FatesGS: Fast and Accurate Sparse-View Surface Reconstruction Using Gaussian Splatting with Depth-Feature Consistency

Han Huang[1,2*], **Yulun Wu**[1,2*], **Chao Deng**[1,2], **Ge Gao**[1,2†], **Ming Gu**[1,2], **Yu-Shen Liu**[2]

[1]Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University, Beijing, China
[2]School of Software, Tsinghua University, Beijing, China
{h-huang20, wu-yl22, dengc23}@mails.tsinghua.edu.cn, {gaoge, guming, liuyushen}@tsinghua.edu.cn

Abstract

Recently, Gaussian Splatting has sparked a new trend in the field of computer vision. Apart from novel view synthesis, it has also been extended to the area of multi-view reconstruction. The latest methods facilitate complete, detailed surface reconstruction while ensuring fast training speed. However, these methods still require dense input views, and their output quality significantly degrades with sparse views. We observed that the Gaussian primitives tend to overfit the few training views, leading to noisy floaters and incomplete reconstruction surfaces. In this paper, we present an innovative sparse-view reconstruction framework that leverages intra-view depth and multi-view feature consistency to achieve remarkably accurate surface reconstruction. Specifically, we utilize monocular depth ranking information to supervise the consistency of depth distribution within patches and employ a smoothness loss to enhance the continuity of the distribution. To achieve finer surface reconstruction, we optimize the absolute position of depth through multi-view projection features. Extensive experiments on DTU and BlendedMVS demonstrate that our method outperforms state-of-the-art methods with a speedup of 60x to 200x, achieving swift and fine-grained mesh reconstruction without the need for costly pre-training.

Introduction

Reconstructing surfaces from multi-view images (Ramon et al. 2021; Chen et al. 2024; Zhang, Liu, and Han 2024) is a long-standing task in 3D vision, graphics, and robotics. Multi-View Stereo (Schönberger et al. 2016; Yao et al. 2018; Xu and Tao 2019) is a traditional reconstruction method consisting of processes such as feature extraction, depth estimation, and depth fusion. This technique achieves favorable results with dense views, but struggles in sparse view reconstruction due to the lack of matching features.

Over the recent years, neural implicit reconstruction has rapidly progressed based on neural radiance fields (Mildenhall et al. 2020). Some methods (Wang et al. 2021; Yariv et al. 2021; Fu et al. 2022) employ neural rendering to optimize implicit geometry fields and color fields from multi-view images. They can achieve smooth and complete surfaces with implicit geometric representations. However, implicit geometric fields tend to overfit with a limited number of input views, leading to geometric collapse.

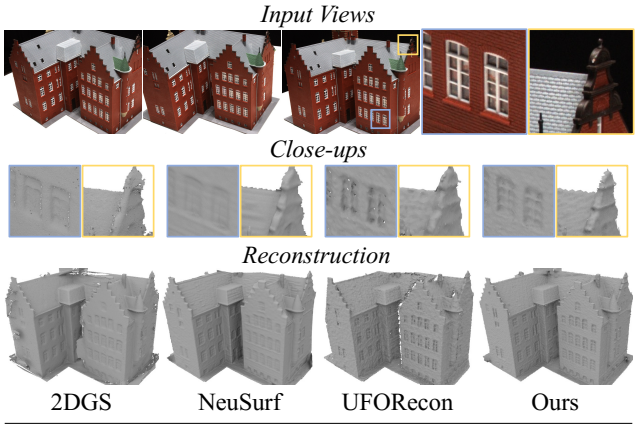


Figure 1: Surface reconstruction from 3-view images of DTU scan 24. The training time (*pre-training time) and Chamfer Distance (CD↓) are shown below the image. The trendiest general method 2DGS (Huang et al. 2024a) is fast but yields coarse results. The state-of-the-art per-scene optimization method, NeuSurf (Huang et al. 2024b), and the generalization method, UFORecon (Na et al. 2024), produce suboptimal surfaces and require long training time. In contrast, our method achieves swift and detailed reconstruction.

To address this issue, two types of neural implicit methods for sparse view reconstruction have been developed. The first type is a generalizable approach (Long et al. 2022; Ren et al. 2023; Na et al. 2024), which is trained on large-scale datasets and subsequently applied to infer new scenes. The second type focuses on per-scene optimization (Yu et al. 2022; Huang et al. 2024b), where no pre-training is needed, and the method directly fits different scenes. Although both types of methods achieve satisfactory geometric results, they require either several days of pre-training or optimization for several hours per scene, as shown in Figure 1.

Lately, Gaussian Splatting (Kerbl et al. 2023) has been widely adopted for novel view synthesis due to its high rendering quality and fast training speed. However, 3D Gaus-

*These authors contributed equally.
†Corresponding author.

sians lack the capability to represent scene geometry consistently, leading to imprecise surface reconstruction. To ensure the surface alignment property, some methods (Huang et al. 2024a; Dai et al. 2024; Turkulainen et al. 2024) modify the shape of Gaussian primitives and the splatting techniques. With depth maps fusion, the geometry of the object can be reconstructed completely and precisely. These methods retain the fast training speed of Gaussian Splatting in multi-view reconstruction. However, with fewer input views, geometric consistency decreases, leading to inaccurate Gaussian primitive localization and flawed depth rendering. This results in noisy and incomplete output meshes.

In this paper, we present a novel sparse view reconstruction framework that leverages the efficient pipeline of Gaussian Splatting along with two consistency constraints to enhance both reconstruction speed and accuracy. Specifically, we transform the 3D ellipsoid Gaussian into a 2D ellipse Gaussian for more precise geometric representation and employ 2D Gaussian rendering to optimize the attributes of the Gaussian primitives. To mitigate local noise induced by overfitting, we segment the image into patches and regulate the ranking relationships within these patches using monocular depth information. Additionally, we introduce a smoothing loss to address abrupt depth changes in textureless regions, thereby ensuring the continuity of the depth distribution. The intra-view depth consistency aided in achieving coarse reconstruction geometry, yet compromised numerous details. To resolve the issue of over-smoothing, we align the reprojection features of depth-rendered points to ensure precise multi-view feature consistency, which significantly enhances the quality of surface reconstruction.

Our contributions are summarized as follows.

- We propose FatesGS for sparse-view surface reconstruction, taking full advantage of the Gaussian Splatting pipeline. Compared with previous methods, our approach neither requires long-term per-scene optimization nor costly pre-training.
- We leverage intra-view depth consistency to facilitate the learning of coarse geometry. Furthermore, we optimize the multi-view feature consistency of depth-rendered points to enhance the learning of detailed geometry.
- We achieve state-of-the-art results in sparse view surface reconstruction under two distinct settings on the widely used DTU and BlendedMVS datasets.

Related Works

Multi-View Stereo (MVS)

In the field of 3D reconstruction, MVS methods have established themselves based on their scalability, robustness, and accuracy. Point clouds (Lhuillier and Quan 2005; Furukawa and Ponce 2010), depth maps (Galliani, Lasinger, and Schindler 2015; Schönberger et al. 2016; Xu and Tao 2019), and voxel grids (Kostrikov, Horbert, and Leibe 2014; Ji et al. 2017; Choe et al. 2021) are used as 3D representations in MVS pipeline to accomplish geometry reconstruction. While these methods can achieve dense reconstruction, they often produce limited results in texture-less regions.

Neural Implicit Reconstruction

NeRF (Mildenhall et al. 2020) represents a scene as density and radiance fields, which are optimized using volumetric rendering. Inspired by this, NeuS (Wang et al. 2021), VolSDF (Yariv et al. 2021), and subsequent optimization methods (Yu et al. 2022; Fu et al. 2022; Darmon et al. 2022; Li et al. 2023) transform signed distance function(SDF) into density, reconstructing multi-view images into implicit surfaces. However, these methods focus on dense view reconstruction, which places high demands on the input.

To enable sparse view reconstruction, both generalization and per-scene optimization methods have been proposed recently. The generalizable methods (Long et al. 2022; Ren et al. 2023; Xu et al. 2023; Peng et al. 2023; Liang, He, and Chen 2024; Na et al. 2024) are trained on large-scale datasets and then generalized to new scenes. These methods require significant time on high-performance GPUs (usually several days) to learn the correspondence between 3D geometry and 2D views in advance. In contrast, per-scene optimization methods (Yu et al. 2022; Vora, Patil, and Zhang 2023; Wang et al. 2023; Somraj and Soundararajan 2023; Somraj, Karanayil, and Soundararajan 2023; Huang et al. 2024b) do not require training on large-scale datasets but instead directly fit the 3D geometry from the sparse images of a given scene. Due to the lack of learned correspondence, these methods often require several hours to fit from scratch.

Gaussian Splatting

3D Gaussian Splatting (3DGS) (Kerbl et al. 2023) represents the latest advancement in novel view synthesis, leveraging explicit Gaussian primitives for scene representation. By integrating a splatting-rendering pipeline, 3DGS maintains high-quality rendering while enabling real-time performance. However, 3DGS still requires dense view input and tends to overfit the training views when dealing with sparse input. To address this issue, some studies introduce monocular depth regularization (Zhu et al. 2023; Chung, Oh, and Lee 2023; Li et al. 2024; Han et al. 2024) for sparse views to constrain geometric relationships, thereby reducing Gaussian overfitting for high-quality rendering.

Lately, to extend the advantages of Gaussian Splatting into the field of surface reconstruction, some work (Chen, Li, and Lee 2023; Guédon and Lepetit 2023; Lyu et al. 2024) have enhanced surface representation by integrating regularization terms and Signed Distance Function (SDF) implicit fields at the cost of reduced training speed. 2DGS (Huang et al. 2024a) and Gaussian Surfels (Dai et al. 2024) flatten the 3D ellipsoid into 2D ellipse to obtain more stable and consistent geometric surfaces. Although these methods achieve satisfactory results with dense views, they can only produce noisy and incomplete surfaces under sparse input.

Method

Our goal is to reconstruct the high-quality geometry $\mathcal{S}$ of a scene from a collection of sparse-view images $\mathcal{I} = \{I_i \mid i \in 1, 2 \ldots, N\}$, with poses $\mathcal{T} = \{T_i \mid i \in 1, 2 \ldots, N\}$. In this paper, we propose *FatesGS*, a Gaussian surface reconstruction approach with sparse views, as shown in Figure 2.

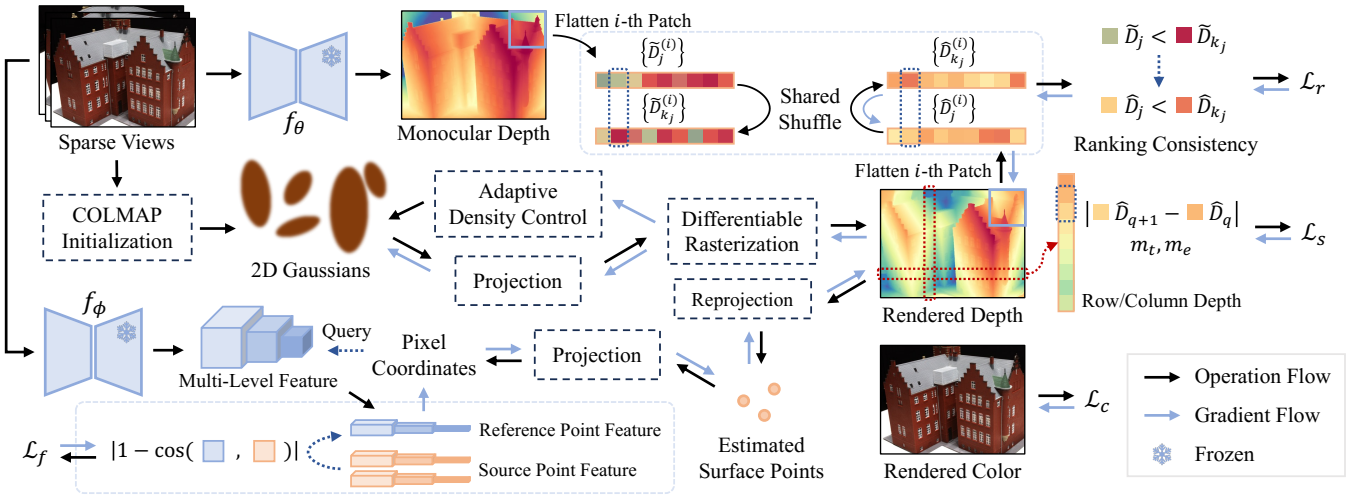


Figure 2: Overview of FatesGS. Starting with a set of sparse input views, we initialize 2D Gaussians using COLMAP and employ splatting to render RGB images and depth maps. To enhance the geometric learning process, we integrate ranking information from monocular depth estimation and apply depth smoothing to ensure intra-view depth consistency. To further refine the geometry, we align the multi-view features extracted by projecting estimated surface points onto the source images.

Since the Gaussian splatting process involves localized operations for fast rendering and optimization, it tends to produce floating artifacts and view misalignments when only a few views are provided (Sun et al. 2024a). This results in the collapse of the learned geometry. Our motivation is to leverage intra-view depth consistency to prevent local noise for coarse geometry and multi-view feature alignment to maintain coherent observations for detailed geometry.

Learning Multi-View Geometry by Gaussian Splatting

3DGS (Kerbl et al. 2023) represents the scene as a series of 3D Gaussians. Each Gaussian can be defined by center position $\boldsymbol{\mu}$, scaling matrix S, rotation matrix R, opacity o, and SH coefficients. The view-dependent appearance can be rendered with local affine transformation (Zwicker et al. 2001) and alpha blending techniques. Although 3DGS can achieve good rendering results, the geometric results remain noisy.

Following the previous work (Huang et al. 2024a; Dai et al. 2024), we flatten the 3D ellipsoid into 2D ellipse to enable the primitives to better cover the surface of objects. Scaling matrix S and rotation matrix R can be expressed as $S = (s_1, s_2)$, $R = (\boldsymbol{t}_1, \boldsymbol{t}_2, \boldsymbol{t}_1 \times \boldsymbol{t}_2)$. Then the 2D ellipse can be defined within a local tangent plane in world space as:

$$P(u, v) = \boldsymbol{\mu} + s_1 \boldsymbol{t}_1 u + s_2 \boldsymbol{t}_2 v. \tag{1}$$

For the point $\boldsymbol{u} = (u, v)$ within the uv plane, its corresponding 2D Gaussian value can be determined using the standard Gaussian function:

$$\mathcal{G}(\boldsymbol{u}) = \exp\left(-\frac{u^2 + v^2}{2}\right). \tag{2}$$

During the scene optimization process, the parameters of the 2D Gaussian primitives are all designed to be learnable. The view-dependent color $\boldsymbol{c}$ is obtained through spherical harmonic (SH) coefficients. For Gaussian rasterization, 2D Gaussians are depth-sorted and then integrated into an image with alpha blending from front to back. Given a pixel from one image, the rendered color $\hat{\boldsymbol{C}}(\boldsymbol{r})$ of a homogeneous ray $\boldsymbol{r}$ emitted from the camera can be expressed as:

$$\hat{\boldsymbol{C}}(\boldsymbol{r}) = \sum_{i=1} \boldsymbol{c}_i \omega_i, \tag{3}$$

$$\omega_i = o_i \mathcal{G}_i\left(\boldsymbol{u}(\boldsymbol{r})\right) \prod_{j=1}^{i-1} \left(1 - o_j \mathcal{G}_j(\boldsymbol{u}(\boldsymbol{r}))\right), \tag{4}$$

where $\boldsymbol{c}_i$ is the i-th view-dependent color, ω_i is blending weight of the i-th intersection.

Similarly, the rendered depth $\hat{D}(\boldsymbol{r})$ for the homogeneous ray $\boldsymbol{r}$ can be accumulated by alpha blending as:

$$\hat{D}(\boldsymbol{r}) = \frac{\sum_{i=1} \omega_i d_i}{\sum_{i=1} \omega_i + \epsilon}. \tag{5}$$

Following (Huang et al. 2024a), the i-th intersection depth d_i is obtained by the ray-splat intersection algorithm.

Intra-View Depth Consistency

Since Gaussian Splatting lacks the concept of geometric fields, surface reconstruction relies on rendered depth extraction. Direct depth optimization seems to avoid overfitting and address geometric noise effectively. Employing absolute scaling for monocular depth to supervise rendered depth (Yu et al. 2022; Xiong et al. 2023) and enhancing the correlation between monocular and rendered depth (Zhu et al. 2023) are regarded as effective depth regularization techniques. However, it has been proven that these strategies might result in a noisy distribution of Gaussian primitives (Sun et al. 2024b). To avoid geometric collapse caused by hard constraints, we utilized monocular depth information to

maintain the ranking consistency of local rendering depth. Since long-range depth ambiguity may exist in monocular depth, we performed local depth information distillation on a patch-by-patch basis.

Specifically, We divide the image I into patches, each of size $M \times M$. The i-th patch $\mathcal{P}_i$ is represented as a list of pixels:

$$\mathcal{P}_i = \left\{ \boldsymbol{p}_j^{(i)} \;\middle|\; j \in 1, \ldots, M^2 \right\}. \tag{6}$$

To simplify, the pixels in the patch are shuffled, denoted as:

$$\mathcal{P}_i' = \text{shuffle}\left(\mathcal{P}_i\right) = \left\{ \boldsymbol{p}_{k_j}^{(i)} \;\middle|\; j \in 1, \ldots, M^2 \right\}. \tag{7}$$

For each pixel in $\mathcal{P}_i$ and $\mathcal{P}_i'$, we obtain its rendered depth $\hat{D}$ and monocular depth $\widetilde{D}$. A patch-based depth ranking loss is then expressed as:

$$\mathcal{L}_r = \sum_{i,j} \sigma\left(\text{sgn}\left(\widetilde{D}_{k_j}^{(i)} - \widetilde{D}_j^{(i)} \right) \cdot \left(\hat{D}_j^{(i)} - \hat{D}_{k_j}^{(i)} \right) + m \right), \tag{8}$$

where $\sigma(\cdot)$ represents the ReLU function, and m is a small positive threshold.

The patch-based depth ranking loss ensures the overall distribution consistency of Gaussian primitives. However, noisy primitives still exist in texture-less areas, resulting in abrupt depth changes. Therefore, we propose a smoothing loss for the depth of adjacent pixels to enhance the distribution continuity of the reconstructed surface:

$$\mathcal{L}_s = \sum_{i,j,k} \sum_{\left|\tilde{D}_k - \tilde{D}_{(i,j)}\right| < m_e} \sigma\left(\left| \hat{D}_k - \hat{D}_{(i,j)} \right| - m_t \right). \tag{9}$$

Here, $\hat{D}_{(i,j)}$ denotes the rendered depth value of the pixel at i-th row and j-th column within the whole image. Small positive thresholds m_e and m_t are utilized to recognize edges and avoid over-smoothing. $k \in \{(i+1, j), (i, j+1)\}$.

Multi-View Feature Alignment

Intra-view depth consistency helps maintain the overall shape and structure of the reconstructed object. While ranking and smoothing are effective in reducing artifacts and preserving the coarse geometry, they fall short in refining the finer details of the reconstruction. Multi-view geometry may present a reliable solution. The traditional Multi-View Stereo (MVS) reconstruction pipeline typically employs photometric consistency across multiple views to refine the surface. Inspired by that, a straightforward idea is to project the 3D points corresponding to the depth of each view onto other views and then compute the color difference on the projected views.

However, due to the influence of lighting, the colors may differ across different viewpoints (Zhan et al. 2018). When there are only a few input views, the number of reference views for projection is limited, and the spacing between views is greater compared to dense views. As a result, the influence of lighting on the color of surface points becomes more pronounced. To resolve these issues, we have designed a multi-level feature projection loss.

Let $I_i^{(l)}$ denote the image whose resolution is downscaled by a scale factor l from the original image I_i, the image set of the downscaled images then can be marked as

$$\mathcal{I}^{(l)} = \left\{ I_i^{(l)} \;\middle|\; i \in 1, 2, \ldots, N \right\},\ l \in 1, 2, \ldots, 2^L. \tag{10}$$

Multi-view feature at single level l can be calculated using a frozen feature extraction network f_ϕ:

$$\mathcal{F}^{(l)} = f_\phi(\mathcal{I}^{(l)}) = \left\{ \boldsymbol{F}_i^{(l)} \;\middle|\; i \in 1, 2, \ldots, N \right\}. \tag{11}$$

Let I_r, I_s denote the reference view image and one of its source view images, respectively. For a pixel $\boldsymbol{p}_{r,i}$ of I_r, with its rendered depth $\hat{D}_{r,i}$, we can calculate the corresponding spatial point $\boldsymbol{x}_{r,i}$ and its projected pixel coordinate $\boldsymbol{p}_{s,i}$ to the source view I_s by

$$\boldsymbol{x}_{r,i} = \boldsymbol{o}_r + \hat{D}_{r,i} \cdot \boldsymbol{d}_{r,i}, \tag{12}$$

$$\boldsymbol{p}_{s,i} = K P_s^{-1} \boldsymbol{x}_{r,i}, \tag{13}$$

where K and P_s represent the intrinsic matrix and camera pose of the source view image I_s. $\boldsymbol{d}_{r,i}$ is the normalized direction vector of the ray omitted form $\boldsymbol{o}_r$ passing through $\boldsymbol{p}_{r,i}$. Then the feature loss can be acquired by

$$\mathcal{L}_f = \sum_{s,i,l} \frac{1}{l} \cdot v_{r,s,i} \left| 1 - \cos\left(\boldsymbol{F}_r^{(l)}\left(\boldsymbol{p}_{r,i}\right), \boldsymbol{F}_s^{(l)}\left(\boldsymbol{p}_{s,i}\right) \right) \right|. \tag{14}$$

Since surface points may be occluded when projected onto the source views. We design visibility item $v_{r,s,i}$, which indicates the visibility of $\boldsymbol{x}_{r,i}$ from the viewpoint $\boldsymbol{o}_s$. For spatial points along the ray $\boldsymbol{r}_{s,i}$ which is emitted from $\boldsymbol{o}_s$ and passes through $\boldsymbol{p}_{s,i}$, only the nearest one is considered visible and its visibility item is set as 1, while the others are set as 0. The process can be expressed as

$$\begin{aligned} v_{r,s,i} &= \left[i = \arg\min_t \left(\|\boldsymbol{x}_{r,t} - \boldsymbol{o}_s\| \right) \right], \\ &\text{where } t \in \left\{ t \;\middle|\; \boldsymbol{p}_{s,i} = K P_s^{-1} \boldsymbol{x}_{r,t} \right\}. \end{aligned} \tag{15}$$

$[\cdot]$ represent the Iverson Bracket.

Loss Functions

The overall loss functions are defined as follows:

$$\mathcal{L} = \mathcal{L}_c + \lambda_1 \mathcal{L}_r + \lambda_2 \mathcal{L}_s + \lambda_3 \mathcal{L}_f + \lambda_4 \mathcal{L}_d + \lambda_5 \mathcal{L}_n, \tag{16}$$

where $\mathcal{L}_r$ and $\mathcal{L}_s$ represent the ranking and smoothing losses from intra-view depth consistency, respectively, and $\mathcal{L}_f$ denotes the multi-view feature loss.

According to 3DGS (Kerbl et al. 2023), the $\mathcal{L}_1$ loss and $\mathcal{L}_{D-SSIM}$ loss are utilized for color supervision $\mathcal{L}_c$. This can be formulated as follows, with $\lambda = 0.2$:

$$\mathcal{L}_c = (1 - \lambda)\mathcal{L}_1 + \lambda \mathcal{L}_{D-SSIM}. \tag{17}$$

As with 2DGS (Huang et al. 2024a), depth distortion loss and normal consistency loss are used as regularization terms to optimize surface geometry.

$$\mathcal{L}_d = \sum_{i,j} \omega_i \omega_j \left| d_i - d_j \right|, \quad \mathcal{L}_n = \sum_i \omega_i (1 - \boldsymbol{n}_i^T \boldsymbol{N}). \tag{18}$$

Here, ω and d are computed during the Gaussian Splatting process, $\boldsymbol{n}_i^T$ represents the estimated normal near the depth point, and $\boldsymbol{N}$ is the estimated normal near the depth point.

Scan ID	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
COLMAP	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
TransMVSNet	1.07	3.14	2.39	1.30	1.35	1.61	0.73	1.60	1.15	0.94	1.34	**0.46**	0.60	1.20	1.46	1.35
SparseNeuS$_{ft}$	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
VolRecon	1.20	2.59	1.56	1.08	1.43	1.92	1.11	1.48	1.42	1.05	1.19	1.38	0.74	1.23	1.27	1.38
ReTR	1.05	2.31	1.44	0.98	1.18	1.52	0.88	1.35	1.30	0.87	1.07	0.77	0.59	1.05	1.12	1.17
C2F2NeuS	1.12	2.42	1.40	**0.75**	1.41	1.77	0.85	1.16	1.26	0.76	0.91	0.60	0.46	0.88	0.92	1.11
GenS$_{ft}$	0.91	2.33	1.46	**0.75**	**1.02**	1.58	0.74	1.16	1.05	0.77	0.88	0.56	0.49	0.78	0.93	1.03
UFORecon	0.76	2.05	1.31	0.82	1.12	**1.18**	0.74	1.17	1.11	0.71	0.88	0.58	0.54	0.86	0.99	0.99
NeuS	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	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
MonoSDF	2.85	3.91	2.26	1.22	3.37	1.95	1.95	5.53	5.77	1.10	5.99	2.28	0.65	2.65	2.44	2.93
NeuSurf	0.78	2.35	1.55	**0.75**	1.04	1.68	**0.60**	1.14	**0.98**	0.70	**0.74**	0.49	**0.39**	**0.75**	**0.86**	0.99
3DGS	3.38	4.19	2.99	1.76	3.38	3.80	5.21	2.91	4.29	3.18	3.23	5.18	2.78	3.48	3.32	3.54
Gaussian Surfels	3.56	5.42	3.95	3.68	4.61	2.72	4.42	5.22	4.71	3.46	4.07	5.42	2.44	3.27	4.00	4.06
2DGS	1.26	2.95	1.73	0.96	1.68	1.97	1.58	1.87	2.50	1.02	1.93	1.91	0.72	1.85	1.37	1.69
Ours	**0.67**	**1.94**	**1.17**	0.77	1.28	1.23	0.63	**1.05**	**0.98**	**0.69**	0.75	0.48	0.41	0.78	0.90	**0.92**

Table 1: The quantitative comparison results of Chamfer Distance (CD↓) on DTU dataset (large-overlap setting). In this table, the best results are in bold, the second best are underlined.

Experiments and Analysis

To demonstrate the effectiveness and generalization performance of our approach, we compare our evaluation results with previous state-of-the-art methods in terms of reconstruction accuracy and training efficiency. Additionally, we provide a detailed ablation study and analysis to validate the efficacy of each component of our proposed method.

Experimental Settings

Datasets. We evaluate our approach on DTU dataset (Jensen et al. 2014), which is extensively utilized in previous surface reconstruction research. DTU comprises 15 scenes, each with 49 or 69 images at a resolution of 1600 × 1200. We follow the previous work (Huang et al. 2024b) to train and evaluate the model on 3 views of both the large-overlap (SparseNeuS) setting and the little-overlap (PixelNeRF) setting. The images are downscaled into 800 × 600 pixels during training procedure, following (Huang et al. 2024a). To assess generalization performance, we further test our method on BlendedMVS dataset (Yao et al. 2020) with randomly selected 3 input views per scene at a resolution of 768 × 576. Consistent with sparse-view settings from previous works, the camera poses are assumed to be known.

Baselines. We compare our approach with abundant SOTA methods of various categories. **i.** MVS methods: COLMAP (Schonberger and Frahm 2016) and TransMVSNet (Ding et al. 2022). **ii.** Generalizable sparse-view neural implicit reconstruction methods: SparseNeuS (Long et al. 2022), VolRecon (Ren et al. 2023), ReTR (Liang, He, and Chen 2024), C2F2NeuS (Xu et al. 2023), GenS (Peng et al. 2023) and UFORecon (Na et al. 2024). **iii.** Per-scene optimization neural implicit methods: NeuS (Wang et al. 2021), VolSDF (Yariv et al. 2021), MonoSDF (Yu et al. 2022) and NeuSurf (Huang et al. 2024b). **iv.** Gaussian splatting based methods: 3DGS (Kerbl et al. 2023), Gaussian Surfels (Dai et al. 2024) and 2DGS (Huang et al. 2024a). For a fair comparison, we initialize 3DGS and 2DGS with the same point clouds used in our method. We also adopt the same TSDF depth fusion approach as ours for 3DGS to extract meshes.

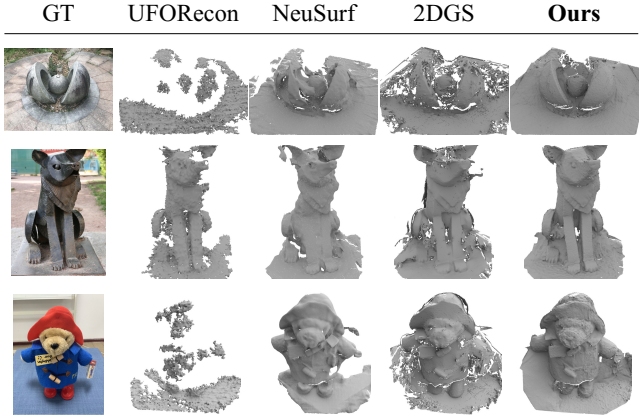


Figure 3: Visual comparison of 3-view reconstruction on BlendedMVS dataset.

Implementation Details. Following previous research, we use COLMAP (Schonberger and Frahm 2016) for Gaussians initialization. Our framework is built upon 2DGS (Huang et al. 2024a) and 3DGS (Kerbl et al. 2023). We adopt Vis-MVSNet (Zhang et al. 2020) as the feature extraction network f_ϕ and Marigold (Ke et al. 2024) as the monocular depth estimation model f_θ. All experiments presented in this paper are conducted on a single NVIDIA RTX 3090 GPU.

Comparisons

Sparse View Reconstruction. The quantitative results of geometry reconstruction from sparse input views on the

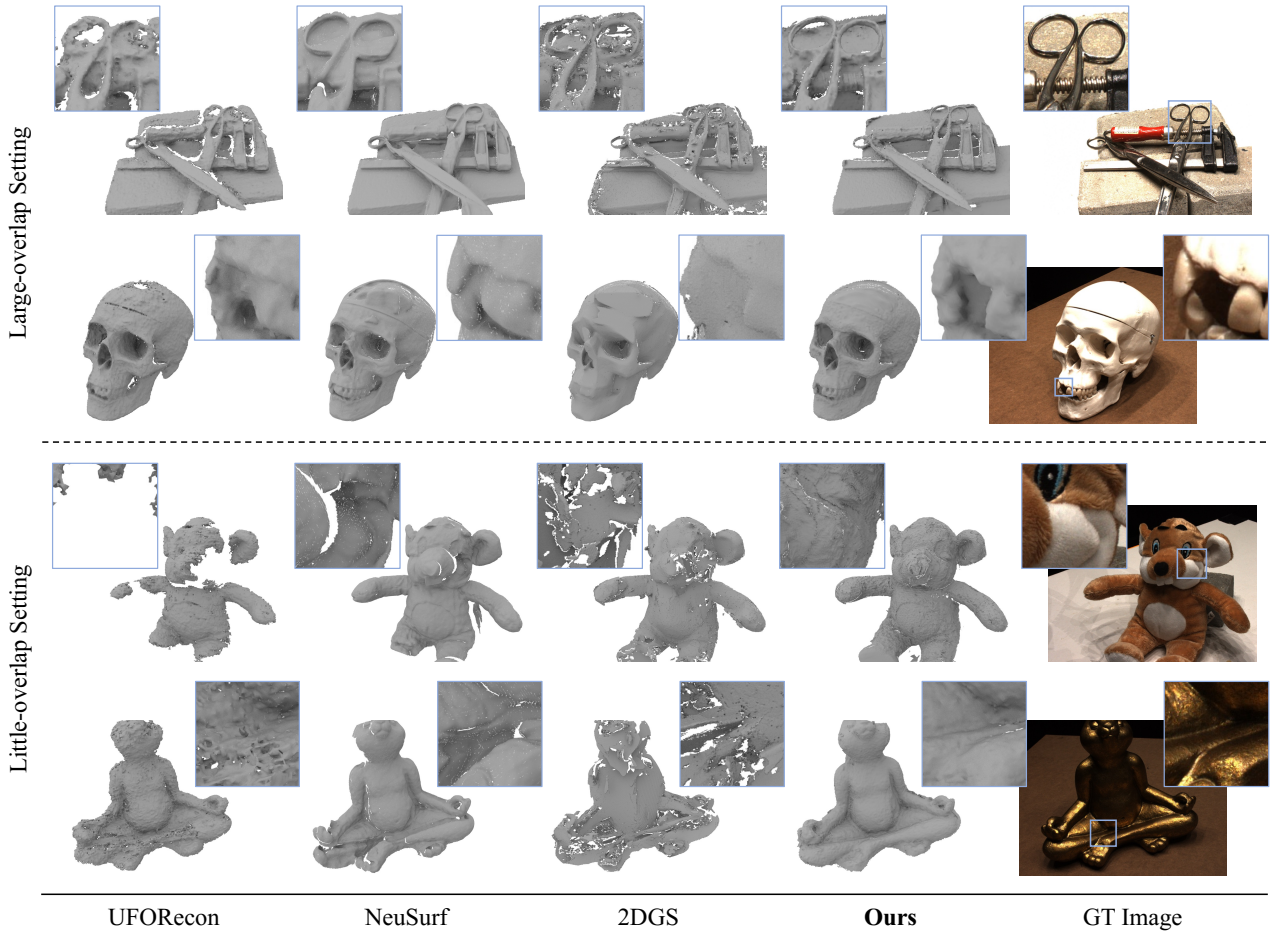


Figure 4: Qualitative comparison of reconstruction results on the DTU with different sparse settings.

DTU dataset (large-overlap setting) are presented in Table 1. Additional experimental results (e.g., little-overlap setting) are presented in the supplementary materials. Our method achieves the best mean Chamfer Distance (CD) performance across 15 scenes compared to others. As illustrated in Figure 4, our approach achieves more comprehensive global geometry and preserves finer details. This highlights our method's superior capability in multi-view feature extraction. Moreover, in contrast to NeuSurf, our method successfully avoids over-smoothing of the geometric surfaces.

Reconstruction results on BlendedMVS are shown in Figure 3. Our method exhibits consistent and stable performance across datasets with the same set of hyperparameters. In contrast, UFORecon, which is currently the most recent generalizable method, has not undergone extensive training on this dataset, resulting in significant reconstruction defects and noise. NeuSurf, being the latest per-scene optimization method, produces surfaces in the SDF field that are overly smooth, leading to a loss of local texture details. 2DGS, a leading Gaussian splatting surface reconstruction method, struggles with sparse image coverage. Insufficient geometric consistency can lead to flawed depth rendering and suboptimal reconstruction results.

Method	Training Time		GPU Mem.
	Pre-Training	Per-Scene	
SparseNeuS$_{ft}$	2.5 days	19 mins	7 GB
GenS$_{ft}$	$\sim$ 1 day*	25 mins	19 GB*
VolRecon	$\sim$ 2 days		17 GB
ReTR	$\sim$ 3 days	-	22 GB
UFORecon	$\sim$ 10 days		23 GB
MonoSDF		6 hours	14 GB
NeuSurf	-	14 hours	8 GB
Ours		**14 mins**	**4 GB**

Table 2: Comparison with the efficiency of sparse-view reconstruction methods. The listed GPU memory values are approximate maximum occupancies during training. *We used 2 NVIDIA RTX 3090 GPUs for GenS pre-training.

Efficiency. We conduct an efficiency study on all specialized sparse-view reconstruction methods using the DTU SparseNeuS 3-view setting, as detailed in Table 2. The presented results are obtained from tests conducted on a single NVIDIA RTX 3090 GPU. To ensure a fair compari-

son, all models are configured with settings optimized for peak performance. In previous methods, generalizable approaches require extensive pre-training, often taking several days. Per-scene optimization methods, on the other hand, need several hours of training for each scene. In contrast, our method completes training in just a few minutes and uses significantly less GPU memory.

Depth Prediction. We trained our model using three seen views and tested it on the same three views, along with three additional unseen views. We then calculated the error with the ground truth depth, and compared the results with methods Marigold (Ke et al. 2024) and 2DGS (Huang et al. 2024a), as shown in Table 3. Marigold is a universal method for monocular depth prediction, limited to predicting relative depth. To facilitate comparison, we rescale the predicted results to real-world dimensions using the ground truth depth. The results demonstrate that our method significantly outperforms both the 2D Gaussian Splatting (2DGS) backbone and the prior method, Marigold, in depth prediction. Our approach effectively integrates monocular depth information with the Gaussian splatting pipeline, leading to more consistent and accurate multi-view depth learning.

Method	Marigold	2DGS	Ours
$< 1 \uparrow$	5.01 / 4.40	35.74 / 31.24	**77.35 / 73.52**
$< 2 \uparrow$	9.98 / 8.84	57.26 / 50.71	**89.58 / 87.38**
$< 4 \uparrow$	19.55 / 17.82	73.78 / 66.92	**94.34 / 92.69**
Abs. $\downarrow$	15.58 / 15.12	7.46 / 18.07	**2.41 / 3.43**
Rel. $\downarrow$	2.37 / 2.31	1.05 / 2.66	**0.33 / 0.49**

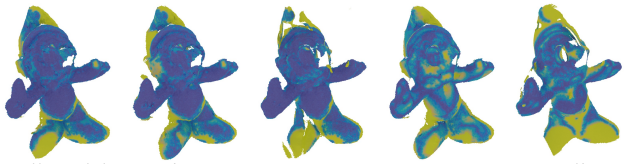
Table 3: Depth map evaluation results on DTU (seen / unseen). The result of mean absolute error (Abs.) is in millimeters. The result of threshold percentage ($< 1\text{mm}, < 2\text{mm}$ and $< 4\text{mm}$) and mean absolute relative error (Rel.) are in percentage (%). The best results are highlighted in bold.

Ablation Study

The Proposed Components. To demonstrate the effectiveness and necessity of each proposed component, we isolate individual design choices and measure their impact on reconstruction quality. Our experiments are conducted on the DTU dataset using the little-overlap setting, maintaining the same hyperparameters as in the main experiment. The mean Chamfer Distance (CD) values of all 15 scenes are reported in Table 4. Furthermore, the ablation results for scan 83 are visualized in Figure 5. Removing each of the proposed optimization losses results in varying degrees of performance decline, demonstrating the effectiveness of each component. Notably, the model with only the intra-view depth ranking loss ($\mathcal{L}_r$) and the smoothing loss ($\mathcal{L}_s$) performs worse than the baseline model, which does not include any of the three losses. This indicates that the contributions of the three optimization losses to the full model are neither isolated nor merely additive. As shown in Figure 5, $\mathcal{L}_r$ and $\mathcal{L}_s$ provide globally complete and coarsely correct geometric guidance. However, they cannot ensure local details due to the lack of absolute scale information. After incorporating feature loss ($\mathcal{L}_f$), we observe that the reconstructed surface details are significantly enhanced, effectively avoiding excessive smoothing.

$\mathcal{L}_r$	$\mathcal{L}_s$	$\mathcal{L}_f$	Mean CD$\downarrow$
			2.47
✓	✓		2.56
✓		✓	1.56
	✓	✓	1.62
✓	✓	✓	**1.37**

Table 4: Comparison of reconstruction from the ablation study for the little-overlap setting on the DTU dataset.



Full model	w/o $\mathcal{L}_s$	w/o $\mathcal{L}_r$	w/o $\mathcal{L}_f$	Baseline
CD: **1.19**	CD: 1.50	CD: 1.58	CD: 2.14	CD: 2.21

Figure 5: Visual comparison of ablation study on DTU scan 83. The transition of the error maps from blue to yellow indicates larger reconstruction errors.

The Number of Training Views. To validate the impact of image counts on our proposed method, we varied the number of views, and the results are summarized in Table 5. As the number of images increases, the reconstruction quality improves progressively. Incorporating additional views can enhance multi-view consistency, ensure stable reconstruction results, and prevent overfitting.

Number of Views	3	6	9	Full
Mean CD$\downarrow$	0.92	0.85	0.79	**0.61**

Table 5: Ablation study of number of views on DTU dataset. The best result is highlighted in bold.

Conclusion

In this paper, we present FatesGS, a novel method for sparse view surface reconstruction utilizing a Gaussian Splatting pipeline. To combat geometric collapse caused by overfitting in sparse views, we enhance the learning of coarse geometry through intra-view depth consistency. For finer geometric details, we optimize multi-view feature consistency. Our method is robust across various sparse settings and does not require large-scale training. Unlike previous methods, our approach eliminates the need for long-term per-scene optimization and expensive in-domain prior training. We demonstrate state-of-the-art results in sparse view surface reconstruction under two distinct settings, validated on the widely used DTU and BlendedMVS datasets.

Acknowledgments

The corresponding author is Ge Gao. This work was supported by Beijing Science and Technology Program (Z231100001723014).

References

Chen, H.; Li, C.; and Lee, G. H. 2023. Neusg: Neural implicit surface reconstruction with 3d gaussian splatting guidance. *arXiv preprint arXiv:2312.00846*.

Chen, H.; Wei, F.; Li, C.; Huang, T.; Wang, Y.; and Lee, G. H. 2024. VCR-GauS: View Consistent Depth-Normal Regularizer for Gaussian Surface Reconstruction. *arXiv preprint arXiv:2406.05774*.

Choe, J.; Im, S.; Rameau, F.; Kang, M.; and Kweon, I.-S. 2021. VolumeFusion: Deep Depth Fusion for 3D Scene Reconstruction. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 16066–16075.

Chung, J.; Oh, J.; and Lee, K. M. 2023. Depth-regularized optimization for 3d gaussian splatting in few-shot images. *arXiv preprint arXiv:2311.13398*.

Dai, P.; Xu, J.; Xie, W.; Liu, X.; Wang, H.; and Xu, W. 2024. High-quality Surface Reconstruction using Gaussian Surfels. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery.

Darmon, F.; Bascle, B.; Devaux, J.-C.; Monasse, P.; and Aubry, M. 2022. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6260–6269.

Ding, Y.; Yuan, W.; Zhu, Q.; Zhang, H.; Liu, X.; Wang, Y.; and Liu, X. 2022. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8585–8594.

Fu, Q.; Xu, Q.; Ong, Y. S.; and Tao, W. 2022. Geo-Neus: Geometry-Consistent Neural Implicit Surfaces Learning for Multi-view Reconstruction. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 3403–3416. Curran Associates, Inc.

Furukawa, Y.; and Ponce, J. 2010. Accurate, Dense, and Robust Multiview Stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8): 1362–1376.

Galliani, S.; Lasinger, K.; and Schindler, K. 2015. Massively Parallel Multiview Stereopsis by Surface Normal Diffusion. In *2015 IEEE International Conference on Computer Vision (ICCV)*, 873–881.

Guédon, A.; and Lepetit, V. 2023. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. *arXiv preprint arXiv:2311.12775*.

Han, L.; Zhou, J.; Liu, Y.-S.; and Han, Z. 2024. Binocular-Guided 3D Gaussian Splatting with View Consistency for Sparse View Synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Huang, B.; Yu, Z.; Chen, A.; Geiger, A.; and Gao, S. 2024a. 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery.

Huang, H.; Wu, Y.; Zhou, J.; Gao, G.; Gu, M.; and Liu, Y.-S. 2024b. NeuSurf: On-Surface Priors for Neural Surface Reconstruction from Sparse Input Views. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2312–2320.

Jensen, R.; Dahl, A.; Vogiatzis, G.; Tola, E.; and Aanæs, H. 2014. Large Scale Multi-view Stereopsis Evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 406–413.

Ji, M.; Gall, J.; Zheng, H.; Liu, Y.; and Fang, L. 2017. SurfaceNet: An End-To-End 3D Neural Network for Multiview Stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2307–2315.

Ke, B.; Obukhov, A.; Huang, S.; Metzger, N.; Daudt, R. C.; and Schindler, K. 2024. Repurposing Diffusion-Based Image Generators for Monocular Depth Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4): 1–14.

Kostrikov, I.; Horbert, E.; and Leibe, B. 2014. Probabilistic Labeling Cost for High-Accuracy Multi-view Reconstruction. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 1534–1541.

Lhuillier, M.; and Quan, L. 2005. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3): 418–433.

Li, J.; Zhang, J.; Bai, X.; Zheng, J.; Ning, X.; Zhou, J.; and Gu, L. 2024. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. *arXiv preprint arXiv:2403.06912*.

Li, Z.; Müller, T.; Evans, A.; Taylor, R. H.; Unberath, M.; Liu, M.-Y.; and Lin, C.-H. 2023. Neuralangelo: High-fidelity neural surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8456–8465.

Liang, Y.; He, H.; and Chen, Y. 2024. ReTR: Modeling Rendering Via Transformer for Generalizable Neural Surface Reconstruction. *Advances in Neural Information Processing Systems*, 36.

Long, X.; Lin, C.; Wang, P.; Komura, T.; and Wang, W. 2022. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, 210–227. Springer.

Lyu, X.; Sun, Y.-T.; Huang, Y.-H.; Wu, X.; Yang, Z.; Chen, Y.; Pang, J.; and Qi, X. 2024. 3dgsr: Implicit surface reconstruction with 3d gaussian splatting. *arXiv preprint arXiv:2404.00409*.

Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2020. NeRF: Representing

Scenes as Neural Radiance Fields for View Synthesis. In Vedaldi, A.; Bischof, H.; Brox, T.; and Frahm, J.-M., eds., *Computer Vision – ECCV 2020*, 405–421. Cham: Springer International Publishing. ISBN 978-3-030-58452-8.

Na, Y.; Kim, W. J.; Han, K. B.; Ha, S.; and Yoon, S.-E. 2024. UFORecon: Generalizable Sparse-View Surface Reconstruction from Arbitrary and Unfavorable Sets.

Peng, R.; Gu, X.; Tang, L.; Shen, S.; Yu, F.; and Wang, R. 2023. GenS: Generalizable Neural Surface Reconstruction from Multi-View Images. In *Thirty-seventh Conference on Neural Information Processing Systems (NeurIPS)*.

Ramon, E.; Triginer, G.; Escur, J.; Pumarola, A.; Garcia, J.; Giro-i Nieto, X.; and Moreno-Noguer, F. 2021. H3d-net: Few-shot high-fidelity 3d head reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 5620–5629.

Ren, Y.; Zhang, T.; Pollefeys, M.; Süsstrunk, S.; and Wang, F. 2023. 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*, 16685–16695.

Schonberger, J. L.; and Frahm, J.-M. 2016. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4104–4113.

Schönberger, J. L.; Zheng, E.; Frahm, J.-M.; and Pollefeys, M. 2016. Pixelwise View Selection for Unstructured Multi-View Stereo. In Leibe, B.; Matas, J.; Sebe, N.; and Welling, M., eds., *Computer Vision – ECCV 2016*, 501–518. Cham: Springer International Publishing. ISBN 978-3-319-46487-9.

Somraj, N.; Karanayil, A.; and Soundararajan, R. 2023. Simplenerf: Regularizing sparse input neural radiance fields with simpler solutions. In *SIGGRAPH Asia 2023 Conference Papers*, 1–11.

Somraj, N.; and Soundararajan, R. 2023. Vip-nerf: Visibility prior for sparse input neural radiance fields. In *ACM SIGGRAPH 2023 Conference Proceedings*, 1–11.

Sun, W.; Zhang, Q.; Zhou, Y.; Ye, Q.; Jiao, J.; and Li, Y. 2024a. Uncertainty-guided Optimal Transport in Depth Supervised Sparse-View 3D Gaussian. *arXiv preprint arXiv:2405.19657*.

Sun, W.; Zhang, Q.; Zhou, Y.; Ye, Q.; Jiao, J.; and Li, Y. 2024b. Uncertainty-guided Optimal Transport in Depth Supervised Sparse-View 3D Gaussian. *arXiv preprint arXiv:2405.19657*.

Turkulainen, M.; Ren, X.; Melekhov, I.; Seiskari, O.; Rahtu, E.; and Kannala, J. 2024. DN-Splatter: Depth and Normal Priors for Gaussian Splatting and Meshing. *arXiv preprint arXiv:2403.17822*.

Vora, A.; Patil, A. G.; and Zhang, H. 2023. DiViNeT: 3D Reconstruction from Disparate Views via Neural Template Regularization. *arXiv preprint arXiv:2306.04699*.

Wang, G.; Chen, Z.; Loy, C. C.; and Liu, Z. 2023. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9065–9076.

Wang, P.; Liu, L.; Liu, Y.; Theobalt, C.; Komura, T.; and Wang, W. 2021. NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. *Advances in Neural Information Processing Systems*, 34: 27171–27183.

Xiong, H.; Muttukuru, S.; Upadhyay, R.; Chari, P.; and Kadambi, A. 2023. Sparsegs: Real-time 360 {\deg} sparse view synthesis using gaussian splatting. *arXiv preprint arXiv:2312.00206*.

Xu, L.; Guan, T.; Wang, Y.; Liu, W.; Zeng, Z.; Wang, J.; and Yang, W. 2023. C2F2NeUS: Cascade Cost Frustum Fusion for High Fidelity and Generalizable Neural Surface Reconstruction. *arXiv preprint arXiv:2306.10003*.

Xu, Q.; and Tao, W. 2019. Multi-Scale Geometric Consistency Guided Multi-View Stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yao, Y.; Luo, Z.; Li, S.; Fang, T.; and Quan, L. 2018. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, 767–783.

Yao, Y.; Luo, Z.; Li, S.; Zhang, J.; Ren, Y.; Zhou, L.; Fang, T.; and Quan, L. 2020. BlendedMVS: A Large-scale Dataset for Generalized Multi-view Stereo Networks. *Computer Vision and Pattern Recognition (CVPR)*.

Yariv, L.; Gu, J.; Kasten, Y.; and Lipman, Y. 2021. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34: 4805–4815.

Yu, Z.; Peng, S.; Niemeyer, M.; Sattler, T.; and Geiger, A. 2022. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems*, 35: 25018–25032.

Zhan, H.; Garg, R.; Weerasekera, C. S.; Li, K.; Agarwal, H.; and Reid, I. 2018. Unsupervised learning of monocular depth estimation and visual odometry with deep feature reconstruction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 340–349.

Zhang, J.; Yao, Y.; Li, S.; Luo, Z.; and Fang, T. 2020. Visibility-aware Multi-view Stereo Network. *British Machine Vision Conference (BMVC)*.

Zhang, W.; Liu, Y.-S.; and Han, Z. 2024. Neural signed distance function inference through splatting 3d gaussians pulled on zero-level set. *arXiv preprint arXiv:2410.14189*.

Zhu, Z.; Fan, Z.; Jiang, Y.; and Wang, Z. 2023. FSGS: Real-Time Few-shot View Synthesis using Gaussian Splatting. *arXiv preprint arXiv:2312.00451*.

Zwicker, M.; Pfister, H.; Van Baar, J.; and Gross, M. 2001. EWA volume splatting. In *Proceedings Visualization, 2001. VIS'01.*, 29–538. IEEE.