). To reconstruct with fine details and little noise, we combine projection features, aggregated from multi-view features with a view transformer, and volume features interpolated from a coarse global feature volume. A ray transformer computes SRDF values of all the samples along a ray to estimate the surface location, which are used for volume rendering of color and depth. Extensive experiments on DTU and ETH3D demonstrate the effectiveness and generalization ability of our method. On DTU, our method outperforms SparseNeuS by about 30% in sparse view reconstruction and achieves comparable quality as MVSNet in full view reconstruction. Besides, our method shows good generalization ability on the large-scale ETH3D benchmark. Project page: <https://fangjinhuawang.github.io/VolRecon>.

1. Introduction

Numerous applications in robotics [52], augmented/virtual reality [29, 35], and autonomous driving [16, 43] depend on the ability to reconstruct 3D geometry from a set of images or video frames. Multi-view stereo (MVS) methods [13, 15, 39, 47, 54, 55] are often used to solve this task. Their pipelines usually consist of multi-view depth estimation, filtering, and fusion [5, 13]. Recently, neural implicit representations also achieve impressive performances on various tasks, e.g., shape modeling [28, 36], novel view synthesis [30], and surface reconstruction [49, 56].

Neural Radiance Fields (NeRF) [30] implicitly model a

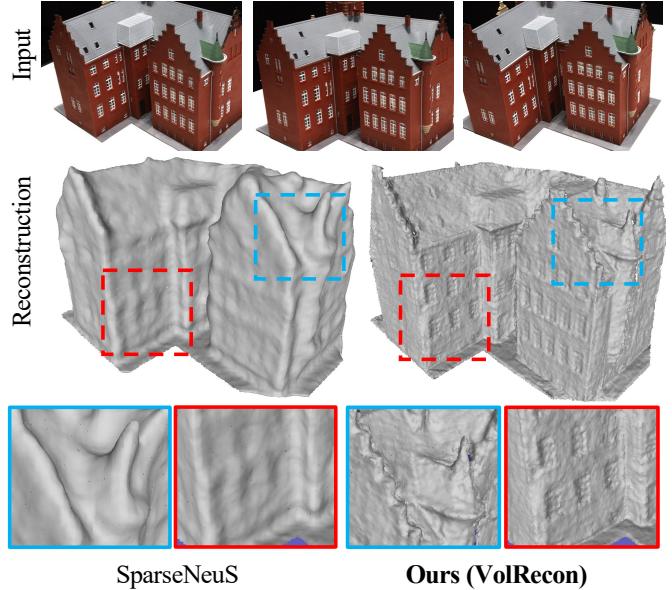


Figure 1. Visualization of generalizable implicit reconstructions from three views. **Our method (VolRecon)** reconstructs surfaces with finer details (middle right) compared with the state-of-the-art SparseNeuS [26] (middle left). Best viewed on a screen when zoomed in.

radiance field with Multi-layer Perceptrons (MLP) to output density and radiance at a given position and viewing direction. NeRF then uses volume rendering to render the image from a given viewpoint. However, NeRF cannot generate accurate surface reconstruction [56]. To improve the reconstruction quality, several works [49, 56] use Signed Distance Function (SDF) to represent the geometry and model the density function for volume rendering. However, color-only supervision for SDF usually results in unsatisfactory reconstruction compared with MVS methods [15, 54], due to the lack of geometry supervision and radiance-geometry ambiguity [51, 62]. Many follow-up works thus use additional priors to improve the reconstruction quality, e.g., sparse Structure-from-Motion (SfM) point clouds [11], dense MVS point clouds [60], normals [48, 59] and depth maps [59]. How-

*Equal contribution

ever, most of them require per-scene optimization and are incapable of generalizing to unseen scenes.

The generalization ability of learning-based methods is essential for practical applications. However, the network cannot generate unseen scenes by simply taking the spatial coordinates as input. Therefore, retrieving the information from the scene, such as the points’ features on the corresponding images, becomes a natural way of achieving such generalization [4, 50]. SparseNeuS [26] recently achieved across-scene generalization in implicit reconstruction with global feature volumes [4], where each voxel stores a feature vector. It predicts SDF by forwarding the coordinates and corresponding feature vectors retrieved from the feature volumes to an MLP. SparseNeuS highly depends on the resolution of the feature volume and uses a two-stage coarse-to-fine training to achieve details in reconstruction. Due to the memory constraints [22, 32], which make high-resolution feature volumes computationally expensive, it usually yields over-smoothing surfaces, as shown in Fig. 1.

We propose VolRecon, a novel framework for generalizable neural implicit reconstruction with Signed Ray Distance Function (SRDF). Unlike SDF, which defines the distance to the nearest surface along any directions, SRDF [63] defines the distance to the nearest surface along a given ray. We project each point on the ray into source views and aggregate the interpolated features into *projection features* with a view transformer. However, only using the projection features, which capture local information, may not be enough to accurately decide the surface location along the ray due to challenging situations like occlusions. Similar to [32], we thus construct a coarse global feature volume to encode global information, where we obtain the interpolated features as *volume features*. Considering *projection features* and *volume features* of all the points along the ray, a ray transformer then estimates the surface location and computes the SRDF values for all points. Following NeuS [49], we model the density function with SRDF and then estimate the image and depth map with volume rendering.

Extensive experiments on DTU [1] and ETH3D [40] verify the effectiveness and generalization ability of our method. On DTU, our method outperforms the state-of-the-art method SparseNeuS [26] by about 30% in sparse view reconstruction and 22% in full view reconstruction. Compared with MVSNet [54], a seminal learning-based MVS method, our method performs better in depth evaluation and performs comparably in full view reconstruction. On ETH3D, we show that our method has good generalization ability on large-scale scenes.

In conclusion, our contributions are as follows:

- We propose a new pipeline for generalizable implicit reconstruction that produce detailed surfaces.
- We propose a novel framework that comprises a view

transformer to aggregate multi-view features and a ray transformer to compute SRDF values of all the points along a ray.

- We use a combination of local projection features and global volume features to produce surface reconstructions with fine details and high quality.

2. Related Work

Neural Implicit Reconstruction. Traditional volumetric reconstruction [5, 18, 33] uses implicit signed distance fields to produce high-quality reconstructions. Recent works use networks to model shapes as continuous decision boundaries, i.e., occupancy function [28, 37] or SDF [36]. In NeRF [30], the authors further show that combining neural implicit functions, e.g., Multi-Layer Perceptron (MLP), together with volume rendering can achieve photo-realism in novel view synthesis [2, 3, 10, 30, 31]. Since NeRF [30], which originally targets a per-scene optimization problem, several additional methods [4, 50, 58] are proposed to perform generalizable novel view synthesis for unseen scenes. For example, IBR-Net [50] projects sampled points along the ray into multiple source views, aggregates multi-view features into density features, and uses a ray transformer, which takes as input the density features for all points along the ray to predict the density for each point. For multi-view reconstruction, IDR [57] reconstructs surfaces by representing the geometry as the zero-level set of an MLP, requiring accurate object masks. To avoid using masks, VolSDF [56] and NeuS [49] incorporate SDF in neural volume rendering, using it to modify the density function. Additional geometric priors [11, 48, 59, 60] were proposed to improve the reconstruction quality. However, these methods usually require a lengthy optimization process for each scene and cannot generalize to unseen scenes.

Recently, SparseNeuS [26] attempts to solve across-scene generalization for surface reconstruction. Similar to [4, 32, 42], SparseNeuS constructs feature volumes with resolution up to 192^3 to aggregate image features from multi-view images. To predict the SDF, an MLP takes as input the coordinates and corresponding interpolated features from the feature volumes. However, SparseNeuS needs high-resolution volumes to reconstruct surfaces with fine details, which is infeasible for large-scale scene reconstructions due to the high memory consumption. In contrast, we reformulate the feature representation in a fixed resolution volume through an implicit function, which allows us to adaptively sample the point, resulting in a high detailed reconstruction without using equal-or-higher volume resolution. In this way, our VolRecon model captures both local and global information to achieve finer detail and less noise compared with SparseNeuS, Fig. 1.

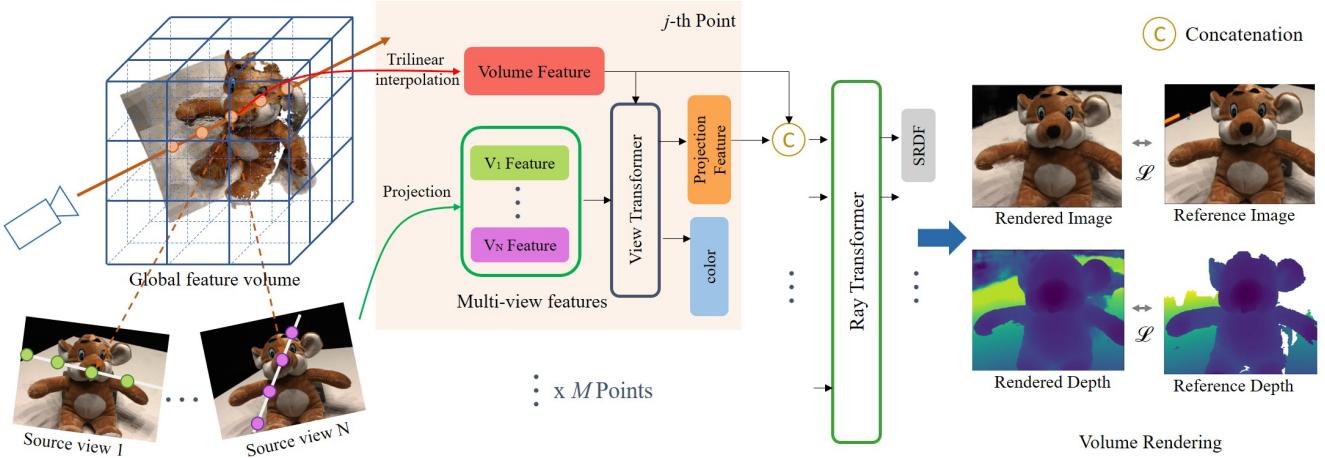


Figure 2. **Detailed structure of VolRecon.** First, for a set of source views, we extract the image features. Then we build a global feature volume to obtain global shape priors. Next, given a ray in the target viewpoint, we project each point into the source views, aggregate its multi-view features with a view transformer to output its color and projection feature. Then we apply the ray transformer to features of all the M points along the ray to predict their SRDF values. Finally, we use volume rendering to render the color and depth. Best viewed on a screen when zoomed in.

Multi-view Stereo. Based on the scene representations, traditional MVS methods can be divided into three main categories: volumetric [22, 23, 41], point cloud based [12, 24] and depth map based [13, 39, 53]. Depth map based methods are considered to be more flexible since they decouple the problem into depth map estimation and fusion [13, 39]. Therefore, most recent learning-based MVS methods [15, 46, 47, 54, 55, 61] perform multi-view depth estimation and then fuse them to a point cloud, which achieves impressive performance on various benchmarks [1, 21, 40]. Note that though there is a lot of progress made in neural implicit reconstruction, the reconstruction performance is still not up to par [49, 56, 57] with the traditional MVS method, COLMAP [39]. We compare our method with MVSNet [54], a seminal learning-based MVS method that outperforms COLMAP [39]. We show that our method is able to achieve comparable reconstruction performance as the MVS methods.

3. Method

In this section, we discuss the detailed structure of VolRecon, illustrated in Fig. 2. The pipeline consists of predicting the Signed Ray Distance Function (SRDF) (Sec. 3.1), volume rendering of the SRDF to predict color and depth (Sec. 3.2), and the loss functions (Sec. 3.3) for training.

3.1. SRDF Prediction

Signed Ray Distance Function. Let the set $\Omega \in \mathbb{R}^3$ define the occupied space and $\mathcal{M} = \partial\Omega$ its boundary surface. The Signed Distance Function $d_\Omega(\mathbf{p})$ defines the shortest dis-

tance of point $\mathbf{p} \in \mathbb{R}^3$ to the surface \mathcal{M} . Its sign denotes whether \mathbf{p} is outside (positive) or inside (negative) of the surface,

$$\mathbf{1}_\Omega(\mathbf{p}) = \begin{cases} 1 & \text{if } \mathbf{p} \in \Omega \\ 0 & \text{if } \mathbf{p} \notin \Omega \end{cases}, \quad (1)$$

$$d_\Omega(\mathbf{p}) = (-1)^{\mathbf{1}_\Omega(\mathbf{p})} \min_{\mathbf{p}^* \in \mathcal{M}} \|\mathbf{p} - \mathbf{p}^*\|_2, \quad (2)$$

where $\|\cdot\|_2$ is the 2-norm and \mathbf{p}^* are points on the surface. Differently, SRDF [63] defines the shortest distance to surface \mathcal{M} along a ray direction \mathbf{v} ($\|\mathbf{v}\|_2 = 1$),

$$\tilde{d}_\Omega(\mathbf{p}, \mathbf{v}) = (-1)^{\mathbf{1}_\Omega(\mathbf{p})} \min_{\mathbf{p}^* \in \mathcal{M}, \frac{\mathbf{p}^* - \mathbf{p}}{\|\mathbf{p}^* - \mathbf{p}\|_2} = \mathbf{v}} \|\mathbf{p} - \mathbf{p}^*\|_2. \quad (3)$$

Note that the concept of SRDF is used in previous classical methods [5]. Theoretically, given a point \mathbf{p} , its SDF $d_\Omega(\mathbf{p})$ equals to the SRDF $\tilde{d}_\Omega(\mathbf{p}, \mathbf{v})$ with minimum absolute value in any ray direction \mathbf{v} :

$$d_\Omega(\mathbf{p}) = (-1)^{\mathbf{1}_\Omega(\mathbf{p})} \min_{\mathbf{v}} |\tilde{d}_\Omega(\mathbf{p}, \mathbf{v})|. \quad (4)$$

Similar to [49, 56] that use SDF in volume rendering, we incorporate SRDF in volume rendering to estimate the depth map from the given viewpoints, which can be fused into mesh [5] or dense point clouds [39].

Feature Extraction. Given the source image set $\mathbb{I} = \{\mathbf{I}_1, \dots, \mathbf{I}_N\}$, where $\mathbf{I} \in \mathbb{D}^{H \times W \times 3}$, $\mathbb{D} \subset [0, 1]$, and H, W are the image height and width, respectively. We use

a Feature Pyramid Network [25] to extract feature maps $\{\mathbf{F}_i\}_{i=1}^N \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4} \times C}$.

Global Feature Volume. To get global information, we construct a global feature volume \mathbf{F}_v similar to [32, 42]. Specifically, we first divide the bounding volume of the scene into K^3 voxels. Each voxel is projected onto source views to obtain the features. This is done with bilinear interpolation, where mean and variance are computed and concatenated as the voxel features. A 3D U-Net [38] then regularizes and aggregates the information. For each point \mathbf{p} , we denote the interpolated feature from \mathbf{F}_v as *volume feature*, \mathbf{f}_v .

View Transformer. Given a pixel in the reference view, we denote the M points on the ray emitted from this pixel as $\{\mathbf{p}(t) = \mathbf{o} + t\mathbf{v}, t \geq 0\}$. By projecting each point \mathbf{p} onto the source view, we extract colors $\{\mathbf{c}_i\}_{i=1}^N$ and features $\{\mathbf{f}_i\}_{i=1}^N$ using bilinear interpolation. We concatenate \mathbf{f}_i with the volume feature \mathbf{f}_v to include the global prior. We apply a *view transformer* to aggregate the multi-view features $\{\mathbf{f}_i\}_{i=1}^N$ into one feature, which we denote as *projection feature*. Structurally, we use a self-attention transformer [44] with linear attention [19]. Based on previous work [7], we add a learnable aggregation token, denoted as \mathbf{f}_0 , to obtain the projection feature. Since no order of source views is assumed, we do not use position encoding in view transformer. The projection feature \mathbf{f}_p and updated multi-view features $\{\mathbf{f}'_i\}_{i=1}^N$ are computed as,

$$\mathbf{f}_p, \{\mathbf{f}'_i\}_{i=1}^N = \text{ViewTrans}(\mathbf{f}_0, \{\mathbf{f}_i\}_{i=1}^N). \quad (5)$$

Visibility is important in multi-view aggregation [39, 47] due to the existence of occlusions. Therefore, using mean or variance to compute aggregated features [50, 54] may not be robust enough since all views are accounted equally. Using a learnable transformer enables the model to reason about the consistency for aggregation across multiple views.

Ray Transformer. To enable each point to aggregate information from other points along the ray, we additionally design *ray transformer*, based on linear attention [19]. After ordering the points in a sequence from near to far, the ray transformer applies position encoding [50] and self attention on the combination of projection features and volume features to predict attended features $\{\tilde{\mathbf{f}}_j\}_{j=1}^M$,

$$\{\tilde{\mathbf{f}}_j\}_{j=1}^M = \text{RayTrans}(\{\text{cat}(\mathbf{f}_v, \mathbf{f}_p, \gamma)\}_{j=1}^M), \quad (6)$$

where $\text{cat}(\cdot)$ denotes concatenation and γ position encoding. Finally, we use an MLP to decode the attended feature to SRDF for each point on the ray.

3.2. Volume Rendering of SRDF

Color Blending. For a point \mathbf{p} at viewing direction \mathbf{v} , we blend colors of N source views, $\{\mathbf{c}_i\}_{i=1}^N$, similar to [45, 50].

The blending weight is computed using the updated multi-view features $\{\mathbf{f}'_i\}_{i=1}^N$ from the view transformer. Similar to [45, 50], we concatenate $\{\mathbf{f}'_i\}_{i=1}^N$ with the difference between \mathbf{v} and the viewing direction in the i -th source view, \mathbf{v}_i . Then we pass the concatenated features through an MLP and use *Softmax* to get the blending weights $\{\eta_i\}_{i=1}^N$. The final radiance at point \mathbf{p} and viewing direction \mathbf{v} is the weighted sum of $\{\mathbf{c}_i\}_{i=1}^N$,

$$\hat{\mathbf{c}} = \sum_{i=1}^N \eta_i \cdot \mathbf{c}_i. \quad (7)$$

Volume rendering. Several works [49, 56] propose to include SDF in volume rendering for implicit reconstruction with the supervision of the pixel reconstruction loss. We adopt the method of NeuS [49] to volume render SRDF, as briefly introduced below. For more details we refer the reader to the supplementary.

Specifically, the color is accumulated along the ray

$$\hat{\mathbf{C}} = \sum_{j=1}^M T_j \alpha_j \hat{\mathbf{c}}_j, \quad (8)$$

where $T_j = \prod_{k=1}^{j-1} (1 - \alpha_k)$ is the discrete *accumulative transmittance*, and α_j are discrete opacity values defined by

$$\alpha_j = 1 - \exp(- \int_{t_j}^{t_{j+1}} \rho(t) dt), \quad (9)$$

where opaque density $\rho(t)$ is similar to the original definition in NeuS [49]. The difference is that we replace the original SDF with SRDF in $\rho(t)$. For more theoretical details, please refer to [49].

Similar to color volume rendering, we can derive the rendered depth as

$$\hat{\mathbf{D}} = \sum_{j=1}^M T_j \alpha_j t_j. \quad (10)$$

3.3. Loss Function

We define our loss function as

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \alpha \mathcal{L}_{\text{depth}}. \quad (11)$$

The color loss $\mathcal{L}_{\text{color}}$ is defined as

$$\mathcal{L}_{\text{color}} = \frac{1}{S} \sum_{s=1}^S \left\| \hat{\mathbf{C}}_s - \mathbf{C}_s \right\|_2, \quad (12)$$

where S is the number of pixels and \mathbf{C}_s is the ground truth color. The depth loss $\mathcal{L}_{\text{depth}}$ is defined as

$$\mathcal{L}_{\text{depth}} = \frac{1}{S_1} \sum_{s=1}^{S_1} |\hat{\mathbf{D}}_s - \mathbf{D}_s|, \quad (13)$$

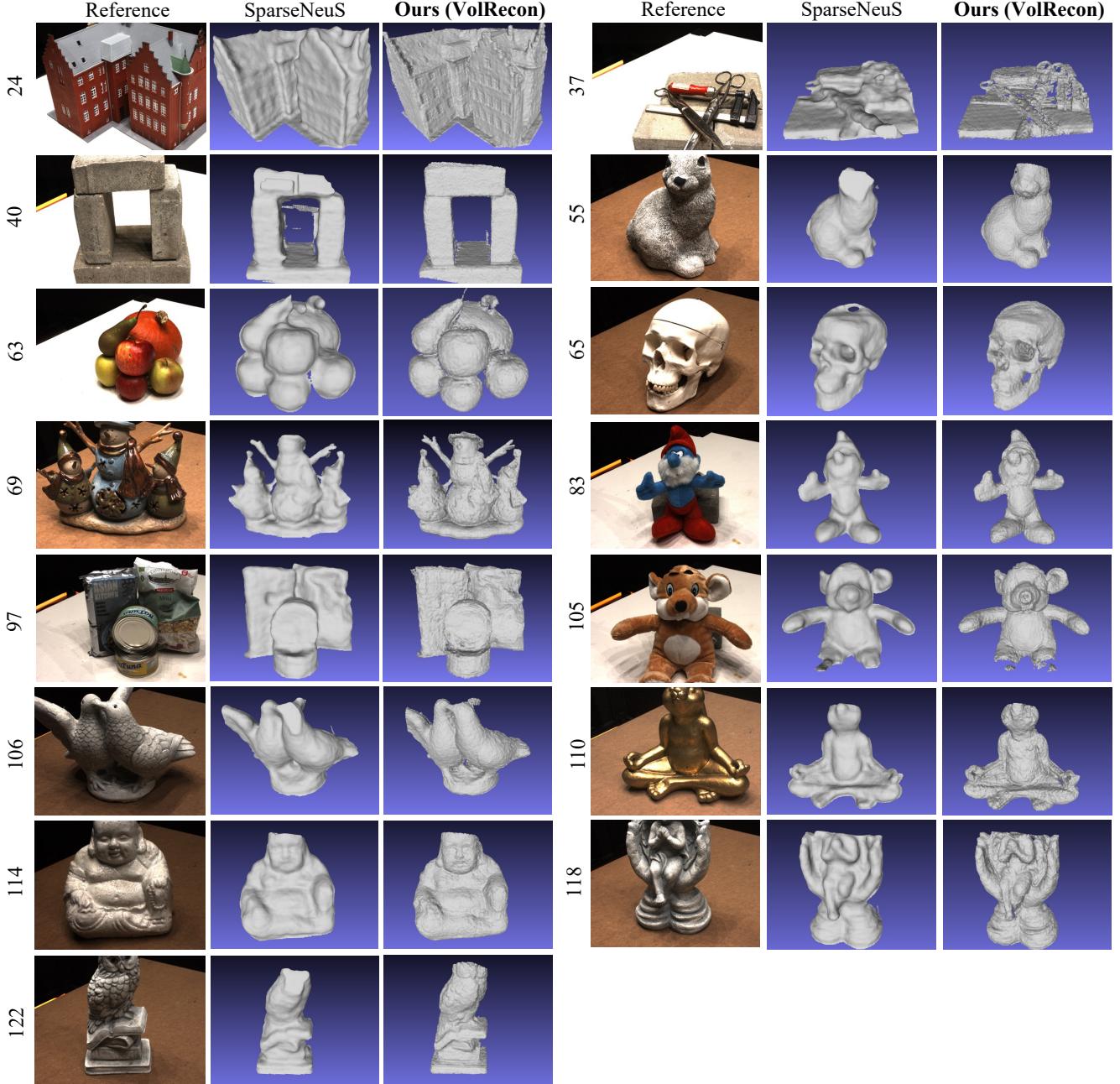


Figure 3. Visualization of **sparse view** ($N = 3$) reconstruction on 15 testing scenes in the DTU dataset [1]. Our method (VolRecon) produces finer details (e.g., scene 24 windows, scene 40 top brick and scene 63 fruit stalks) and sharper boundaries (e.g., scene 97 cans touching part, scene 118 sculpture base, and scene 122 owl wings) in reconstruction than SparseNeuS [26]. Best viewed on a screen when zoomed in.

where S_1 is the number of pixels with valid depth and D_s is the ground truth depth. In our experiments, we choose $\alpha = 1.0$.

4. Experiments

4.1. Experimental Settings

Datasets. Following existing works [11, 26, 49, 56], we mainly use the DTU dataset [1] for training and evaluation. The DTU dataset [1] is an indoor multi-view stereo dataset with 124 different scenes and 7 different lighting conditions.

Scan	Mean↓	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122
COLMAP [39]	<u>1.52</u>	0.90	<u>2.89</u>	<u>1.63</u>	1.08	<u>2.18</u>	<u>1.94</u>	<u>1.61</u>	1.30	<u>2.34</u>	<u>1.28</u>	1.10	<u>1.42</u>	<u>0.76</u>	1.17	1.14
IDR [57]	3.39	4.01	6.40	3.52	1.91	3.96	2.36	4.85	1.62	6.37	5.97	1.23	4.73	0.91	1.72	1.26
VolSDF [56]	3.41	4.03	4.21	6.12	0.91	8.24	1.73	2.74	1.82	5.14	3.09	2.08	4.81	0.60	3.51	2.18
UNISURF [34]	4.39	5.08	7.18	3.96	5.30	4.61	2.24	3.94	3.14	5.63	3.40	5.09	6.38	2.98	4.05	2.81
NeuS [49]	4.00	4.57	4.49	3.97	4.32	4.63	1.95	4.68	3.83	4.15	2.50	1.52	6.47	1.26	5.57	6.11
PixelNeRF [58]	6.18	5.13	8.07	5.85	4.40	7.11	4.64	5.68	6.76	9.05	6.11	3.95	5.92	6.26	6.89	6.93
IBRNet [50]	2.32	2.29	3.70	2.66	1.83	3.02	2.83	1.77	2.28	2.73	1.96	1.87	2.13	1.58	2.05	2.09
MVSNeRF [4]	2.09	1.96	3.27	2.54	1.93	2.57	2.71	1.82	1.72	2.29	1.75	1.72	1.47	1.29	2.09	2.26
SparseNeuS [26]	1.96	2.17	3.29	2.74	1.67	2.69	2.42	1.58	1.86	1.94	1.35	1.50	1.45	0.98	1.86	1.87
Ours (VolRecon)	1.38	<u>1.20</u>	2.59	1.56	1.08	1.43	1.92	1.11	<u>1.48</u>	1.42	1.05	<u>1.19</u>	1.38	0.74	<u>1.23</u>	<u>1.27</u>

Table 1. Quantitative results of **sparse view** reconstruction on 15 testing scenes of DTU dataset [1]. We report Chamfer distance (lower is better). Methods are separated into four categories (from top to bottom): 1) Multi-view Stereo (MVS), 2) per-scene optimization based neural implicit reconstruction methods, 3) generalizable neural rendering methods, and 4) generalizable neural implicit reconstruction methods. Best scores are in **bold** and second best are underlined.

During experiments, we use 15 scenes [57] for testing and the remaining scenes for training. The ground truth depth maps are rendered from the mesh [54]. We use ETH3D [40] benchmark to test the generalization ability of our method. ETH3D [40] is a challenging MVS benchmark that consists of high-resolution images of real-world large-scale scenes with strong viewpoint variations.

Implementation details. We implement our model in PyTorch [17] and PyTorch Lightning [9]. During training, we use an image resolution of 640×512 and set the number of source images to $N = 4$. We train our model for 16 epochs using Adam [20] on one A100 GPU. The learning rate is set to 10^{-4} . The ray number sampled per batch and the batch size are set to 1024 and 2 respectively. Note that similar to other volume rendering methods [30, 49], we use a hierarchical sampling strategy in both training and testing. We first uniformly sample N_{coarse} points on the ray and then conduct importance sampling to sample another N_{fine} points on top of the coarse probability estimation. We set $N_{\text{coarse}} = 64$ and $N_{\text{fine}} = 64$ during our experiments. For global feature volume \mathbf{F}_v , we set the resolution as $K = 96$. During testing, we set the image resolution to 800×600 .

Baselines. We mainly compare our method with: (1) SparseNeuS [26], the state-of-the-art generalizable neural implicit reconstruction method; note that we report reproduced results using their official repository and the released model checkpoint¹ by the date of submission; (2) generalizable neural rendering methods [4, 50, 58]; (3) per-scene optimization based neural implicit reconstruction methods [34, 49, 56, 57]; (4) MVS methods [39, 54]. We train MVSNet [54] with our training split for 16 epochs.

4.2. Evaluation Results

Sparse View Reconstruction on DTU. On DTU [1], we conduct sparse reconstruction with 3 views only. For a fair

¹<https://github.com/xxlong0/SparseNeuS>

comparison, we use the same image sets and evaluation process as SparseNeuS [26]. For each view, we generate a virtual rendering viewpoint by shifting the original camera coordinate frame for $d = 25mm$ along the x -axis. After rendering the depth maps, we adopt TSDF fusion [5] to fuse the depth maps in a volume with the voxel size of $1.5mm$, and then use Marching Cube [27] to extract the mesh. As shown in Table 1, our method outperforms other methods. Specifically, our method outperforms the state-of-the-art method SparseNeuS by about 30% and the traditional MVS method COLMAP [39] by about 10%.

As for qualitative visualization shown in Fig. 3, our method generates finer details than SparseNeuS, e.g., the windows in scan 24 and the stalk of pear in scan 63.

Depth map evaluation on DTU. In this experiment, we compare depth estimation with SparseNeuS [26] and MVSNet [54] by evaluating all 49 views in each scan. For each reference view, we use its best 4 source views for depth rendering. For SparseNeuS [26], we set the image resolution to 800×600 and render the depth in a similar way as our method. For MVSNet [54], we set the image resolution to 1600×1184 . As shown in Table 2, our method achieves better performance in all the metrics than MVSNet and SparseNeuS.

Method	< 1 ↑	< 2 ↑	< 4 ↑	Abs. ↓	Rel. ↓
MVSNet [54]	29.95	52.82	72.33	13.62	1.67
SparseNeuS [26]	38.60	56.28	68.63	21.85	2.68
Ours (VolRecon)	44.22	65.62	80.19	7.87	1.00

Table 2. Depth map evaluation results on DTU [1]. The result of mean absolute error (Abs.) is in millimeters. The results of threshold percentage ($< 1mm$, $< 2mm$, $< 4mm$) and mean absolute relative error (Rel.) are in percentage (%). Best scores are in **bold**.

Full View Reconstruction on DTU. Based on the depth maps for all the views, we further evaluate 3D reconstruction



Figure 4. **Point cloud** comparison of full view reconstruction on the DTU dataset [1]. Compared with SparseNeuS [26], our method (VolRecon) reconstructs better point clouds, e.g. sharper boundary (the steeple in the top left, pear stalk in the top right) and more complete representation, i.e., fewer holes (skull head top in bottom left, foot in the bottom right). Best viewed on a screen when zoomed in.

quality. For a fair comparison, we follow the MVS methods to fuse all 49 depth maps of each scan into one point cloud [13,54]. As shown in Table 3, our method performs significantly better than SparseNeuS and achieves comparable performance as MVSNet. As shown in Fig. 4, compared with SparseNeuS, our method demonstrates better accuracy, e.g. sharper boundary and more complete representation, i.e., fewer holes.

Method	Acc. \downarrow	Comp. \downarrow	Chamfer \downarrow
MVSNet [54]	<u>0.55</u>	0.59	0.57
SparseNeuS [26]	0.75	0.76	0.76
Ours (VolRecon)	0.55	0.66	0.60

Table 3. Point cloud evaluation results on DTU [1]. For Accuracy (Acc.), Completeness (Comp.), and Chamfer distance, lower is better. Best scores are in **bold** and second best are underlined.

Generalization on ETH3D. To validate the generalization ability of our method, we directly test our model, pretrained on the DTU [1], on the ETH3D [40] benchmark. We choose 4 scenes for testing: *door*, *statue*, *relief*, and *relief_2*, which have 6, 11, 31, and 31 images, respectively. Compared with DTU, the scale of the scenes increases about 10 \times . Since the scenes are large-scale and thus it is not suitable to use TSDF fusion [5], we render the depth maps and then fuse all depth maps into a point cloud [54] for each scene. As shown

in Fig. 5, our method reconstructs large-scale scenes with high quality. This demonstrates that our method has good generalization capability.

4.3. Ablation Study

To analyze the effectiveness of different components in our model, we conduct an ablation study. All the experiments are done on DTU [1]. We summarize the results of the first 3 experiments on sparse view ($N = 3$) reconstruction, depth map evaluation, and full view reconstruction in Table 4.

Ray Transformer. By default, a ray transformer enables each point to attend to the features of other points on the ray. Then we remove ray transformer and directly use the unattended features to predict SRDF. As shown in Table 4, the performance drops in all the experiments. Without the ray transformer, the SRDF prediction only uses the local information of each point, which is not enough to accurately find the surface location along the ray.

Global Feature Volume. By default, we build a coarse global feature volume to encode global priors. We compare with not using global feature volume. The performance becomes worse. We conjecture that the local information from projection features is not enough to accurately locate the surface along a ray. The global feature volume provides global shape priors that are helpful for geometry estimation.

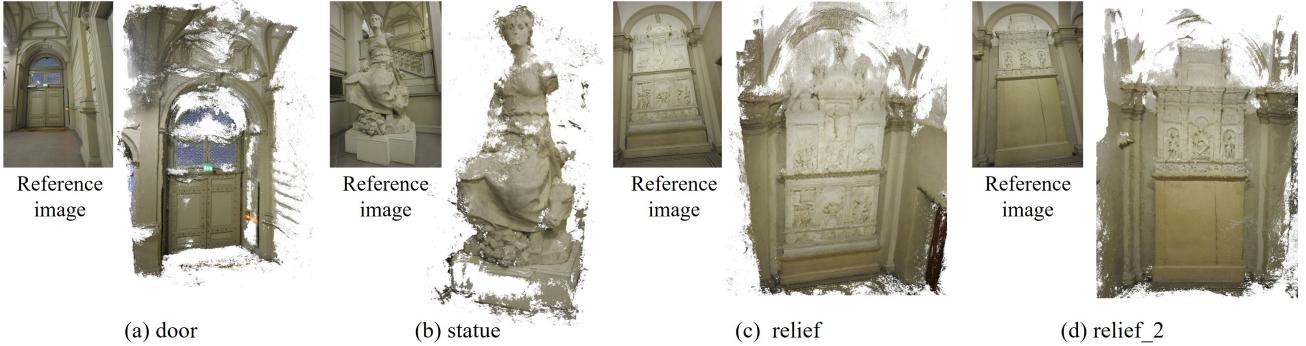


Figure 5. Illustration on the **generalization ability** of VolRecon. Our model trained on DTU [1] generalizes well the to large-scale strong viewpoint variation benchmark ETH3D [40] without finetuning. Best viewed on a screen when zoomed in.

Method	Sparse View Recon.	Depth Map Eval.					Full View Recon.
	Chamfer \downarrow	< 1 \uparrow	< 2 \uparrow	< 4 \uparrow	Abs. \downarrow	Rel. \downarrow	Chamfer \downarrow
w/o Ray Trans.	1.79	39.20	60.73	77.38	8.80	1.12	0.66
w/o \mathbf{F}_v	1.83	23.29	40.67	59.64	14.90	1.92	0.78
w/o $\mathcal{L}_{\text{depth}}$	2.04	12.84	22.55	34.91	35.00	4.41	1.24
Full	1.38	44.22	65.62	80.19	7.87	1.00	0.60

Table 4. Ablation study of ray transformer, global feature volume, and depth loss on DTU [1] dataset. Best scores are in **bold**.

Depth Loss. We remove the depth loss $\mathcal{L}_{\text{depth}}$ during training and observe that the reconstruction quality drops. However, in sparse view reconstruction, our method still performs comparably to the SparseNeuS [26] and better than MVS-NeRF [4] and IBRNet [50], as shown in Table 1. Many works find that only using pixel color loss $\mathcal{L}_{\text{color}}$ produces bad geometry in novel view synthesis [51], especially in areas with little texture or repetitive patterns. Therefore, many implicit reconstruction methods use careful geometry initialization [49, 56] and geometric priors such as depth maps [59], normals [48, 59] and sparse point clouds [11] to provide more geometric supervision. SparseNeuS and other methods [6, 11, 26] use patch loss, which is common in unsupervised depth estimation methods [14] to provide more robust self-supervision in geometry than pixel color loss.

Number of Views. We vary the number of views N in sparse view reconstruction and summarize the results in Table 5. The reconstruction quality gradually improves with more images. Multi-view information enlarges the observed areas and helps to alleviate problems such as occlusions.

Number of Views	Chamfer \downarrow
2	1.72
3	1.38
4	1.35
5	1.33

Table 5. Ablation study of number of views on DTU [1] dataset. The Chamfer distance is reported (lower is better). Best score is in **bold**.

5. Limitations & Future Work

First, the rendering efficiency of our method is limited, which is a common problem that exists in other volume rendering based methods [4, 30, 50]. It takes about 30s to render an image and depth map with a resolution of 800×600 . Second, our current model is not suitable for reconstructing very large-scale scenes. The low resolution of our global feature volume decreases the representation performance when the scale of scene increases. One potential solution is to increase the resolution of the global feature volume. However, this will increase the memory consumption. Instead, we believe it will be a promising direction to reconstruct progressively in small local volumes like NeuralRecon [42]. For example, given a rendering viewpoint, we will select several source views [8, 39, 54] to build a local bounding volume that encloses their view-frustums. This usually limits the space to a reasonable size. Then we will construct the global feature volume for this local bounding volume and use our method to estimate the depth.

6. Conclusion

We introduced VolRecon, a novel generalizable implicit reconstruction method with SRDF. We propose a view transformer to aggregate multi-view features and a ray transformer that computes SRDF values of all the points along a ray to find the surface location. Using projection features and volume features together enables our method to combine local prior and global prior, and thus produce reconstructions with fine details and of high quality. Our method shows sig-

nificantly better performance than the state-of-the-art neural implicit reconstruction methods on DTU. Experiments on ETH3D without any fine-tuning demonstrate good generalization ability on large-scale scenes.

References

- [1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjørholm Dahl. Large-scale data for multiple-view stereopsis. *IJCV*, 2016. [2](#), [3](#), [5](#), [6](#), [7](#), [8](#)
- [2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. [2](#)
- [3] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5799–5809, 2021. [2](#)
- [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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14124–14133, 2021. [2](#), [6](#), [8](#)
- [5] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 303–312, 1996. [1](#), [2](#), [3](#), [6](#), [7](#)
- [6] François Darmon, Bénédicte Basclé, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6260–6269, 2022. [8](#)
- [7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. [4](#)
- [8] Arda Duzceker, Silvano Galliani, Christoph Vogel, Pablo Speciale, Mihai Dusmanu, and Marc Pollefeys. Deepvideomvs: Multi-view stereo on video with recurrent spatio-temporal fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15324–15333, 2021. [8](#)
- [9] William Falcon and The PyTorch Lightning team. PyTorch Lightning, 3 2019. [6](#)
- [10] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5501–5510, 2022. [2](#)
- [11] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *arXiv preprint arXiv:2205.15848*, 2022. [1](#), [2](#), [5](#), [8](#)
- [12] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *PAMI*, 2010. [3](#)
- [13] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *ICCV*, 2015. [1](#), [3](#), [7](#)
- [14] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 270–279, 2017. [8](#)
- [15] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020. [1](#), [3](#)
- [16] Christian Häne, Torsten Sattler, and Marc Pollefeys. Obstacle detection for self-driving cars using only monocular cameras and wheel odometry. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5101–5108. IEEE, 2015. [1](#)
- [17] Sagar Imambi, Kolla Bhanu Prakash, and GR Kanagachidambaresan. Pytorch. In *Programming with TensorFlow*, pages 87–104. Springer, 2021. [6](#)
- [18] Shahram Izadi, David Kim, Otmar Hilliges, David Molyneaux, Richard Newcombe, Pushmeet Kohli, Jamie Shotton, Steve Hodges, Dustin Freeman, Andrew Davison, et al. Kinectfusion: real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 559–568, 2011. [2](#)
- [19] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, pages 5156–5165. PMLR, 2020. [4](#)
- [20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. [6](#)
- [21] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *TOG*, 2017. [3](#)
- [22] Ilya Kostrikov, Esther Horbert, and Bastian Leibe. Probabilistic labeling cost for high-accuracy multi-view reconstruction. In *CVPR*, pages 1534–1541, 2014. [2](#), [3](#)
- [23] Kiriacos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *IJCV*, 38(3):199–218, 2000. [3](#)
- [24] Maxime Lhuillier and Long Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *PAMI*, 27(3):418–433, 2005. [3](#)
- [25] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. *CVPR*, 2017. [4](#)
- [26] Xiaoxiao Long, Cheng Lin, Peng Wang, Taku Komura, and Wenping Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. *arXiv preprint arXiv:2206.05737*, 2022. [1](#), [2](#), [5](#), [6](#), [7](#), [8](#)
- [27] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. [6](#)

- [28] 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. [1](#), [2](#)
- [29] Sven Middelberg, Torsten Sattler, Ole Untzelmann, and Leif Kobbelt. Scalable 6-dof localization on mobile devices. In *European conference on computer vision*, pages 268–283. Springer, 2014. [1](#)
- [30] 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. *Communications of the ACM*, 65(1):99–106, 2021. [1](#), [2](#), [6](#), [8](#)
- [31] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022. [2](#)
- [32] Zak Murez, Tarrence van As, James Bartolozzi, Ayan Sinha, Vijay Badrinarayanan, and Andrew Rabinovich. Atlas: End-to-end 3d scene reconstruction from posed images. In *European conference on computer vision*, pages 414–431. Springer, 2020. [2](#), [4](#)
- [33] Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Marc Stamminger. Real-time 3d reconstruction at scale using voxel hashing. *ACM Transactions on Graphics (ToG)*, 32(6):1–11, 2013. [2](#)
- [34] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. [6](#)
- [35] Martin Ralf Oswald, Jan Stühmer, and Daniel Cremers. Generalized connectivity constraints for spatio-temporal 3d reconstruction. In *European Conference on Computer Vision*, pages 32–46. Springer, 2014. [1](#)
- [36] 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. [1](#), [2](#)
- [37] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. [2](#)
- [38] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. [4](#)
- [39] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, pages 501–518. Springer, 2016. [1](#), [3](#), [4](#), [6](#), [8](#)
- [40] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *CVPR*, 2017. [2](#), [3](#), [6](#), [7](#), [8](#)
- [41] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *IJCV*, 35(2):151–173, 1999. [3](#)
- [42] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021. [2](#), [4](#), [8](#)
- [43] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8258, 2022. [1](#)
- [44] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. [4](#)
- [45] Dan Wang, Xinrui Cui, Septimiu Salcudean, and Z Jane Wang. Generalizable neural radiance fields for novel view synthesis with transformer. *arXiv preprint arXiv:2206.05375*, 2022. [4](#)
- [46] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, and Marc Pollefeys. Itermvs: Iterative probability estimation for efficient multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8606–8615, 2022. [3](#)
- [47] 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. [1](#), [3](#), [4](#)
- [48] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. Neuris: Neural reconstruction of indoor scenes using normal priors. *arXiv preprint arXiv:2206.13597*, 2022. [1](#), [2](#), [8](#)
- [49] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [8](#)
- [50] Qianqian Wang, Zhicheng Wang, Kyle Genova, Pratul P Srinivasan, Howard Zhou, Jonathan T Barron, Ricardo Martin-Brualla, Noah Snavely, and Thomas Funkhouser. Ibrnet: Learning multi-view image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4690–4699, 2021. [2](#), [4](#), [6](#), [8](#)
- [51] Yi Wei, Shaohui Liu, Yongming Rao, Wang Zhao, Jiwen Lu, and Jie Zhou. Nerfingmvs: Guided optimization of neural radiance fields for indoor multi-view stereo. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5610–5619, 2021. [1](#), [8](#)
- [52] Stephan Weiss, Davide Scaramuzza, and Roland Siegwart. Monocular-slam-based navigation for autonomous micro helicopters in gps-denied environments. *Journal of Field Robotics*, 28(6):854–874, 2011. [1](#)
- [53] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *CVPR*, 2019. [3](#)

- [54] 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, 6, 7, 8
- [55] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent MVSNet for high-resolution multi-view stereo depth inference. In *CVPR*, 2019. 1, 3
- [56] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 1, 2, 3, 4, 5, 6, 8
- [57] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 2, 3, 6
- [58] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021. 2, 6
- [59] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *arXiv preprint arXiv:2206.00665*, 2022. 1, 2, 8
- [60] Jingyang Zhang, Yao Yao, Shiwei Li, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Critical regularizations for neural surface reconstruction in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6270–6279, 2022. 1, 2
- [61] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network, 2020. 3
- [62] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 1
- [63] Pierre Zins, Yuanlu Xu, Edmond Boyer, Stefanie Wuhrer, and Tony Tung. Multi-view reconstruction using signed ray distance functions (srdf). *arXiv preprint arXiv:2209.00082*, 2022. 2, 3