SparseNeuS: Fast Generalizable Neural Surface Reconstruction from Sparse Views

Xiaoxiao Long[1] Cheng Lin[2] Peng Wang[1]
Taku Komura[1] Wenping Wang[3]

[1] The University of Hong Kong
[2] Tencent Games
[3] Texas A&M University

Abstract. We introduce *SparseNeuS*, a novel neural rendering based method for the task of surface reconstruction from multi-view images. This task becomes more difficult when only sparse images are provided as input, a scenario where existing neural reconstruction approaches usually produce incomplete or distorted results. Moreover, their inability of generalizing to unseen new scenes impedes their application in practice. Contrarily, *SparseNeuS* can generalize to new scenes and work well with sparse images (as few as 2 or 3). *SparseNeuS* adopts signed distance function (SDF) as the surface representation, and learns generalizable priors from image features by introducing *geometry encoding* volumes for generic surface prediction. Moreover, several strategies are introduced to effectively leverage sparse views for high-quality reconstruction, including 1) a multi-level geometry reasoning framework to recover the surfaces in a coarse-to-fine manner; 2) a multi-scale color blending scheme for more reliable color prediction; 3) a consistency-aware fine-tuning scheme to control the inconsistent regions caused by occlusion and noise. Extensive experiments demonstrate that our approach not only outperforms the state-of-the-art methods, but also exhibits good efficiency, generalizability, and flexibility[1].

Keywords: reconstruction, volume rendering, sparse views

1 Introduction

Reconstructing 3D geometry from multi-view images is a fundamental problem in computer vision and has been extensively researched for decades. Conventional methods for multi-view stereo [2,8,36,18,37,7,19] reconstruct 3D geometry from input images by finding corresponding matches across the input images. However, when only a sparse set of images are available as input, image noises, weak textures and reflections make it difficult for these methods to build dense and complete matches.

[1] Visit our project page: https://www.xxlong.site/SparseNeuS

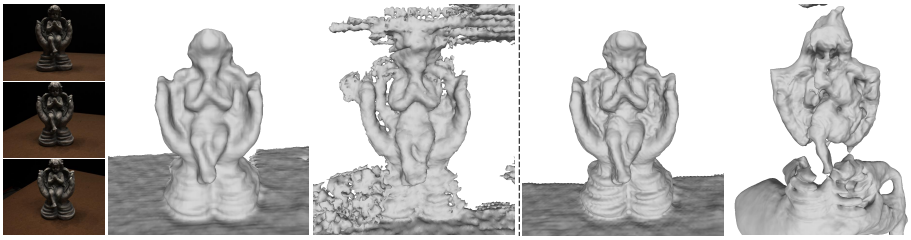

Fig. 1: Our method can generalize across diverse scenes, and reconstruct neural surfaces from only three input images (a) via fast network inference (b). The reconstruction quality of the fast inference step is more accurate and faithful than the result of MVSNerf [3] (c). Our inference result can be further improved by a per-scene fine-tuning process. Compared to NeuS [44] (e), our per-scene optimization result (d) not only achieves noticeably better reconstruction quality, but also takes much less time to converge (12 minutes v.s. 15 hours).

With the recent advances in neural implicit representations, neural surface reconstruction methods [44,50,49,32] leverage neural rendering to jointly optimize the implicit geometry and the radiance field by minimizing the difference of rendered views and ground truth views. Although the methods can produce plausible geometry and photorealistic novel views, they suffer from two major limitations. First, existing methods heavily depend on a large number of input views, i.e. dense views, that are often not available in practice. Second, they require time-consuming per-scene optimization for reconstruction, thus incapable of generalizing to new scenes. The limitations need to be resolved for making such reconstruction methods relevant and useful for practical application.

We propose *SparseNeuS*, a novel multi-view surface reconstruction method with two distinct advantages: 1) it generalizes well to new scenes; 2) it needs only a sparse set of images (as few as 2 or 3 images) for successful reconstruction. *SparseNeuS* achieves these goals by learning generalizable priors from image features and hierarchically leverages the information encoded in the sparse input.

To learn generalizable priors, following MVSNerf [3], we construct a *geometry encoding* volume which aggregates the 2D image features from multi-view input, and use these informative latent features to infer 3D geometry. Consequently, our surface prediction network takes a hybrid representation as input, i.e., xyz coordinates and the corresponding features from the geometry encoding volume, to predict the network-encoded signed distance function (SDF) for the reconstructed surface.

The most crucial part of our pipeline is in how to effectively incorporate the limited information from sparse input images to obtain high-quality surfaces through neural rendering. To this end, we introduce several strategies to tackle this challenge. The first is a *multi-level geometry reasoning scheme* to progressively construct the surface from coarse to fine. We use a cascaded volume encoding structure, i.e., a coarse volume that encodes relatively global features

to obtain the high-level geometry, and a fine volume guided by the coarse level to refine the geometry. A per-scene fine-tuning process is further incorporated into this scheme, which is conditioned on the inferred geometry to construct subtle details to generate even finer-grained surfaces. This multi-level scheme divides the task of high-quality reconstruction into several steps. Each step is based upon the geometry from the preceding step and focuses on constructing a finer level of details. Besides, due to the hierarchical nature of the scheme, the reconstruction efficiency is significantly boosted, because numerous samples far from the coarse surface can be discarded, so as not to burden the computation in the fine-level geometry reasoning.

The second important strategy that we propose is a *multi-scale color blending scheme* for novel view synthesis. Given the limited information in the sparse images, the network would struggle to directly regress accurate colors for rendering novel views. Thus, we mitigate this issue by predicting the linear blending weights of the input image pixels to derive colors. Specifically, we adopt both pixel-based and patch-based blending to jointly evaluate local and contextual radiance consistency. This multi-scale blending scheme yields more reliable color predictions when the input is sparse.

Another challenge in multi-view 3D reconstruction is that 3D surface points often do not have consistent projections across different views, due to occlusion or image noises. With only a small number of input views, the dependence of geometry reasoning on each image further increases, which aggravates the problem and results in distorted geometry. To tackle this challenge, we propose a *consistency-aware fine-tuning scheme* in the fine-tuning stage. This scheme automatically detects regions that lack consistent projections, and excludes these regions in the optimization. This strategy proves effective in making the fine-tuned surface less susceptible to occlusion and noises, thus more accurate and cleaner, contributing to a high-quality reconstruction.

We evaluated our method on the DTU [11] and BlendedMVS [48] datasets, and show that our method outperforms the state-of-the-art unsupervised neural implicit surface reconstruction methods both quantitatively and qualitatively.

In summary, our main contributions are:

- We propose a new surface reconstruction method based on neural rendering. Our method learns generalizable priors across scenes and thus can generalize to new scenes for 3D reconstruction with high-quality geometry.
- Our method is capable of high-quality reconstruction from a sparse set of images, as few as 2 or 3 images. This is achieved by effectively inferring 3D surfaces from sparse input images using three novel strategies: a) multi-level geometry reasoning; b) multi-scale color blending; and c) consistency-aware fine-tuning.
- Our method outperforms the state-of-the-arts in both reconstruction quality and computational efficiency.

2 Related Work

2.1 Multi-view stereo (MVS)

Classical MVS methods utilize various 3D representations for reconstruction such as: voxel grids based [12,13,15,18,37,40], 3D point clouds based [7,19], and depth maps based [2,8,36,42,46,47,10,27,26,25]. Compared with voxel grids and 3D point clouds, depth maps are much more flexible and appropriate for parallel computation, so depth map based methods are most common, like the well-known method COLMAP [36]. Depth map based methods first estimate the depth map of each image, and then utilize filtering operations to fuse the depth maps together into a global point cloud, which can be further processed using a meshing algorithm like Screened Poisson surface reconstruction [16]. These methods achieve promising results with densely captured images. However, with a limited number of images, these methods become more sensitive to image noises, weak textures and reflections, making it difficult for these methods to produce complete reconstructions.

2.2 Neural surface reconstruction

Recently, neural implicit representations of 3D geometry are successfully applied in shape modeling [1,4,9,28,33,29], novel view synthesis [39,24,30,21,34,38,38,43] and mutli-view 3D reconstruction [14,50,31,17,22,44,49,32,52,6]. For the task of multi-view reconstruction, the 3D geometry is represented by a neural network which outputs either occupancy field or Signed Distance Function (SDF). Some methods utilize surface rendering [31] for multi-view reconstruction, but they always need extra object masks [50,31] or depth priors [52], which is inefficient for practical applications. To avoid extra masks or depth priors, some methods [44,49,32,6] leverage volume rendering for reconstruction. However, they also heavily depend on a large number of images to perform a time-consuming per-scene optimization, thus incapable of generalizing to new scenes.

In terms of generalization, there are some successful attempts [51,45,3,23,5] for neural rendering. These methods take sparse views as input and make use of the radiance information of the images to generate novel views, and can generalize to unseen scenes. Although they can generate plausible synthesized images, the extracted geometries from these methods always suffer from noises, incompleteness and distortion.

3 Method

Given a few (i.e., three) views with known camera parameters, we present a novel method that hierarchically recovers surfaces and generalizes across scenes. As illustrated in Figure 2, our pipeline can be divided into three parts: (1) **Geometry reasoning**. *SparseNeuS* first constructs cascaded *geometry encoding* volumes that encode local geometry surface information, and recover surfaces

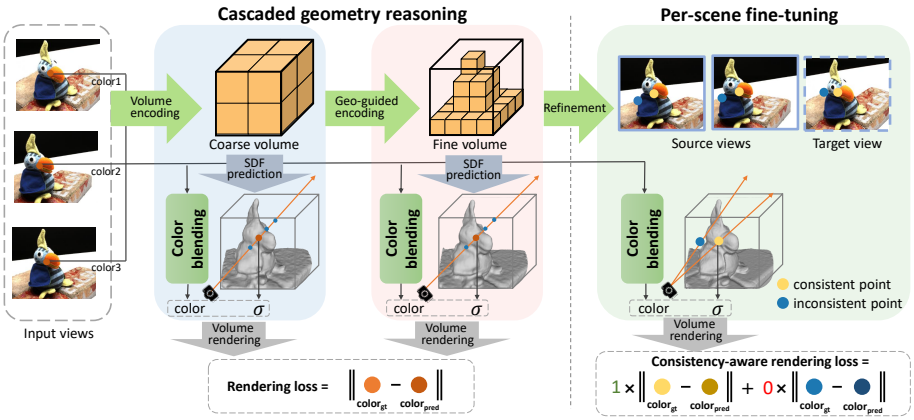

Fig. 2: The overview of *SparseNeuS*. The cascaded geometry reasoning scheme first constructs a coarse volume that encodes relatively global features to obtain the fundamental geometry, and then constructs a fine volume guided by the coarse level to refine the geometry. Finally, a consistency-aware fine-tuning strategy is used to add subtle geometry details, thus yielding high-quality reconstructions with fine-grained surfaces. Specially, a multi-scale color blending module is leveraged for more reliable color prediction.

from the volumes in a coarse-to-fine manner (see Section 3.1). (2) **Appearance prediction**. *SparseNeuS* leverages a multi-scale color blending module to predict colors by aggregating information from input images, and then combines the estimated geometry with predicted colors to render synthesized views using volume rendering (see Section 3.2). (3) **Per-scene fine-tuning**. Finally, a consistency-aware fine-tuning scheme is proposed to further improve the obtained geometry with fine-grained details (see Section 3.3).

3.1 Geometry reasoning

SparseNeuS constructs cascaded *geometry encoding* volumes of two different resolutions for geometry reasoning, which aggregates image features to encode the information of local geometry. Specially, the coarse geometry is first extracted from a *geometry encoding* volume of low resolution, and then it is used to guide the geometry reasoning of the fine level.

Geometry encoding volume. For the scene captured by N input images $\{I_i\}_{i=0}^{N-1}$, we first estimate a bounding box which can cover the region of interests. The bounding box is defined in the camera coordinate system of the centered input image, and then grided into regular voxels. To construct a *geometry encoding* volume M, 2D feature maps $\{F_i\}_{i=0}^{N-1}$ are extracted from the input images $\{I_i\}_{i=0}^{N-1}$ by a 2D feature extraction network. Next, with the camera parameters of one image I_i, we project each vertex v of the bounding box to each feature map F_i and obtain its features $F_i(\pi_i(v))$ by interpolation, where $\pi_i(v)$

denotes the projected pixel location of v on the feature map F_i. For simplicity, we abbreviate $F_i(\pi_i(v))$ as $F_i(v)$.

The *geometry encoding* volume M is constructed using all the projected features $\{F_i(v)\}_{i=0}^{N-1}$ of each vertex. Following prior methods [46,3], we first calculate the variance of all the projected features of a vertex to build a cost volume B, and then apply a sparse 3D CNN Ψ to aggregate the cost volume B to obtain the *geometry encoding* volume M:

$$M = \psi(B), \quad B(v) = \mathrm{Var}\left(\{F_i(v)\}_{i=0}^{N-1}\right), \tag{1}$$

where Var is the variance operation, which computes the variance of all the projected features $\{F_i(v)\}_{i=0}^{N-1}$ of each vertex v.

Surface extraction. Given an arbitrary 3D location q, an MLP network f_θ takes the combination of the 3D coordinate and its corresponding interpolated features of *geometry encoding* volume $M(q)$ as input, to predict the Signed Distance Function (SDF) $s(q)$ for surface representation. Specially, positional encoding PE is applied on its 3D coordinates, and the surface extraction operation is expressed as: $s(q) = f_\theta(\mathrm{PE}(q), M(q))$.

Cascaded volumes scheme. For balancing the computational efficiency and reconstruction accuracy, *SparseNeuS* constructs cascaded *geometry encoding* volumes of two resolutions to perform geometry reasoning in a coarse-to-fine manner. A coarse *geometry encoding* volume is first constructed to infer the fundamental geometry, which presents the global structure of the scene but is relatively less accurate due to limited volume resolution. Guided by the obtained coarse geometry, a fine level *geometry encoding* volume is constructed to further refine the surface details. Numerous vertices far from the coarse surfaces can be discarded in the fine-level volume, which significantly reduces the computational memory burden and improves efficiency.

3.2 Appearance prediction

Given an arbitrary 3D location q on a ray with direction d, we predict its color by aggregating appearance information from the input images. Given limited information in the sparse input images, it is difficult for a network to directly regress color values for rendering novel views. Unlike prior works [51,3], *SparseNeuS* predicts blending weights of the input images to generate new colors. A location q is first projected to the input images to obtain the corresponding colors $\{I_i(q)\}_{i=0}^{N-1}$. Then the colors from different views are blended together as the predicted color of q using the estimated blending weights.

Blending weights. The key of generating the blending weights $\{w_i^q\}_{i=0}^{N-1}$ is to consider the photography consistency of the input images. We project q onto the feature maps $\{F_i\}_{i=0}^{N-1}$ to extract the corresponding features $\{F_i(q)\}_{i=0}^{N-1}$ using bilinear interpolation. Moreover, we calculate the mean and variance of the features $\{F_i(q)\}_{i=0}^{N-1}$ from different views to capture the global photographic consistency information. Each feature $F_i(q)$ is concatenated with the mean and variance together, and then fed into a tiny MLP network to generate a new

feature $F_i'(q)$. Next, we feed the new feature $F_i'(q)$, the viewing direction of the query ray relative to the viewing direction of the i_{th} input image $\Delta d_i = d - d_i$, and the trilinearly interpolated volume encoding feature $M(q)$ into an MLP network f_c to generate blending weight: $w_i^q = f_c(F_i'(q), M(q), \Delta d_i)$. Finally, blending weights $\{w_i^q\}_{i=0}^{N-1}$ are normalized using a Softmax operator.

Pixel-based color blending. With the obtained blending weights, the color c_q of a 3D location q is predicted as the weighted sum of its projected colors $\{I_i(q)\}_{i=0}^{N-1}$ on the input images. To render the color of the query ray, we first predict the color and SDF values of 3D points sampled on the ray. The color and SDF values of the sampled points are aggregated to obtain the final colors of the ray using SDF based volume rendering [44]. Since the color of a query ray corresponds to a pixel of the synthesized image, we name this operation pixel-based blending. Although supervision on the colors rendered by pixel-based blending already induces effective geometry reasoning, the information of a pixel is local and lacks contextual information, thus usually leading to inconsistent surface patches when input is sparse.

Patch-based color blending. Inspired by classical patch matching, we consider enforcing the synthesized colors and ground truth colors to be contextually consistent; that is, not only in pixel level but also in patch level. To render the colors of a patch with size $k \times k$, a naive implementation is to query the colors of k^2 rays using volume rendering, which causes a huge amount of computation. We, therefore, leverage local surface plane assumption and homography transformation to achieve a more efficient implementation.

The key idea is to estimate a local plane of a sampled point to efficiently derive the local patch. Given a sampled point q, we leverage the property of the SDF network $s(q)$ to estimate the normal direction n_q by computing the spatial gradient, i.e., $n_q = \nabla s(q)$. Then, we sample a set of points on the local plane (q, n_q), project the sampled points to each view, and obtain the colors by interpolation on each input image. All the points on the local plane share the same blending weights with q, and thus only one query of the blending weights is needed. Using local plane assumption, we consider the neighboring geometric information of a query 3D position, which encodes contextual information of local patches and enforces better geometric consistency. By adopting patch-based volume rendering, synthesized regions contain more global information than single pixels, thus producing more informative and consistent shape context, especially in the regions with weak texture and changing intensity.

Volume rendering. To rendering the pixel-based color $C(r)$ or patch-based color $P(r)$ of a ray r passing through the scene, we query the pixel-based colors c_i, patch-based colors p_i and sdf values s_i of M samples on the ray, and then utilize [44] to convert sdf values s_i into densities σ_i. Finally, the densities are used to accumulate pixel-based and patch-based colors along the ray:

$$U(r) = \sum_{i=1}^{M} T_i \left(1 - \exp\left(-\sigma_i\right)\right) u_i, \quad \text{where} \quad T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j\right). \tag{2}$$

Here $U(r)$ denotes $C(r)$ or $P(r)$, while u_i denotes the pixel-based color c_i or patch-based color p_i of the i_{th} sample on the ray.

3.3 Per-scene fine-tuning

With the generalizable priors and effective geometry reasoning framework, given sparse images from a new scene, *SparseNeuS* can already recover geometry surfaces via fast network inference. However, due to the limited information in the sparse input views and the high diversity and complexity of different scenes, the geometry obtained by the generic model may contain inaccurate outliers and lack subtle details. Therefore, we propose a novel fine-tuning scheme, which is conditioned on the inferred geometry, to reconstruct subtle details and generate finer-grained surfaces. Thanks to the initialization given by the network inference, the per-scene optimization can fast converge to a high-quality surface.

Fine-tuning networks. In the fine-tuning, we directly optimize the obtained fine-level *geometry encoding* volume and the signed distance function (SDF) network f_θ, while the 2D feature extraction network and 3D sparse CNN networks are discarded. Moreover, the CNN based blending network used in the generic setting is replaced by a tiny MLP network. Although the CNN based network can be also used in per-scene fine-tuning, by experiments, we found that a new tiny MLP can speed up the fine-tuning without loss of performance since the MLP is much smaller than the CNN based network. The MLP network still outputs blending weights $\{w_i^q\}_{i=0}^{N-1}$ of a query 3D position q, but it takes the input as the combination of 3D coordinate q, the surface normal n_q, the ray direction d, the predicted SDF $s(q)$, and the interpolated feature of the *geometry encoding* volume $M(q)$. Specially, positional encoding PE is applied on the 3D position q and the ray direction d. The MLP network f_c' is defined as : $\{w_i^q\}_{i=0}^{N-1} = f_c'(\mathrm{PE}(q), \mathrm{PE}(d), n_q, s(q), M(q))$, where $\{w_i^q\}_{i=0}^{N-1}$ are the predicted blending weights, and N is the number of input images.

Consistency-aware color loss. We observe that in multi-view stereo, 3D surface points often do not have consistent projections across different views, since the projections may be occluded or contaminated by image noises. As a result, the errors of these regions suffer from sub-optima, and the predicted surfaces of the regions are always inaccurate and distorted. To tackle this problem, we propose a consistency-aware color loss to automatically detect the regions lacking consistent projections and exclude these regions in the optimization:

$$\begin{aligned}\mathcal{L}_{color} = &\sum_{r\in\mathbb{R}} O(r)\cdot\mathcal{D}_{pix}\left(C(r), \tilde{C}(r)\right) + \sum_{r\in\mathbb{R}} O(r)\cdot\mathcal{D}_{pat}\left(P(r), \tilde{P}(r)\right) \\ &+ \lambda_0 \sum_{r\in\mathbb{R}} log\left(O(r)\right) + \lambda_1 \sum_{r\in\mathbb{R}} log\left(1 - O(r)\right),\end{aligned} \tag{3}$$

where r is a query ray, $\mathbb{R}$ is the set of all query rays, $O(r)$ is the sum of accumulated weights along the ray r obtained by volume rendering. From Eq. 2, we

can easily derive $O(r) = \sum_{i=1}^{M} T_i (1 - \exp(-\sigma_i))$. $C(r)$ and $\tilde{C}(r)$ are the rendered and ground truth pixel-based colors of the query ray respectively, $P(r)$ and $\tilde{P}(r)$ are the rendered and ground truth patch-based colors of the query ray respectively, and $\mathcal{D}_{pix}$ and $\mathcal{D}_{pat}$ are the loss metrics of the rendered pixel color and rendered patch colors respectively. Empirically, we choose $\mathcal{D}_{pix}$ as L1 loss and $\mathcal{D}_{pat}$ as Normalized Cross Correlation (NCC) loss.

The rationale behind this formulation is, the points with inconsistent projections always have relatively large color errors that cannot be minimized in the optimization. Therefore, if the color errors are difficult to be minimized in optimization, we force the sum of the accumulated weights $O(r)$ to be zero, such that the inconsistent regions will be excluded in the optimization. To control the level of consistency, we introduce two logistic regularization terms: decreasing the ratio λ_0/λ_1 will lead to more regions being kept; otherwise, more regions are excluded and the surfaces are cleaner.

3.4 Training loss

By enforcing the consistency of the synthesized colors and ground truth colors, the training of *SparseNeuS* does not rely on 3D ground-truth shapes. The overall loss function is defined as a weighted sum of the three loss terms:

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \alpha \mathcal{L}_{\text{eik}} + \beta \mathcal{L}_{\text{sparse}}. \tag{4}$$

We note that, in the early stage of generic training, the estimated geometry is relatively inaccurate, and 3D surface points may have large errors, where the errors do not provide clear clues on whether the regions are radiance consistent or not. We utilize consistency-aware color loss in the per-scene fine-tuning, and remove the last two consistence-aware logistic terms of Eq. 3 in the training of the generic model.

An Eikonal term [9] is applied on the sampled points to regularize the SDF values derived from the surface prediction network f_θ:

$$\mathcal{L}_{eik} = \frac{1}{\|\mathbb{Q}\|} \sum_{q \in \mathbb{Q}} \left(\|\nabla f_\theta(q)\|_2 - 1 \right)^2, \tag{5}$$

where q is a sampled 3D point, $\mathbb{Q}$ is the set of all sampled points, $\nabla f_\theta(q)$ is the gradient of network f_θ relatively to sampled point q, and $\|\cdot\|_2$ is l_2 norm. The Eikonal term enforces the network f_θ to have unit l_2 norm gradient, which encourages f_θ to generate smooth surfaces.

Besides, due to the property of accumulated transmittance in volume rendering, the invisible query samples behind the visible surfaces lack supervision, which causes uncontrollable free surfaces behind the visible surfaces. To enable our framework to generate compact geometry surfaces, we adopt a sparseness regularization term to penalize the uncontrollable free surfaces:

$$\mathcal{L}_{sparse} = \frac{1}{\|\mathbb{Q}\|} \sum_{q \in \mathbb{Q}} \exp(-\tau \cdot |s(q)|), \tag{6}$$

Table 1: Evaluation on DTU [11] dataset.

Scan	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
PixelNerf [51]	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	6.28
IBRNet [45]	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	2.32
MVSNerf [3]	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	2.09
Ours	**1.68**	**3.06**	**2.25**	**1.10**	**2.37**	**2.18**	**1.28**	**1.47**	**1.80**	**1.23**	**1.19**	**1.17**	**0.75**	**1.56**	**1.55**	**1.64**
IDR†[50]	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	3.39
VolSdf [49]	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	3.41
UniSurf [32]	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	4.39
Neus [44]	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	4.00
IBRNet-ft [45]	1.67	2.97	2.26	1.56	2.52	2.30	1.50	2.05	2.02	1.73	1.66	1.63	1.17	1.84	1.61	1.90
Colmap [35]	**0.90**	2.89	1.63	1.08	2.18	1.94	1.61	1.30	2.34	1.28	1.10	1.42	0.76	1.17	**1.14**	1.52
Ours-ft	1.29	**2.27**	**1.57**	**0.88**	**1.61**	**1.86**	**1.06**	**1.27**	**1.42**	**1.07**	**0.99**	**0.87**	**0.54**	**1.15**	1.18	**1.27**

† Optimization using extra object masks.

where $|s(q)|$ is the absolute SDF value of sampled point q, τ is a hyperparamter to rescale the SDF value. This term will encourage the SDF values of the points behind the visible surfaces to be far from 0. When extracting 0-level set SDF to generate mesh, this term can avoid uncontrollable free surfaces.

4 Datasets and Implementation

Datasets. We train our framework on the DTU [11] dataset to learn a generalizable network. We use 15 scenes for testing, same as those used in IDR [50], and the remaining non-overlapping 75 scenes for training. All the evaluation results on the testing scenes are generated using three views with a resolution of 600×800, and each scene contains two sets of three images. The foreground masks provided by IDR [50] are used for evaluating the testing scenes. For memory efficiency, we use the center cropped images with resolution of 512×640 for training. We observe that the images of DTU dataset contain large black backgrounds and the regions have considerable image noises, so we utilize a simple threshold based denoising strategy to alleviate the noises of such regions in the training images. Optionally, the black backgrounds with zero RGB values can be used as a simple dataset prior to encourage the geometry predictions of such regions to be empty. We further tested on 7 challenging scenes from the BlendedMVS [48] dataset. For each scene, we select one set of three images with a resolution of 768×576 as input. Note that, in the per-scene fine-tuning stage, we still use the three images for optimization without any new images.

Implementation details. Feature Pyramid Network [20] is used as the image feature extraction network to extract multi-scale features from input images. We implement the sparse 3D CNN networks using a U-Net like architecture, and use torchsparse [41] as the implementation of 3D sparse convolution. The resolutions of the coarse level and fine level *geometry encoding* volumes are $96 \times 96 \times 96$ and $192 \times 192 \times 192$ respectively. The patch size used in patch-based blending is

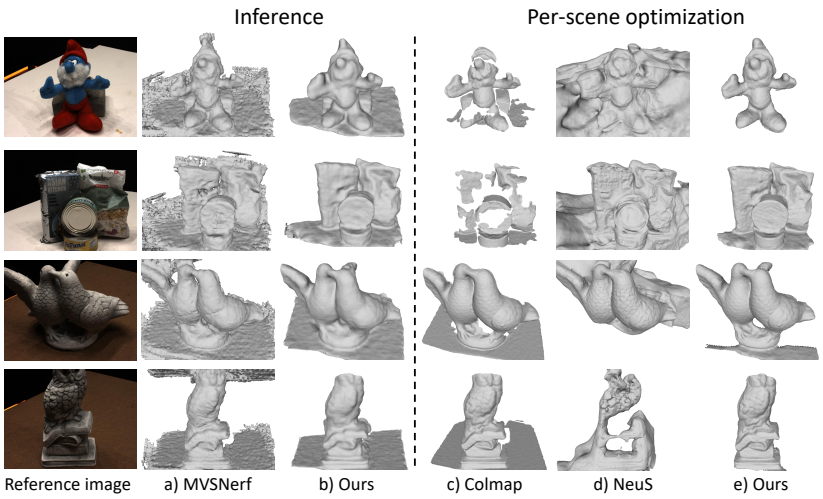

Fig. 3: Visual comparisons on DTU [11] dataset.

5×5. We adopt a two-stage training strategy to train our generic model: in the first stage, the networks of coarse level are first trained for 150k iterations; in the second stage, the networks of fine level are trained for another 150k iterations while the networks of coarse level are fixed. We train our model on two RTX 2080Ti GPUs with a batch size of 512 rays.

5 Experiments

We compare our method with the state-of-the-art approaches from three classes: 1) generic neural rendering methods, including PixelNerf [51], IBRNet [45] and MVSNerf [3], where we use a density threshold to extract meshes from the learned implicit field; 2) per-scene optimization based neural surface reconstruction methods, including IDR [50], NeuS [44], VolSDF [49], and UniSurf [32]; 3) a widely used classic MVS method COLMAP [35], where we reconstruct a mesh from the output point cloud of COLMAP with Screened Poisson Surface Reconstruction [16]. All the methods take three images as input.

5.1 Comparisons

Quantitative comparisons. We perform quantitative comparisons with the SOTA methods on DTU dataset. We measure the Chamfer Distances of the predicted meshes with ground truth point clouds, and report the results in Table 1. The results show that our method outperforms the SOTA methods by a large margin in both generic setting and per-scene optimization setting. Our results obtained by a per-scene fine-tuning with 10k iterations (20 mins) shows remarkable improvements than those of per-scene optimization methods. Note

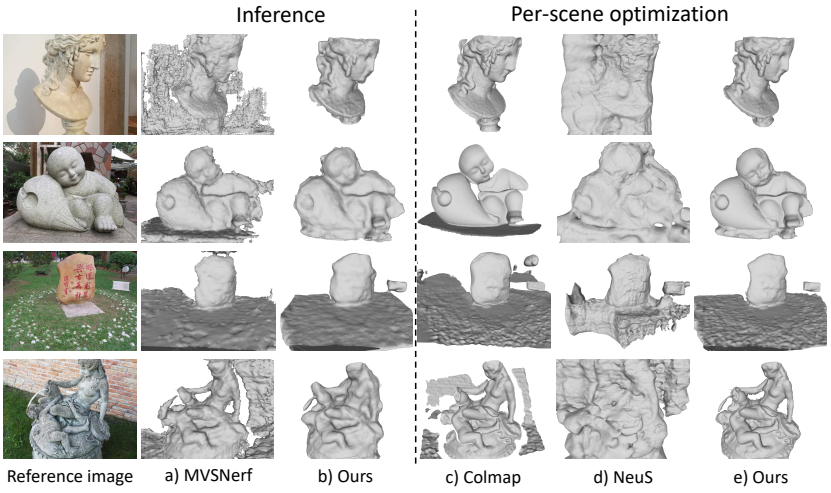

Fig. 4: Visual comparisons on BlendedMVS [48] dataset.

Table 2: Ablation studies on DTU dataset.

Scheme	Setting	Chamfer dist.
Single volume	Generic	1.80
Cas. volumes	Generic	1.56
Single volume	Fine-tuning	1.32
Cas. volumes	Fine-tuning	1.21

(a) The usefulness of Cascaded volumes in both generic and fine-tuning settings.

Pixel	Patch	Consistency	Chamfer dist.
✓	×	×	1.39
✓	✓	×	1.28
✓	✓	✓	1.21

(b) The usefulness of Pixel-based and Patch-based blending, and Consistency-aware scheme in per-scene fine-tuning.

that IDR [50] needs extra object masks for per-scene optimization while others do not need object masks, and we provide the results of IDR for reference.

We further perform a fine-tuning with 10k iterations for IBRNet and MVSNerf with the three input images. With the fine-tuning, the results of IBRNet are improved compared with its generic setting but still worse than our fine-tuned results. MVSNerf fails to perform a fine-tuning with the three input images, therefore, no meaningful geometries are extracted. Furthermore, we observe that MVSNerf usually needs more than 10 images to perform a successful fine-tuning, and thus the failure might be caused by the radiance ambiguity problem.

Qualitative comparisons. We conduct qualitative comparisons with MVSNerf [3], COLMAP [35] and NeuS [44] on DTU [11] and BlendedMVS [48] datasets. As shown in Figure 3, our results obtained via network inference are much smoother and less noisy than those of MVSNerf. The extracted meshes of MVSNerf are noisy since its representation of density implicit field does not have sufficient constraint on level sets of 3D geometry surfaces.

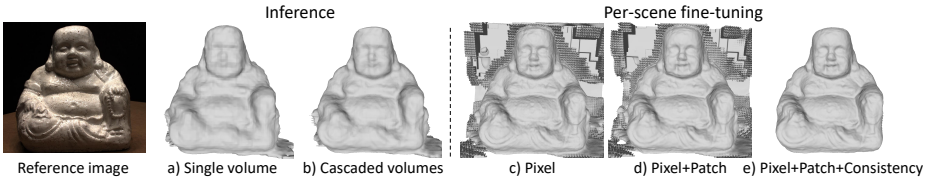

Fig. 5: Qualitative ablation studies. The result obtained by cascaded volumes presents more fine-grained details than that of a single volume. The consistency-aware scheme can automatically detect the regions lacking radiance consistency and exclude them in the fine-tuning, thus yielding cleaner result (e) than the results without consistency-aware scheme (c,d).

After a short-time per-scene fine-tuning, our results are largely improved with fine-grained details and become more accurate and cleaner. Compared with the results of COLMAP, our results are more complete, especially for the objects with weak textures. With only three input images, NeuS suffers from the radiance ambiguity problem and its geometry surfaces are distorted and incomplete.

To validate the generalizability and robustness of our method, we further perform cross dataset evaluation on BlendedMVS dataset. As shown in Figure 4, although our method is not trained on BlendedMVS, our generic model shows strong generalizability and produces cleaner and more complete results than those of MVSNerf. Take the fourth scene in Figure 4 as an example, our method successfully recovers subtle details like the hose, while COLMAP misses the fine-grained geometry. For the scenes with weak textures, NeuS can only produces rough shapes and struggles to recover the details of geometry.

5.2 Ablations and analysis

Ablation studies. We conduct ablation studies (Table 2 and Figure 5) to investigate the individual contribution of the important designs of our method. The ablation studies are evaluated on one set of three images of the 15 testing scenes. The first key module is a *multi-level geometry reasoning scheme* for progressively constructing the surface from coarse to fine. Specially, a cascaded volume scheme is proposed, a coarse volume to generate coarse but high-level geometry, a fine volume guided by the coarse level to refine the geometry. As shown in (a) of Table 2, the cascaded volumes scheme considerably boosts the performance of our method than single volume scheme. In Figure 5, we can see the geometry obtained by cascaded volumes contains more detailed geometry than that of a single volume.

The second important design is a multi-scale color blending strategy, which can enforce the local and contextual radiance consistency of rendered colors and ground truth colors. As shown in (b) of Table 2, the combination of pixel-based and patch-based blending is better than solely using the pixel-based blending. Another important strategy is a consistency-aware scheme that automatically

detects the regions lacking photographic consistency and excludes these regions in fine-tuning. As shown in (b) of Table 2 and Figure 5, result using consistency-aware scheme is noticeably better than those that do not, which is cleaner and gets rid of distorted geometries.

Per-scene optimization with or without priors. Owing to the good initialization provided by the learned priors, the per-scene optimization of our method converges much faster and avoids sub-optimal caused by the radiance ambiguity problem. To validate the effectiveness of the learned priors, we directly perform an optimization without using the learned priors. As shown in Figure 6, the Chamfer Distance of the result with priors is 1.65 while that without prior-based initialization is 1.98. Obviously, the result with learned priors is more complete and smooth, which shows a stark contrast to the direct optimization.

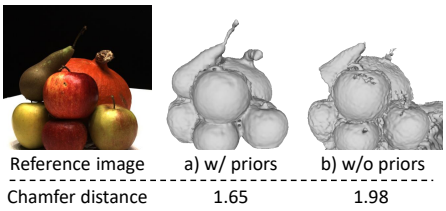

Fig. 6: Per-scene optimization with priors or without priors.

6 Conclusions

We propose *SparseNeuS*, a novel neural rendering based surface reconstruction method to recover surfaces from multi-view images. Our method generalizes to new scenes and produces high-quality reconstructions with sparse images, which prior works [44,49,32] struggle with. To make our method generalize to new scenes, we therefore introduce *geometry encoding* volumes to encode geometry information for generic geometry reasoning. Moreover, a series of strategies are proposed to handle the difficult sparse views setting. First, we propose a multi-level geometry reasoning framework to recover the surfaces in a coarse-to-fine manner. Second, we adopt a multi-scale color blending scheme, which jointly evaluates local and contextual radiance consistency for more reliable color prediction. Third, a consistency-aware fine-tuning scheme is used to control the inconsistent regions caused by occlusion and image noises, yielding accurate and clean reconstruction. Experiments show our method achieve better performance than the state-of-the-arts in both reconstruction quality and computational efficiency. Due to signed distance field adopted, our method can only produce closed-surfaces reconstructions. Possible future directions include utilizing other representations like unsigned distance field to reconstruct open-surfaces objects.

Acknowlegements

We thank the valuable feedbacks of reviewers. Xiaoxiao Long is supported by Hong Kong PhD Fellowship Scheme.

References

1. Atzmon, M., Lipman, Y.: Sal: Sign agnostic learning of shapes from raw data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2565–2574 (2020)
2. Campbell, N.D., Vogiatzis, G., Hernández, C., Cipolla, R.: Using multiple hypotheses to improve depth-maps for multi-view stereo. In: European Conference on Computer Vision. pp. 766–779. Springer (2008)
3. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14124–14133 (2021)
4. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5939–5948 (2019)
5. Chibane, J., Bansal, A., Lazova, V., Pons-Moll, G.: Stereo radiance fields (srf): Learning view synthesis for sparse views of novel scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7911–7920 (2021)
6. Darmon, F., Bascle, B., Devaux, J.C., Monasse, P., Aubry, M.: Improving neural implicit surfaces geometry with patch warping. arXiv preprint arXiv:2112.09648 (2021)
7. Furukawa, Y., Ponce, J.: Accurate, dense, and robust multiview stereopsis. IEEE transactions on pattern analysis and machine intelligence **32**(8), 1362–1376 (2009)
8. Galliani, S., Lasinger, K., Schindler, K.: Massively parallel multiview stereopsis by surface normal diffusion. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 873–881 (2015)
9. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. arXiv preprint arXiv:2002.10099 (2020)
10. Gu, X., Fan, Z., Zhu, S., Dai, Z., Tan, F., Tan, P.: 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. pp. 2495–2504 (2020)
11. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 406–413 (2014)
12. Ji, M., Gall, J., Zheng, H., Liu, Y., Fang, L.: Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2307–2315 (2017)
13. Ji, M., Zhang, J., Dai, Q., Fang, L.: Surfacenet+: An end-to-end 3d neural network for very sparse multi-view stereopsis. IEEE Transactions on Pattern Analysis and Machine Intelligence **43**(11), 4078–4093 (2020)
14. Jiang, Y., Ji, D., Han, Z., Zwicker, M.: Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1251–1261 (2020)
15. Kar, A., Häne, C., Malik, J.: Learning a multi-view stereo machine. Advances in neural information processing systems **30** (2017)
16. Kazhdan, M., Hoppe, H.: Screened poisson surface reconstruction. ACM Transactions on Graphics (ToG) **32**(3), 1–13 (2013)

17. Kellnhofer, P., Jebe, L.C., Jones, A., Spicer, R., Pulli, K., Wetzstein, G.: Neural lumigraph rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4287–4297 (2021)
18. Kutulakos, K.N., Seitz, S.M.: A theory of shape by space carving. International journal of computer vision **38**(3), 199–218 (2000)
19. Lhuillier, M., Quan, L.: A quasi-dense approach to surface reconstruction from uncalibrated images. IEEE transactions on pattern analysis and machine intelligence **27**(3), 418–433 (2005)
20. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
21. Liu, L., Gu, J., Zaw Lin, K., Chua, T.S., Theobalt, C.: Neural sparse voxel fields. Advances in Neural Information Processing Systems **33**, 15651–15663 (2020)
22. Liu, S., Zhang, Y., Peng, S., Shi, B., Pollefeys, M., Cui, Z.: Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2019–2028 (2020)
23. Liu, Y., Peng, S., Liu, L., Wang, Q., Wang, P., Theobalt, C., Zhou, X., Wang, W.: Neural rays for occlusion-aware image-based rendering. arXiv preprint arXiv:2107.13421 (2021)
24. Lombardi, S., Simon, T., Saragih, J., Schwartz, G., Lehrmann, A., Sheikh, Y.: Neural volumes: Learning dynamic renderable volumes from images. arXiv preprint arXiv:1906.07751 (2019)
25. Long, X., Lin, C., Liu, L., Li, W., Theobalt, C., Yang, R., Wang, W.: Adaptive surface normal constraint for depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12849–12858 (2021)
26. Long, X., Liu, L., Li, W., Theobalt, C., Wang, W.: Multi-view depth estimation using epipolar spatio-temporal networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8258–8267 (2021)
27. Long, X., Liu, L., Theobalt, C., Wang, W.: Occlusion-aware depth estimation with adaptive normal constraints. In: European Conference on Computer Vision. pp. 640–657. Springer (2020)
28. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy networks: Learning 3d reconstruction in function space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4460–4470 (2019)
29. Michalkiewicz, M., Pontes, J.K., Jack, D., Baktashmotlagh, M., Eriksson, A.: Implicit surface representations as layers in neural networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4743–4752 (2019)
30. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European conference on computer vision. pp. 405–421. Springer (2020)
31. Niemeyer, M., Mescheder, L., Oechsle, M., Geiger, A.: Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3504–3515 (2020)
32. Oechsle, M., Peng, S., Geiger, A.: Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5589–5599 (2021)

33. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: Deepsdf: Learning continuous signed distance functions for shape representation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 165–174 (2019)
34. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2304–2314 (2019)
35. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)
36. Schönberger, J.L., Zheng, E., Frahm, J.M., Pollefeys, M.: Pixelwise view selection for unstructured multi-view stereo. In: European Conference on Computer Vision. pp. 501–518. Springer (2016)
37. Seitz, S.M., Dyer, C.R.: Photorealistic scene reconstruction by voxel coloring. International Journal of Computer Vision **35**(2), 151–173 (1999)
38. Sitzmann, V., Thies, J., Heide, F., Nießner, M., Wetzstein, G., Zollhofer, M.: Deepvoxels: Learning persistent 3d feature embeddings. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2437–2446 (2019)
39. Sitzmann, V., Zollhöfer, M., Wetzstein, G.: Scene representation networks: Continuous 3d-structure-aware neural scene representations. Advances in Neural Information Processing Systems **32** (2019)
40. Sun, J., Xie, Y., Chen, L., Zhou, X., Bao, H.: Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15598–15607 (2021)
41. Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., Han, S.: Searching efficient 3d architectures with sparse point-voxel convolution. In: European conference on computer vision. pp. 685–702. Springer (2020)
42. Tola, E., Strecha, C., Fua, P.: Efficient large-scale multi-view stereo for ultra high-resolution image sets. Machine Vision and Applications **23**(5), 903–920 (2012)
43. Trevithick, A., Yang, B.: Grf: Learning a general radiance field for 3d representation and rendering. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15182–15192 (2021)
44. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. Advances in Neural Information Processing Systems **34** (2021)
45. Wang, Q., Wang, Z., Genova, K., Srinivasan, P.P., Zhou, H., Barron, J.T., Martin-Brualla, R., Snavely, N., Funkhouser, T.: Ibrnet: Learning multi-view image-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4690–4699 (2021)
46. Yao, Y., Luo, Z., Li, S., Fang, T., Quan, L.: Mvsnet: Depth inference for unstructured multi-view stereo. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 767–783 (2018)
47. Yao, Y., Luo, Z., Li, S., Shen, T., Fang, T., Quan, L.: Recurrent mvsnet for high-resolution multi-view stereo depth inference. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5525–5534 (2019)
48. Yao, Y., Luo, Z., Li, S., Zhang, J., Ren, Y., Zhou, L., Fang, T., Quan, L.: Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1790–1799 (2020)

49. Yariv, L., Gu, J., Kasten, Y., Lipman, Y.: Volume rendering of neural implicit surfaces. Advances in Neural Information Processing Systems **34** (2021)
50. Yariv, L., Kasten, Y., Moran, D., Galun, M., Atzmon, M., Ronen, B., Lipman, Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. Advances in Neural Information Processing Systems **33**, 2492–2502 (2020)
51. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4578–4587 (2021)
52. Zhang, J., Yao, Y., Quan, L.: Learning signed distance field for multi-view surface reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 6525–6534 (2021)