arXiv:2406.04301v5 [cs.CV] 4 Aug 2026

Neural Surface Reconstruction from Sparse Views Using Epipolar Geometry

Xinhai Chang
Yuanpei College, Peking University
changxinhai@stu.pku.edu.cn

Kaichen Zhou ✉
MIT Media Lab and EECS, MIT
zhouk777@mit.edu

Abstract

Reconstructing accurate surfaces from sparse multi-view images remains challenging due to severe geometric ambiguity and occlusions. Existing generalizable neural surface reconstruction methods primarily rely on cost volumes that summarize multi-view features using simple statistics (e.g., mean and variance), which discard critical view-dependent geometric structure and often lead to over-smoothed reconstructions. We propose EpiS, a generalizable neural surface reconstruction framework that explicitly leverages epipolar geometry for sparse-view inputs. Instead of directly regressing geometry from cost-volume statistics, EpiS uses coarse cost-volume features to guide the aggregation of fine-grained epipolar features sampled along corresponding epipolar lines across source views. An epipolar transformer fuses multi-view information, followed by ray-wise aggregation to produce SDF-aware features for surface estimation. To further mitigate information loss under sparse views, we introduce a geometry regularization strategy that leverages a pretrained monocular depth model through scale-invariant global and local constraints. Extensive experiments on DTU and BlendedMVS demonstrate that EpiS significantly outperforms state-of-the-art generalizable surface reconstruction methods under sparse-view settings, while maintaining strong generalization without per-scene optimization.

1. Introduction

Surface reconstruction from multi-view images is a fundamental problem in computer vision [5, 8, 48], robotics [56], and virtual reality [23, 55]. Traditional multi-view stereo pipelines decompose the problem into depth estimation, depth fusion, and surface meshing, which often leads to cumulative errors and brittle performance under challenging conditions. Recent advances in neural implicit representations have significantly reshaped this landscape, enabling end-to-end surface reconstruction and novel view synthesis. Methods such as NeuS [43] and NeuS2 [46], inspired by NeRF [24], represent geometry using Signed Distance Functions (SDFs) and volume rendering, achieving high-quality reconstructions when dense views or strong supervision are available.

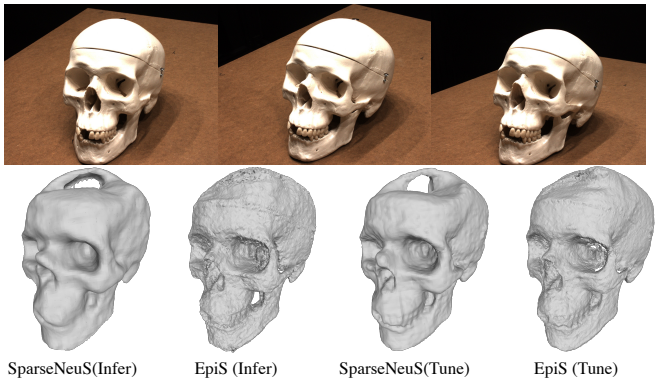


Figure 1. **Reconstruction results on the DTU dataset.** Our approach has remarkable generalization capabilities across various scenes, successfully reconstructing neural surfaces using only three source images through fast network inference. Notably, the reconstruction quality of our fast inference process surpasses that of SparseNeuS, offering enhanced accuracy and fidelity. Additionally, our results can be further refined through per-scene adjustments. (All meshes are visualized with the help of MeshLab2022).

However, reconstructing accurate surfaces from *sparse* multi-view inputs remains highly challenging. Limited viewpoints introduce severe geometric ambiguity and occlusions, while the lack of dense supervision restricts the quality of learned geometry. Existing approaches attempt to alleviate this issue by incorporating additional priors, such as sparse Structure-from-Motion (SfM) point clouds [37] or depth maps. Although effective, these methods typically rely on costly per-scene optimization and struggle to generalize across scenes. More recent generalizable methods, such as SparseNeuS [20], avoid per-scene training but suffer from resolution limitations imposed by memory constraints, often producing over-smoothed surfaces. VolRecon [36] improves reconstruction fidelity by supervising with high-resolution ground-truth depth maps, yet this requirement substantially limits its applicability in real-world scenarios.

A key limitation shared by existing generalizable sparse-view methods lies in their reliance on cost volumes that

summarize multi-view features using simple statistics, such as mean and variance. While effective for capturing coarse geometric cues, these statistics discard view-dependent correspondences encoded along epipolar lines, which are critical for resolving fine-grained geometry under sparse views. As a result, important structural details are lost during feature aggregation, leading to blurred or incomplete surface reconstructions.

To address this limitation, we propose **EpiS**, a fast and generalizable neural surface reconstruction framework tailored for sparse-view inputs. Instead of directly regressing geometry from cost-volume statistics, EpiS uses coarse cost-volume features to *guide the aggregation of fine-grained epipolar features* sampled from multiple source views. An epipolar-guided attention mechanism explicitly fuses view-dependent information along corresponding epipolar lines, while a subsequent ray-wise aggregation module integrates features along each target ray to produce SDF-aware representations. This design preserves critical geometric structure while remaining compatible with generalizable training.

To further compensate for missing information under sparse views, we leverage a pretrained monocular depth model as a geometry regularizer. Rather than enforcing absolute depth consistency, we introduce scale-invariant global and local constraints through a triplet-based global regularization and a gradient-based local regularization, improving geometric stability without requiring ground-truth depth supervision.

Our contributions are summarized as follows:

- We propose **EpiS**, a generalizable neural surface reconstruction framework that explicitly incorporates epipolar geometry to preserve fine-grained multi-view structure under sparse-view inputs.
- We introduce an epipolar-guided feature aggregation strategy that uses cost-volume information to steer multi-view epipolar fusion, followed by ray-wise aggregation for SDF prediction.
- We design scale-invariant depth regularization strategies leveraging a pretrained monocular depth model, improving reconstruction accuracy under sparse views without per-scene optimization.
- Extensive experiments demonstrate that EpiS consistently outperforms state-of-the-art generalizable surface reconstruction methods, particularly in sparse-view and cross-scene settings (Fig. 1).

2. Related Works

2.1. Multi-view depth estimation

Reconstructing 3D geometry from multi-view images poses a fundamental challenge in 3D vision, traditionally approached via depth-based or voxel-based methodologies. Multi-view stereo (MVS) methods, such as those by [3, 9, 39], rely on stereo correspondence for reconstructing depth maps. While early MVS methods utilized hand-crafted similarity metrics, recent advancements integrate deep learning for more precise matching. These approaches span volumetric [10, 12, 15], point cloud-based [7, 16], and depth map-based methodologies [1, 8, 38, 41, 48]. Volumetric and point cloud-based techniques directly model objects but face memory limitations [3, 39]. In contrast, depth map-based methods offer flexibility by separating depth map estimation and fusion, yielding commendable performance on diverse benchmarks [9]. However, their intricate processes, including depth filtering and fusion, may introduce cumulative errors. Despite recent strides in neural implicit representation, their performance trails behind state-of-the-art MVS methods [42, 45, 58]. Addressing this gap, our paper introduces EpiS, showcasing superior performance compared to MVSNet [48] and COLMAP [37] under sparse view conditions through neural implicit representation.

2.2. Neural Implicit Surface Estimation

Neural implicit functions have recently gained traction as effective representations of 3D geometry [6, 21, 22, 26, 30, 32] and appearance [18, 19, 25, 28, 33, 40]. These functions are utilized in both surface and volume rendering approaches to realize 3D geometry reconstruction without the need for 3D supervision. While surface rendering [27, 50, 54] methods focus on single surface intersection points, volume rendering [29, 43, 51, 53] considers multiple points along the ray, resulting in more impressive results. However, both approaches require expensive per-scene optimization and struggle to generalize to new scenes. Successful attempts at generalization [4, 11, 44, 52], in novel view synthesis based on differentiable rendering have leveraged sparse views and radiance information. While these approaches yield more complete surfaces compared to traditional methods, they may struggle with complex structures and sharp corners.

In response to these concerns, recent research has aimed to enhance generalizability and accommodate sparse input by integrating traditional MVS techniques with differentiable rendering. SparseNeuS [20], for example, reconstructs surfaces from nearby viewpoints but may lack detail and suffer from error accumulation. Furthermore, its two-stage training process prolongs training times and complexity. In contrast, VolRecon [36] and ReTR [17] rely on detailed depth ground truth for training, limiting their applicability and fine-tuning stability. Despite attempts at fine-tuning, their results remain inconsistent. Another recent approach, GenS [31], requires more views than the conventional sparse view setting to achieve comparable results.

This paper introduces EpiS, a method capable of achieving remarkable 3D reconstruction under conventional

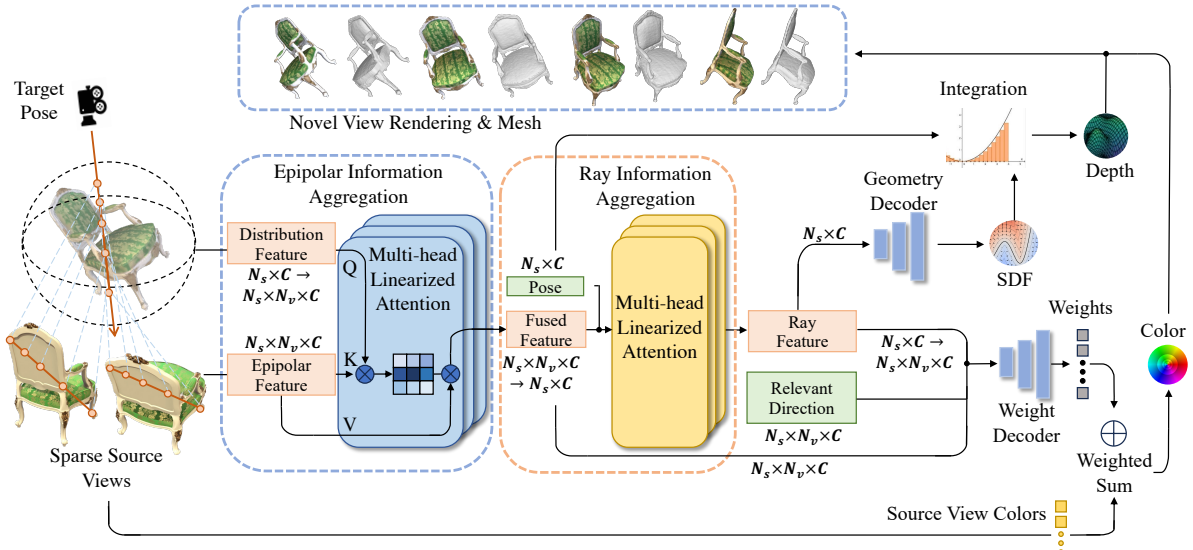


Figure 2. **Illustration of the Pipeline.** Given a ray in the target view, it is projected onto source views to extract the epipolar feature and distribution feature (variance and mean) using a cost volume. Subsequently, the distribution features are utilized as queries, while the epipolar features serve as keys and values for cross-attention transformers, facilitating cross-view epipolar feature fusion. This fused feature set serves as input for subsequent ray transformers, enabling feature aggregation along the target ray. Finally, the resulting feature is used in the geometry MLP and weight decoder to predict corresponding signed distance functions (SDF) and multi-view color weights.

sparse view settings without requiring 3D ground truth supervision. Additionally, we propose efficient regularization techniques to ensure optimal fine-tuning performance using sparse view inputs as shown in Figure. 1.

3. Methodology

Given a set of sparse input images $\{I_i\}_{i=1}^n$ and their corresponding camera poses $\{P_i\}_{i=1}^n$, our goal is to reconstruct accurate surface geometry while generalizing across scenes without per-scene optimization, and to support stable fine-tuning under sparse supervision. **(1)** Most existing sparse-view surface reconstruction methods, such as SparseNeuS [20] and GenS [31], rely on cost volumes that summarize multi-view information using simple statistics (e.g., mean and variance). While effective for coarse geometry, this representation discards fine-grained view correspondences along epipolar lines, often leading to over-smoothed or ambiguous surfaces under sparse views. To overcome this limitation, we combine coarse cost-volume cues with explicit epipolar feature aggregation across source views. An epipolar transformer is introduced to fuse multi-view information along epipolar lines, enabling more precise and geometry-aware surface estimation. **(2)** Fine-tuning in sparse-view settings is further challenged by geometric ambiguity and occlusions. To stabilize optimization, we leverage pretrained monocular depth models as a source of geometric prior. Instead of relying on absolute depth supervision, we introduce two lightweight regularization terms that exploit relative depth consistency, improving depth and surface estimation accuracy during fine-tuning without requiring metric ground-truth depth.

3.1. Preliminaries

The goal of EpiS is to produce appearance and geometry information for a given novel view, following the pipeline established by NeuS [43].

Depth Information Estimation: This is realized by a network that takes a feature considering the 3D point position as input and predicts the surface information. The surface is represented by the zero-level set of the Signed Distance Function (SDF): $S = \{p \in \mathbb{R}^3 | \text{sdf}_\theta(p) = 0\}$. Following NeuS [43], we compute opaque density function $\rho(t)$ from S. To make the depth supervision differential, we could derive the rendered depth as in [56]:

$$\hat{D} = \sum_{j=1}^{N_S} T_j \alpha_j t_j, \quad T_j = \prod_{k=1}^{j-1} (1 - \alpha_k), \tag{1}$$

where $\alpha_j = 1 - \exp(-\int_{t_j}^{t_{j+1}} \rho(t)dt)$; t is the z value from the sampled point to the camera origin and N_S is number of sampled points along each ray.

Appearance Information Estimation: Following IBRNET [44], we make use of the color blending technique. Give a point p_t on the ray of a target frame, this point is projected into source frames to get the color information c_s of multiple pixels. A decoder would be used to predict the weights w_s of each source frame. Based on the volume rendering in NeRF [24], the final color is computed as:

$$\hat{C} = \sum_{j=1}^{N_S} T_j \alpha_j c_j, \quad c_j = \sum_{s=1}^{N_V} w_s * c_s, \tag{2}$$

where c_j is the predicted radiance and N_V is the number of source views.

3.2. Generalizable Framework

3.2.1. Cost Volume Construction

To construct a cost volume for the target camera pose with N_V source frames, we first extract 2D feature maps $\{\boldsymbol{F_i}\}_{i=0}^{N_V}$ from the input images $\{I_i\}_{i=0}^{N_V}$. Then, we construct a corresponding bounding box B for this target pose. To get the feature for each grid b in this bounding box. We project each grid on each source plane and get its feature $\{\boldsymbol{F_i}(P_i(b))\}_{i=0}^{N_V}$, where P_i is the transformation matrix between the target pose and the source pose. Following previous methods [48], we calculate the variance and mean of the projected features for each grid b of the bounding box to create a cost volume.

$$B_1(b) = \mathrm{Var}\left(\boldsymbol{F_i}(P_i(b))_{i=1}^{N_V}\right), \tag{3}$$

$$B_2(b) = \mathrm{Mean}\left(\boldsymbol{F_i}(P_i(b))_{i=1}^{N_V}\right), \tag{4}$$

Then, we employ a sparse 3D convolutional neural network Ψ to process the cost volume and get the coarse geometry aware feature volume $\hat{B}$.

3.2.2. Epipolar & Ray Information Aggregation

Relying solely on variance and means to characterize the distribution of multi-view information overlooks the intricate relationships between multiple views. To address this limitation, we propose an approach that aggregates multi-view information along epipolar lines from multiple source views, while still considering distribution information.

Specifically, for each ray originating from a pixel, EpiS samples N_S points along the ray and projects them onto each source view. This process yields epipolar features from each source view. Subsequently, EpiS utilizes features from the cost volume as the query and the epipolar features from each source view as the key and value within a linearized attention mechanism [13]. This mechanism facilitates the fusion of multi-view epipolar information. Finally, a linear attention mechanism is employed to fuse information along the target ray.

(a) Epipolar Aggregation: This aggregation is realized through constructing a cross-attention transformer, as shown in Figure. 2. Given a ray $r_t = \{p_i\}_{i=1}^{i=N_S}$ from the sampled pixel of the target frame, EpiS extracts the corresponding feature $\boldsymbol{F_B} \in \mathbb{R}^{N_S \times C}$ from cost-volume $\hat{B}$ and the multi-view epipolar feature $\boldsymbol{F_E} \in \mathbb{R}^{N_V \times N_S \times C}$ with the help of $\boldsymbol{F_i}$ and P_i. EpiS initially applies corresponding matrices to process them and compute the query, key, and value as follows:

$$\boldsymbol{Q} = \boldsymbol{F_B W_Q},\ \boldsymbol{K} = \boldsymbol{F_E W_K},\ \boldsymbol{V} = \boldsymbol{F_E W_V}, \tag{5}$$

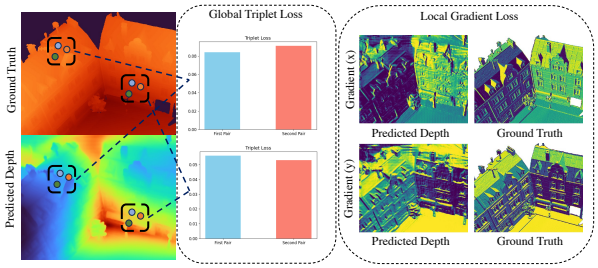


Figure 3. **Visualization of Our Fine-Tuning Strategy Designs.** On the left, we present the predicted and ground truth depth maps. In the middle, we illustrate the triplet loss. On the right, we showcase the derivative gradients along the X and Y axes of the images.

where $\boldsymbol{W_Q}, \boldsymbol{W_K}, \boldsymbol{W_V} \in \mathbb{R}^{C \times C}$. Considering that the query, key, and value have different dimensions, EpiS repeats the query feature along the first dimension and we have $\boldsymbol{Q} \in \mathbb{R}^{N_S \times C} \rightarrow \boldsymbol{Q} \in \mathbb{R}^{N_V \times N_S \times C}$. To ensure Linearized Attention is applied across both multi-view and channel dimensions, we apply dimension permutation to the query, key, and value. In this way, we have $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{N_S \times N_V \times C}$. These matrices are then split into h heads $\boldsymbol{Q} = \{\boldsymbol{Q}^i\}_{i=1}^h$, $\boldsymbol{K} = \{\boldsymbol{K}^i\}_{i=1}^h$, and $\boldsymbol{V} = \{\boldsymbol{V}^i\}_{i=1}^h$, each with $d = C/h$ channels. The cost-volume information is used to guide the learning process through the following Linearized Attention mechanism as:

$$\boldsymbol{X}_r^i = \frac{\phi(\boldsymbol{Q}_r^i)^T(\sum \phi(\boldsymbol{K}_r^i)(\boldsymbol{V}_r^i)^T)}{\phi(\boldsymbol{Q}_r^i)^T \sum \phi(\boldsymbol{K}_r^i)}, \tag{6}$$

where subscripting a matrix with i returns the $i-th$ row along N_V dimension and $\phi(.)$ is the kernal function. In our implementation, we use the same kernel function as in [13], which could be written as $\phi(.) = \mathrm{elu}(x) + 1$. Ultimately, we obtain fused feature $\boldsymbol{X} = \{\boldsymbol{X}^i\}_{i=1}^h \in \mathbb{R}^{N_S \times N_V \times C}$.

(b) Ray Aggregation: Efficient aggregation of epipolar information facilitates the integration of information from epipolar lines across various source views, as shown in Figure. 2. However, taking into account the feature of SDF, which is zero at the surface points and increases or decreases when away from the surface. We also need to consider the information along the ray. To realize this task, we first take the mean of $\boldsymbol{X}$ along the N_V dimension and we have $\boldsymbol{X}' \in \mathbb{R}^{N_S \times C}$.

Furthermore, given that N_S samples are randomly selected along each ray, the distance between each position is uncertain, particularly during the fine-sampling stage, as observed in NeRF [24]. To incorporate this information, we adopt the approach used in previous work, NeRF [24] to embed 3D position information as following $emd(\boldsymbol{x})$. We first concatenate this embedding information with feature information as $\hat{\boldsymbol{X}} = concat(\boldsymbol{X}', emb(\boldsymbol{x}))$. Finally following the previous section, we first process this feature with the help of linear mappings: $\hat{\boldsymbol{Q}} = \hat{\boldsymbol{X}}\hat{\boldsymbol{W}}_{\boldsymbol{Q}}$, $\hat{\boldsymbol{K}} = \hat{\boldsymbol{X}}\hat{\boldsymbol{W}}_{\boldsymbol{K}}$, $\hat{\boldsymbol{V}} = \hat{\boldsymbol{X}}\hat{\boldsymbol{W}}_{\boldsymbol{V}}$. The final ray feature could be writ-

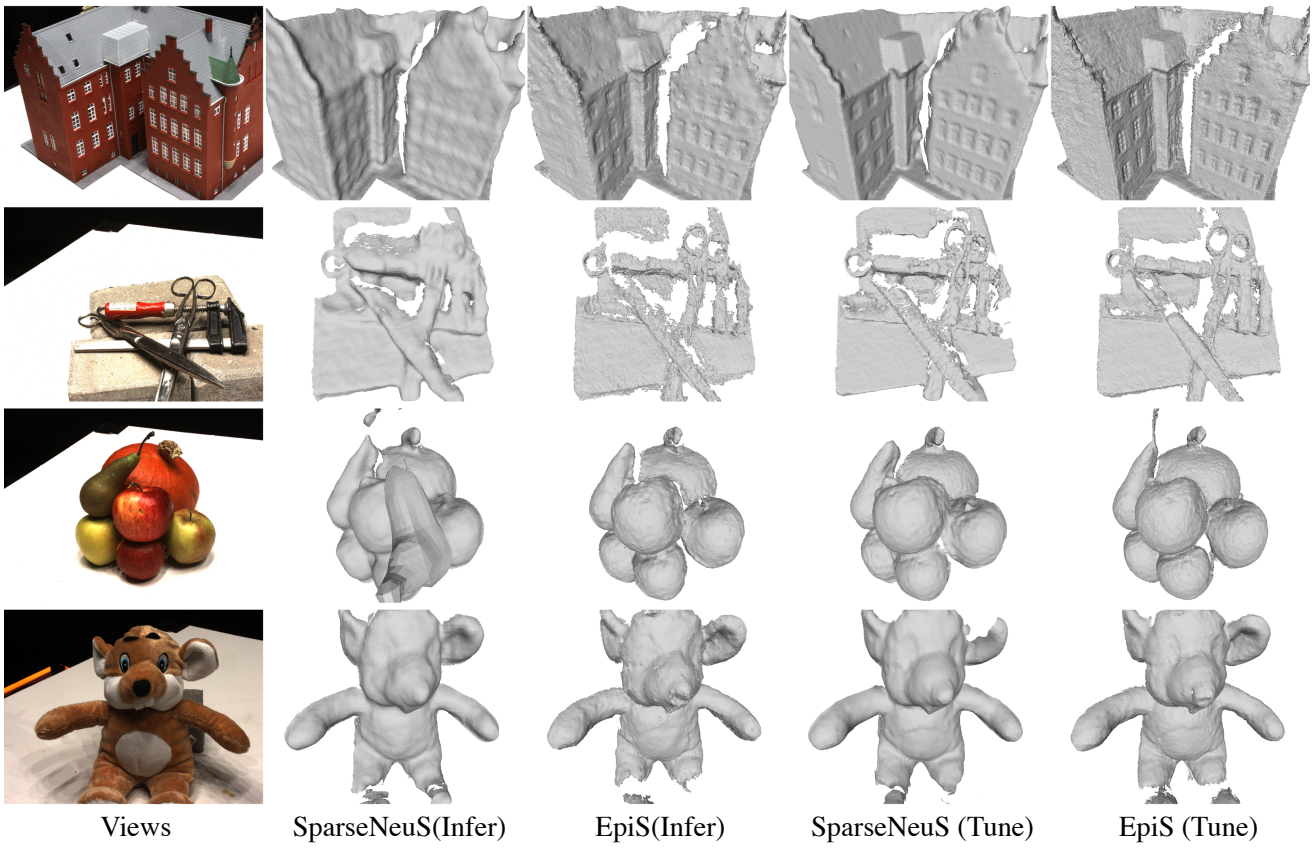


Figure 4. **Visualization results on the DTU dataset.** EpiS produces precise outcomes without requiring fine-tuning. Moreover, fine-tuning further enhances the realism of our results, which is evident in the comparison.

ten as $\hat{\boldsymbol{X}} \in \mathbb{R}^{N_S \times C}$.

3.2.3. Geometry Decoder & Weights Decoder

To compute the final weights and Signed Distance Function (SDF), we employ a decoder, as shown in Figure. 2, similar to the approach used in NeuS [43]. Specifically, the Geometry Decoder f_θ utilizes the final ray feature $\hat{\boldsymbol{X}}$ as input to predict the final SDF. The decision not to include pose information as an input is because $\hat{\boldsymbol{X}}$ inherently encodes pose information. Additionally, the Weight Decoder takes the ray feature $\hat{\boldsymbol{X}}$, fused feature $\boldsymbol{X}$, and relevant directional information as inputs to predict the final color weights.

3.3. Fine-Tuning Strategy for Accuracy

Pre-trained large-scale monocular depth estimation models, trained on extensive datasets, demonstrate efficiency in estimating depth across diverse scenarios [2, 35]. However, these models typically output depth information without scaling. While they perform well in estimating relative depth, absolute depth estimation can be challenging. Often, the generated depth lacks a scale factor and offset compared to the ground truth, as illustrated in DynPoint [57], where $d_{gt} = \alpha\hat{d} + \beta$. This feature complicates the use of generated depths for supervising depth estimation tasks. To address this, we propose both a global triplet loss function and a local gradient loss function.

(a) Global Triplet Loss: Addressing the challenge posed by the scale discrepancy between predicted depth and ground truth depth, directly employing predicted depth for supervision proves difficult. However, by capitalizing on the sampling of NR from the target frame during each iteration for training, exploiting the relative relationship between different rays becomes viable. Specifically, given sampled rays r_s, two rays r_1 and r_2 are randomly selected from NR. Subsequently, the global triplet loss function can be expressed as:

$$\mathcal{L}_{\text{global}} = \Big((\hat{d}_1 - \hat{d}_s) \times (\tilde{d}_2 - \tilde{d}_s) - (\hat{d}_2 - \hat{d}_s) \times (\tilde{d}_1 - \tilde{d}_s) \Big)^2, \tag{7}$$

where $\hat{d}_1, \hat{d}_2, \hat{d}_s$ represent the estimated depth by EpiS, and $\tilde{d}_1, \tilde{d}_2, \tilde{d}_s$ represent the estimated depth by the Pretrained Depth Model. Please note that while this loss function is named "global," it is applied only to local patches of the depth map. This approach is adopted because ensuring the relevance of the relationship between two distant pixels of the depth map cannot be guaranteed. The intuition of this loss function is demonstrated in the left part of Figure. 3.

(b) Local Gradient Loss: The global triplet loss primarily considers the overall structure of the depth map, often overlooking detailed local information. To incorporate local gradients, we utilize the partial derivatives of the depth map along the x and y axes. The partial derivative vectors are defined as $\hat{\boldsymbol{v}} = (\frac{\partial \hat{d}}{\partial x}, \frac{\partial \hat{d}}{\partial y})$ and $\tilde{\boldsymbol{v}} = (\frac{\partial \tilde{d}}{\partial x}, \frac{\partial \tilde{d}}{\partial y})$. The loss function could be written as:

$$\mathcal{L}_{\text{local}} = (1 - \frac{\hat{\boldsymbol{v}} \cdot \tilde{\boldsymbol{v}}}{||\hat{\boldsymbol{v}}|| \cdot ||\tilde{\boldsymbol{v}}||})^2. \tag{8}$$

This loss function is employed to regularize the direction of the partial derivative vector while disregarding their absolute value. The intuition of this loss function is demonstrated in the right part of Figure. 3.

3.4. Loss Function

Given that EpiS relies solely on color information for supervising neural surface reconstruction, akin to prior neural surface reconstruction methods [46], we utilize the following loss functions for training:

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \lambda_1 \mathcal{L}_{\text{eik}} + \lambda_2 \mathcal{L}_{\text{sparse}} + \lambda_3 \mathcal{L}_{\text{global}} + \lambda_4 \mathcal{L}_{\text{local}} \tag{9}$$

The first four loss functions are frequently employed in sparse view neural surface reconstruction tasks. The color loss function [46] is used to compute the distance between the predicted color from Eqn. 2, which could be written as:

$$\mathcal{L}_{color} = \frac{1}{\|\text{Patch}\|} \sum_{pix \in \text{Patch}} |\hat{c}(pix) - c(pix)|, \tag{10}$$

where Patch means the sampled patch used in one iteration and pix represents pixels within this patch.

The second Eikonal loss function [46] is used to regularize predicted SDF value predicted by our network, which could be written as:

$$\mathcal{L}_{eik} = \frac{1}{\|\mathbb{P}\|} \sum_{p \in \mathbb{P}} (\|\nabla \text{sdf}_\theta(p)\|_2 - 1)^2 \tag{11}$$

where $\mathbb{P}$ is the set of all sampled points based on the sampled patch. This loss function forces network $\text{sdf}_\theta(p)$ to have a unit gradient.

To oversee the points beneath the predicted surface and prevent the occurrence of free surfaces, akin to [20], we integrate the sparse loss function term. This term aims to minimize the distance of points beneath the surface to zero as much as possible:

$$\mathcal{L}_{sparse} = \frac{1}{\|\mathbb{P}\|} \sum_{p \in \mathbb{P}} \exp\left(-\tau \cdot |\text{sdf}_\theta(p)|\right), \tag{12}$$

where τ is the hyperparameter.

4. Experiments

4.1. Experimental Settings

4.1.1. Baselines

In our study, we follow the methodology outlined by MVSNet [4] for dense view analysis, utilizing the first 49 images for training. Our method undergoes comparison with state-of-the-art approaches from three classes: generic neural rendering methods, including PixelNerf [52], IBRNet [44], MVSNerf [4] and SparseView [20]; per-scene optimization-based neural surfaces reconstruction methods like IDR [50], NeuS [46], VolSDF [51], and UniSurf [29]; and the classic MVS method COLMAP [37] and MVSNet [48]. All methods use three images as input. It's worth noting that MVS methods, unlike neural implicit reconstruction, do not explicitly model scene parameters and cannot render novel views.

Besides, It's worth noting that while GenS shows promising results, conducting a direct comparison poses challenges. The previous study [20] focused on utilizing three views (N = 3), while GenS [31] employs four views (N = 4), leading to significant differences in results. Moreover, the specific configuration of the four views used in GenS remains undisclosed, hindering clarity on their impact. Additionally, GenS's use of 19 views for their Dense View experiment further complicates comparisons, especially considering the undisclosed rationale behind this choice. Due to these factors, we refrain from providing a direct comparison between EpiS and GenS.

4.1.2. Datasets

We utilize the DTU dataset [1] for training, a multi-view stereo dataset comprising 124 scenes with ground truth point clouds and varying lighting conditions. Testing is conducted on the same 15 scenes as SparseNeuS, with the remaining scenes allocated for training. Depth maps rendered from the mesh serve as ground truth. Our framework, trained on the DTU dataset to ensure network generalization, employs 15 scenes for testing and the remaining 75 for training. Evaluation on testing scenes involves three views at 600 × 800 resolution, with each scene containing two sets of three images. Foreground masks provided by IDR are used for evaluation. To enhance memory efficiency during training, center-cropped images with 512 × 640 resolution are used. A simple threshold-based denoising strategy is applied to mitigate image noise. Additionally, we test our model on 7 challenging scenes from the BlendedMVS dataset [49], using one set of three images per scene with a resolution of 768 × 576. It's worth noting that in the per-scene fine-tuning stage, we optimize using the same three images without introducing new ones. Our experiments on both DTU and BlendedMVS datasets align with previous methods and include reporting Chamfer Distance for DTU

Table 1. Quantitative results of sparse view reconstruction on 15 testing scenes from the DTU dataset. The upper part presents the performance of generalizable models, while the lower part displays the performance of fine-tuning models. If the result of EpiS outperforms all other methods, it will be **highlighted**. If the result of EpiS is the second best among all methods, it will be underlined.

Method	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean↓
PixelNerf [52]	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 [44]	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 [4]	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
SparseNeuS [20]	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
EpiS	**1.11**	**2.71**	**1.85**	1.12	**1.47**	**1.69**	1.05	**1.45**	**1.35**	**0.97**	1.21	1.35	**0.71**	**1.20**	**1.22**	**1.36**
UniSurf [29]	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 [43]	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
VolSDF [51]	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
IBRNet (ft) [44]	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 [37]	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
SparseNeuS (ft) [20]	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
MVSNet [48]	1.05	2.52	1.71	1.04	1.45	1.52	0.88	1.29	1.38	1.05	0.91	0.66	0.61	1.08	1.16	1.22
EpiS (ft)	0.93	**2.13**	**1.32**	**0.87**	**1.01**	**1.56**	**0.84**	**1.21**	**1.10**	**0.84**	**0.79**	0.89	**0.51**	**1.05**	**1.06**	**1.07**

Table 2. Depth evaluation on the DTU dataset. 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 (%). If the result of EpiS outperforms all other methods, it will be **highlighted**. If the result of EpiS is the second best among all methods, it will be underlined.

Method	Supervision	$< 1\uparrow$	$< 2\uparrow$	$< 4\uparrow$	Abs. ↓	Rel. ↓
MVSNet [48]	RGBD	29.95	52.82	72.33	13.62	1.67
VolRecon [36]	RGBD	44.22	65.62	80.19	7.87	1.00
SparseNeuS [20]	RGB	38.60	56.28	68.63	21.85	2.68
EpiS	RGB	43.97	**66.16**	**83.33**	**7.79**	**0.99**

and showcasing visual effects for BlendedMVS.

4.1.3. Implementation

Our model, developed using PyTorch [34] and PyTorch Lightning, operates with an image resolution of 640×512 during training. Training occurs over 16 epochs, utilizing the Adam optimizer [14], on a single 4090 GPU with a learning rate of 10^{-4}. The batch size is set to 2, with 1024 rays sampled per batch. Both training and testing involve a hierarchical sampling strategy, initially sampling N coarse points uniformly on the ray and then employing importance sampling to sample additional N fine points on top of the coarse probability estimation, where $Ncoarse = 64$ and $Nfine = 64$. The fine-level geometry encoding volumes are configured at a resolution of $96 \times 96 \times 96$. The sparse 3D CNN networks, structured akin to a U-Net, utilize a patch size of 5×5 for patch-based blending. During testing, the image resolution is adjusted to 800×600.

4.2. Evaluation Results

4.2.1. Sparse View Reconstruction on DTU

In the DTU dataset [1], we conduct sparse reconstruction using only 3 views, assessing performance through quantitative measures like Chamfer Distances between predicted meshes and ground truth point clouds, as shown in Table. 1 In contrast to MVS methods such as COLMAP [37] and MVSNet [48], our approach exhibits approximately a 11% improvement over COLMAP, albeit slightly trailing behind MVSNet. It's worth noting that MVSNet requires ground truth depth maps and lacks the capability to render novel views. Our method surpasses the state-of-the-art neural implicit reconstruction method SparseNeuS [20], achieving an 18% enhancement in the generalizable experiment and a 16% enhancement in the fine-tuning case. Moreover, our method outperforms VolRecon, which necessitates ground truth depth maps for training. Qualitative visualization in Figure. 4 highlights our method's capacity to generate finer details and sharper boundaries than SparseNeuS.

4.2.2. Depth map evaluation on DTU

In this experiment, we evaluate depth estimation performance using SparseNeuS [20]a nd MVSNet [48] across all views in each scan. For each reference view, we select the top 4 source views based on view selection scores from [48] for depth rendering. The results in Table. 2 indicate that our method outperforms both MVSNet and SparseNeuS across all metrics, and even surpasses VolRecon. This finding aligns with the results reported in Table. 1

4.2.3. Full View Reconstruction on DTU

For a comprehensive comparison, we also conduct a mesh reconstruction experiment using dense views (the first 49 views for all scenarios). As demonstrated in Table. 3, our method consistently outperforms previous per-scene optimization methods and MVS methods without the need for further training.

4.2.4. Generalization on BlendedMVS

To assess the generalization capability of our method, we extend our testing to BlendedMVS. The quantitative results are presented in Figure. 5. It is evident that EpiS exhibits superior performance in reconstructing details but may produce noisy boundaries compared to SparseNeuS.

Table 3. Quantitative results of **dense view** reconstruction on 15 testing scenes of DTU dataset. If the result of EpiS outperforms all other methods, it will be **highlighted**. If the result of EpiS is the second best among all methods, it will be underlined.

Method	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean↓
NeRF [24]	1.90	1.60	1.85	0.58	2.28	1.27	1.47	1.67	2.05	1.07	0.88	2.53	1.06	1.15	0.96	1.49
IDR [50]	1.63	1.87	0.63	0.48	1.04	0.79	0.77	1.33	1.16	0.76	0.67	0.90	0.42	0.51	0.53	0.90
MVSDF [54]	0.83	1.76	0.88	0.44	1.11	0.90	0.75	1.26	1.02	1.35	0.87	0.84	0.34	0.47	0.46	0.88
VolSDF [51]	1.14	1.26	0.81	0.49	1.25	0.70	0.72	1.29	1.18	0.70	0.66	1.08	0.42	0.61	0.55	0.86
NeuS [43]	1.00	1.37	0.93	0.43	1.10	0.65	0.57	1.48	1.09	0.83	0.52	1.20	0.35	0.49	0.54	0.84
Voxurf [47]	0.65	0.74	0.39	0.35	0.96	0.64	0.85	1.58	1.01	0.68	0.60	1.11	0.37	0.45	0.47	0.72
COLMAP [37]	0.45	0.91	0.37	0.37	0.90	1.00	0.54	1.22	1.08	0.64	0.48	0.59	0.32	0.45	0.43	0.65
EpiS	0.51	0.81	0.42	0.44	**0.82**	0.67	**0.54**	1.24	**0.95**	0.70	0.54	0.66	0.38	**0.44**	0.44	**0.63**

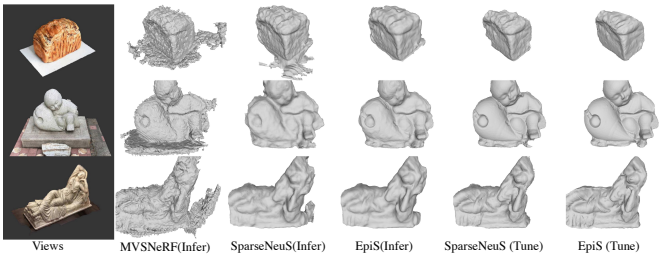


Figure 5. **Reconstruction results on the BlendedMVS dataset.** EpiS yields reasonably accurate estimation even without pretraining on BlendedMVS. Fine-tuning enhances EpiS's performance, leading to further improvements in accuracy.

4.3. Ablation Studies

4.3.1. Epipolar Aggregation.

In the first part of our ablation studies, we analyze the impact of removing the epipolar aggregation component from our framework. This adjustment results in the ray aggregation directly utilizing features from the cost-volume as input. The results in Table. 4 indicate that without epipolar aggregation, the ray aggregation process alone is insufficient to adequately capture the detailed geometry from the cost-volume data.

4.3.2. Global Triplet Loss.

Moving on to the second part, we examine the consequences of eliminating the global triplet loss function, which plays a crucial role in supervising the rendered depth. Without this loss function, our model fails to effectively leverage the relevant depth relationships provided by the pretrained large monocular depth model. As a result, there is a notable decline in performance.

4.3.3. Local Gradient Loss.

In the third part of our ablation studies, we focus on the removal of the local gradient loss function, which also contributes to supervising the rendered depth. Without this loss function, our model experiences a further deterioration in performance, underscoring the critical role it plays in maintaining the accuracy of our results.

Table 4. Ablation studies of epipolar transformer, global depth loss, and local depth loss on the DTU dataset. We report results on both sparise view setting and dense view setting. The best results will be **highlighted**. The second best results will be underlined.

Method	Sparse View Recon.	Depth Map Eval.					Full View Recon.
	Chamfer↓	$<1\uparrow$	$<2\uparrow$	$<4\uparrow$	Abs.↓	Rel. ↓	Chamfer↓
w/o Epipolar Aggregation	1.70	31.04	67.19	82.55	08.03	1.10	0.71
w/o Global Triplet Loss	1.86	20.64	38.86	56.48	15.49	2.11	0.83
w/o Local Gradient Loss	1.90	15.84	33.65	54.72	18.93	3.72	0.98
EpiS	**1.37**	**43.97**	**66.16**	**83.33**	**07.79**	**0.99**	**0.63**

Table 5. Ablation study of the input view number on the DTU dataset. In this experiment, we modify the number of input views to test the performance of EpiS under different settings. The best results will be **highlighted**.The second best results will be underlined.

Method	Supervision	2 Views	3 Views	4 Views	6 Views
VolRecon [36]	RGBD	1.72	1.38	1.35	1.33
EpiS	RGB	**1.66**	**1.36**	**1.27**	**1.25**

4.3.4. Number of Input Views.

In the final segment of our ablation studies, we investigate the performance of EpiSs with varying numbers of input views. All results are reported in Table. 5. Our findings reveal that increasing the number of input views significantly enhances the performance of EpiS. This improvement is attributed to the increased effectiveness in eliminating occlusion, particularly evident in sparse view scenarios. The addition of more views effectively mitigates occlusion challenges, consequently leading to notable enhancements.

5. Conclusion

In this paper, we introduce EpiS, a generalizable method for sparse view neural surface reconstruction. Our approach incorporates an epipolar aggregation module to facilitate multi-view epipolar line fusion, guided by the cost volume. Additionally, we introduce a ray aggregation module to fuse information along the target ray. To regularize the depth learning process, we propose a global triplet loss and a local derivative loss, leveraging insights from a pretrained monocular depth model. Our model significantly outperforms state-of-the-art generalizable methods in neural surface reconstruction on the DTU dataset. Furthermore, the generalizability of our approach is demonstrated through compelling results on the BlendedMVS dataset.

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120:153–168, 2016. 2, 6, 7

[2] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 5

[3] Neill DF Campbell, George Vogiatzis, Carlos Hernández, and Roberto Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I 10*, pages 766–779. Springer, 2008. 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, 7

[5] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8585–8594, 2022. 1

[6] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7154–7164, 2019. 2

[7] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020. 2

[8] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2020. 1, 2

[9] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE international conference on computer vision*, pages 2307–2315, 2017. 2

[10] Mengqi Ji, Jinzhi Zhang, Qionghai Dai, and Lu Fang. 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. 2

[11] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375, 2022. 2

[12] Abhishek Kar, Christian Häne, and Jitendra Malik. Learning a multi-view stereo machine. *Advances in neural information processing systems*, 30, 2017. 2

[13] 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

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[15] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International journal of computer vision*, 38:199–218, 2000. 2

[16] Maxime Lhuillier and Long Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE transactions on pattern analysis and machine intelligence*, 27(3):418–433, 2005. 2

[17] Yixun Liang, Hao He, and Yingcong Chen. Retr: Modeling rendering via transformer for generalizable neural surface reconstruction. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[18] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020. 2

[19] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2019–2028, 2020. 2

[20] Xiaoxiao Long, Cheng Lin, Peng Wang, Taku Komura, and Wenping Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, pages 210–227. Springer, 2022. 1, 2, 3, 6, 7

[21] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 2

[22] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4743–4752, 2019. 2

[23] Sven Middelberg, Torsten Sattler, Ole Untzelmann, and Leif Kobbelt. Scalable 6-dof localization on mobile devices. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*, pages 268–283. Springer, 2014. 1

[24] 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, 3, 4, 8

[25] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 2

[26] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4d reconstruction by

learning particle dynamics. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5379–5389, 2019. 2

[27] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020. 2

[28] Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4531–4540, 2019. 2

[29] 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. 2, 6, 7

[30] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 2

[31] Rui Peng, Xiaodong Gu, Luyang Tang, Shihe Shen, Fanqi Yu, and Ronggang Wang. Gens: Generalizable neural surface reconstruction from multi-view images. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 3, 6

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

[33] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 2

[34] Automatic Differentiation In Pytorch. Pytorch, 2018. 7

[35] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020. 5

[36] Yufan Ren, Tong Zhang, Marc Pollefeys, Sabine Süsstrunk, and Fangjinhua Wang. Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16685–16695, 2023. 1, 2, 7, 8

[37] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1, 2, 6, 7, 8

[38] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 501–518. Springer, 2016. 2

[39] Robust Multiview Stereopsis. Accurate, dense, and robust multiview stereopsis. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, 32(8), 2010. 2

[40] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5459–5469, 2022. 2

[41] Engin Tola, Christoph Strecha, and Pascal Fua. Efficient large-scale multi-view stereo for ultra high-resolution image sets. *Machine Vision and Applications*, 23:903–920, 2012. 2

[42] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5294–5306, 2025. 2

[43] 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, 5, 7, 8

[44] 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, 3, 6, 7

[45] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. DUSt3R: Geometric 3D vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20697–20709, 2024. 2

[46] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3295–3306, 2023. 1, 6

[47] Tong Wu, Jiaqi Wang, Xingang Pan, Xudong Xu, Christian Theobalt, Ziwei Liu, and Dahua Lin. Voxurf: Voxel-based efficient and accurate neural surface reconstruction. In *International Conference on Learning Representations (ICLR)*, 2023. 8

[48] 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, 4, 6, 7

[49] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *Computer Vision and Pattern Recognition (CVPR)*, 2020. 6

[50] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and ap-

pearance. *Advances in Neural Information Processing Systems*, 33, 2020. 2, 6, 8

[51] 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. 2, 6, 7, 8

[52] 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, 7

[53] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems*, 35:25018–25032, 2022. 2

[54] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6525–6534, 2021. 2, 8

[55] Kaichen Zhou, Changhao Chen, Bing Wang, Muhamad Risqi U Saputra, Niki Trigoni, and Andrew Markham. Vmloc: Variational fusion for learning-based multimodal camera localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6165–6173, 2021. 1

[56] Kaichen Zhou, Lanqing Hong, Changhao Chen, Hang Xu, Chaoqiang Ye, Qingyong Hu, and Zhenguo Li. Devnet: Self-supervised monocular depth learning via density volume construction. In *European Conference on Computer Vision*, pages 125–142. Springer, 2022. 1, 3

[57] Kaichen Zhou, Jia-Xing Zhong, Sangyun Shin, Kai Lu, Yiyuan Yang, Andrew Markham, and Niki Trigoni. Dynpoint: Dynamic neural point for view synthesis. *Advances in Neural Information Processing Systems*, 36, 2024. 5

[58] Kaichen Zhou, Yuhan Wang, Grace Chen, Gaspard Beaudouin, Fangneng Zhan, Paul Liang, and Mengyu Wang. Page-4d: Disentangled pose and geometry estimation for vggt-4d perception. In *International Conference on Learning Representations*, pages 36401–36414, 2026. 2