

# Sparse-View Surface Reconstruction using Gaussian Splatting through High-Confidence Depth Propagation with Normal Priors

Liang Han<sup>1</sup>, Bangcai Wei<sup>2</sup>, Junsheng Zhou<sup>1✉</sup>, Yu-Shen Liu<sup>1✉</sup>, and Zhizhong Han<sup>3</sup>

<sup>1</sup> School of Software, Tsinghua University, Beijing, China  
{hanl23, zhou-js24}@mails.tsinghua.edu.cn,  
liuyushen@tsinghua.edu.cn

<sup>2</sup> China Telecom weibc.gs@chinatelecom.cn

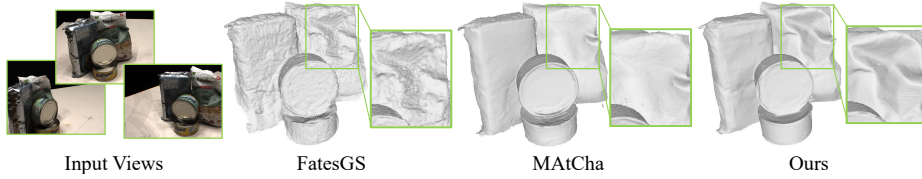
<sup>3</sup> Department of Computer Science, Wayne State University, Detroit, USA  
h312h@wayne.edu

**Abstract.** 3D reconstruction from sparse views is a challenging task in 3D computer vision. Recent studies on 3D Gaussian Splatting (3DGS) have achieved remarkable results with sparse views in novel view synthesis, yet reconstructing high-quality geometric surfaces from sparse views remains a challenge, due to the limited geometry clues and the discreteness of Gaussians. In this paper, we propose a novel 3DGS-based method for high-fidelity surface reconstruction from sparse views. Our key insight is to introduce a normal-guided depth propagation approach, which can extend depth information from high-confidence regions to constrain the depth in low-confidence areas. Additionally, we propose an abnormal depth edge-aware regularization to address depth discontinuities caused by the discreteness of Gaussians. Extensive experiments on DTU and Tanks-and-Temples datasets demonstrate that our method outperforms the state-of-the-art methods in sparse view surface reconstruction. Project page: <https://hanl2010.github.io/DP-GS>.

**Keywords:** Surface Reconstruction · Sparse-View · 3D Gaussian Splatting

## 1 Introduction

3D reconstruction technology has advanced rapidly with the development of deep learning. The emergence of neural radiance field [37] has significantly improved both of the reconstruction quality and efficiency in surface reconstruction tasks [7, 12, 32, 51, 53, 54, 58, 66, 71]. More recently, the emergence of 3D Gaussian Splatting (3DGS) [24] has further improved rendering efficiency, accelerating rendering-based reconstruction methods [4, 6, 14, 18, 35, 59, 72, 77]. However, these methods typically require a dense set of input views. With limited number of input views, their performance degenerates significantly, severely degrading the reconstruction quality.



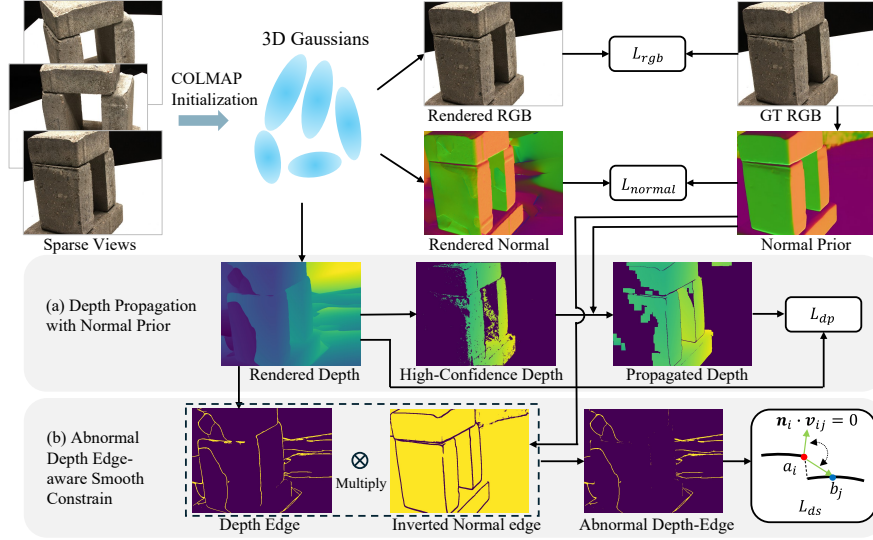
**Fig. 1:** Reconstructed surfaces from three small-overlap views of the DTU dataset. Compared with state-of-the-art 3DGS-based methods FatesGS [19] and MAtCha [13], Our method produces more faithful surface reconstruction.

Two types of solutions have been proposed for addressing this issue: generalizable methods [33, 34, 41, 43, 62] and scene-specific optimization methods [15, 19, 20, 56, 69, 71]. Generalizable methods require data-driven training on large-scale datasets to generalize learned priors to new scenes. However, the pre-training process is time-consuming, and the reconstruction degenerates when the input views do not match the sparse view configuration used during training. Thus, they usually struggle to reconstruct surfaces when testing scenes differ from training scenes a lot.

In contrast, scene-specific optimization methods do not require training on large-scale datasets but instead fitting 3D geometry directly from sparse views. However, some current scene-specific sparse view reconstruction methods [15, 20, 56] adopt NeRF [37] as the representation, which is slow to optimize. Compared to NeRF [37], 3DGS [24] significantly improves the rendering efficiency and quality. Although numerous studies [5, 16, 19, 26, 57, 75, 81] have explored 3DGS-based approaches under sparse-view conditions, most of them focused on novel view synthesis rather than reconstructing the geometry. Due to the limited geometry clue and the discreteness of Gaussians, it is susceptible to ambiguous Gaussian primitives when processing sparse input views, leading to defective depth rendering and the failure of surface reconstruction.

In this paper, we propose a novel 3DGS-based scene-specific optimization method for high fidelity surface reconstruction from sparse views. To address the depth uncertainty, we introduce a depth propagation technique that effectively supervises ambiguous regions. Specifically, we leverage both multi-view geometric and photometric consistency to identify regions with high-confidence depth which is more reliable during the inference. We then utilize the normal prior to propagate these high-confidence depths along the direction perpendicular to the normal. This approach enables robust supervisions on regions with low-confidence depth, ultimately leading to more accurate geometric surface reconstruction.

However, the accuracy of the propagated depths decreases as the propagation range increases, making it challenging to effectively supervise occluded or low-texture regions far from area with high-confidence depth. Moreover, in these regions, depth confidence cannot be determined through multi-view consistency. These facts make some areas may exhibit depth discontinuities during Gaus-



**Fig. 2:** The overview of our method. (a) We leverage a normal prior to propagate depth information from high-confidence regions to low-confidence regions, effectively constraining the depth in regions with lower confidence. (b) The edge-aware regularization for abnormal depths is employed to eliminate depth discontinuities caused by the insufficient constraints of the unstructured Gaussian. In addition, we also use color loss and normal prior loss, as similar previous methods [45, 49].

sian splitting optimizing, especially when only using the normal prior to impose constraints. To maintain depth continuity in the under-constrained regions, we propose an edge-aware smoothing loss for abnormal depth, which mitigates excessive Gaussian discretization and promotes smoother local surfaces, ultimately improving surface reconstruction quality.

In summary, our main contributions are as follows:

- We propose a novel 3DGS-based method for high-fidelity surface reconstruction from sparse views. The key idea lies in a normal-guided depth propagation constraint which effectively supervises the learning of 3D Gaussians with uncertain depths.
- We introduce an edge-aware smoothing loss for abnormal depth to mitigate depth discontinuities caused by excessive Gaussian dispersion in under-constrained regions.
- Extensive experiments on widely-used forward-facing and 360-degree scene datasets demonstrate that our method achieves state-of-the-art results compared to existing sparse view surface reconstruction methods.

## 2 Related Work

### 2.1 Neural Radiance Field

Neural implicit representations have become a powerful paradigm for 3D scene understanding and reconstruction from both images [2, 37, 38, 76, 78, 80] and point clouds [27, 29, 31, 36, 52, 55, 61, 65]. The emergence of NeRF [37] has led to the rapid development of neural volume rendering-based reconstruction methods. NeuS [51], VolSDF [66], and other subsequent methods [7, 12, 32, 54, 58] introduced Signed Distance Functions (SDF) in NeRF as an implicit representation of 3D geometry to achieve multi-view 3D reconstruction. To improve the reconstruction efficiency, researchers have proposed a series of high-fidelity and computationally efficient methods, such as improvements in sampling strategies [9, 46, 79] or the use of hash grids [3, 53].

Recently, 3DGS [24] has achieved unprecedented optimization speed and rendering quality in novel view synthesis tasks. However, it is challenging to extract smooth surfaces due to the discreteness and disorderliness of Gaussian distributions. Subsequent works have addressed this by flattening Gaussian ellipsoids [6, 14, 18, 28, 30], combining with neural implicit radiance fields [35, 70, 77], or improving the accuracy of rendered depth [4, 74] to obtain more stable and accurate geometric surfaces.

Although these methods show significant quality and speed improvements in surface reconstruction, they heavily rely on dense views during the optimization process.

### 2.2 Surface Reconstruction with Sparse Views

Although there have been some methods based on NeRF or 3DGS with sparse view as input, most of them [5, 8, 16, 21, 26, 40, 47, 48, 63, 73, 75, 81] focus on novel view synthesis rather than geometric surface reconstruction. Recent studies on sparse view surface reconstruction can be divided into two categories: generalization based methods [33, 34, 39, 41, 43, 62] or scene-specific optimization based methods [15, 19, 20, 45, 56, 57, 69, 71]. Generalizable methods require pre-training on large-scale datasets, which is typically time-intensive, and the reconstruction results are often exhibit limited fidelity on unseen scenes. Additionally, reconstruction results can only be plausible when the sparse view configuration aligns with the pre-training setup. When there are significant differences in view angle setup, the reconstruction quality will degrade significantly. In particular, geometric reconstruction cannot be achieved effectively, when there are significant domain discrepancies between the training and testing data.

In contrast, scene-specific optimization methods do not require training on large-scale datasets but instead leverage monocular geometric cues [17, 23, 64] to fit 3D geometry directly from sparse views. Some previous studies [15, 20, 56] introduced prior constraints based on neural implicit reconstruction methods to achieve high-quality sparse view reconstruction, but they were limited by



the slow convergence and heavy computational cost of implicit field optimization. SparseCraft [69] builds upon Instant-NGP [38] by incorporating multi-view stereo (MVS) priors to achieve fast sparse-view reconstruction. However, it is only effective for sparse views with large overlap, and the reconstruction quality degrades significantly when the overlap of views decreases. Recently, FatesGS [19], SolidGS [45] and MAtCha [13] have achieved sparse view reconstruction based on 3DGS [24], enabling fast optimization within minutes. Nevertheless, these methods still face challenges in producing smooth and complete surfaces, as illustrated in Fig. 1.

Although many multi-view reconstruction methods, such as SolidGS [45], GaussianRoom [60], MonoSDF [71], NeuRIS [50] and GausSurf [49], leverage monocular normal priors to achieve higher-quality geometric reconstructions, they merely apply regularization to the rendered normal without fully exploiting the potential of normal priors. On the other hand, some monocular depth estimation methods, such as GeoNet+ [42] and IronDepth [1], refine depth estimation accuracy by utilizing estimated normal. However, due to the distortions and scale ambiguity inherent in monocular depth estimation, these methods cannot effectively leverage normal priors and are not directly applicable for multi-view geometric reconstruction constraints.

### 3 Methods

The pipeline of our method is illustrated in Fig. 2. In this section, we first describe the process of obtaining the depth confidence map and propagating high-confidence depth using the normal prior. Next, we introduce the method of detecting abnormal depth edges and the smoothing loss. Finally, we present the overall loss function.

#### 3.1 Depth Propagation with Normal Prior

The key to improving the quality of sparse view reconstruction is to ensure that 3DGS renders accurate and consistent depth across different views. However, due to the unstructured and discrete nature of Gaussians, relying solely on color reconstruction constraints on the input sparse views is insufficient to accurately distribute most Gaussians on the scene surface and obtain precise depth. A straightforward approach to incorporating additional supervision is to leverage monocular depth priors to constrain the depth rendered by 3DGS for sparse view reconstruction. However, monocular depth priors are often distorted and fail to accurately align with the real scene surface even after calibration.

To overcome the above difficulties, we propose a novel *depth propagation constraint* that indirectly leverages the normal prior to constrain the depth rendered by 3DGS, where the normal prior is a image-based prior predicted by a pretrained normal estimation model. First, we jointly use multi-view geometric consistency and multi-view photometric consistency to predict a depth confidence map. Then, using the normal prior as guidance, we propagate high-confidence depths outward to constrain the regions with low depth confidence.

**Depth Confidence Map** As shown in Fig. 3, given two adjacent input views as the reference view  $I_r$  and the source view  $I_s$ , their depth maps  $D_r$  and  $D_s$  can be obtained by rendering 3D Gaussians. For a particular pixel  $p_r$  on the reference view with coordinates  $(u_r, v_r)$  and its depth  $d_r$  on  $D_r$ , it can be mapped to the source view through the depth-aware reprojection transformation  $H_{rs}$ , resulting in a projected pixel  $p_s$  with coordinates  $(u_s, v_s)$ . Similarly, we can map the pixel  $p_s(u_s, v_s)$  from the source view back to the reference view using the depth  $d_s$  interpolated from  $D_s$  and the depth-aware reprojection transformation  $H_{sr}$ , resulting in a pixel  $p'_r$  with coordinates  $(u'_r, v'_r)$ . The distance error between  $p_r(u_r, v_r)$  and  $p'_r(u'_r, v'_r)$  reflects the consistency of the rendered depths  $d_r$  and  $d_s$ , so we incorporate it as part of the depth confidence. The confidence map of geometric consistency is defined as

$$C_{geo} = \begin{cases} \frac{1}{e^{\|p_r - p'_r\|}}, & \text{if } \|p_r - p'_r\| \leq 1 \\ 0, & \text{if } \|p_r - p'_r\| > 1 \end{cases}, \quad (1)$$

$$p_s = H_{rs}p_r, \quad (2)$$

$$p'_r = H_{sr}p_s. \quad (3)$$

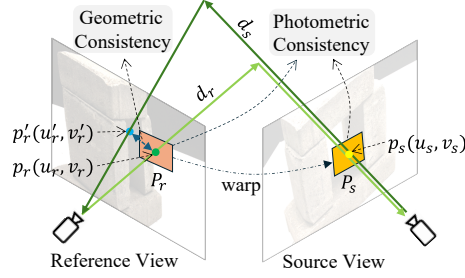
Nevertheless, multi-view geometric consistency cannot fully reflect the accuracy of depth predictions, so we also leverage the patch-based multi-view photometric consistency to further determine the depth confidence. Specifically, as shown in Fig. 3, for a patch  $P_r$  centered at pixel  $p_r$ , we warp it to the source view using the homography matrix  $H_{rs}$ , resulting in a patch  $\hat{P}_s$ . We then employ Normalized Cross-Correlation (NCC) [68] to measure the photometric consistency between patch  $P_r$  and  $P_s$  as another part of the depth confidence as:

$$C_{pho} = NCC(P_r, P_s), \quad (4)$$

where  $P_s = H_{rs}P_r$ . Ultimately, we use the product of multi-view geometric consistency and multi-view photometric consistency to describe the depth confidence as:

$$C = C_{geo} \cdot C_{pho}. \quad (5)$$

It is noteworthy that, unlike PGSR [4], we do not directly use multi-view geometric consistency and multi-view photometric consistency as loss terms to constrain the geometric surface. Instead, we primarily exploits them to estimate depth confidence, enabling the identification of regions with high-fidelity



**Fig. 3:** Illustration of depth confidence calculation.

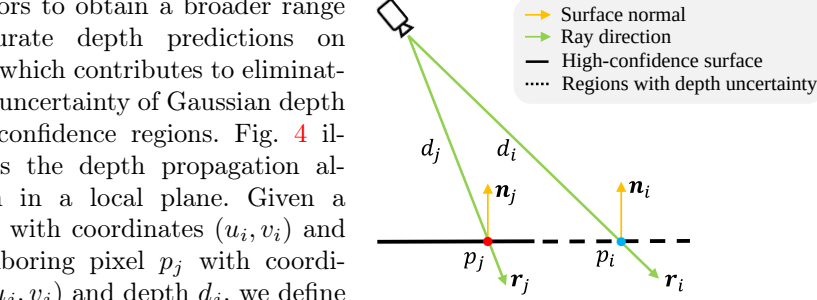
depth. By focusing on high-confidence depth estimation rather than direct multi-view optimization, our method mitigates the limitations of geometric consistency losses in sparse-view settings and leads to more robust reconstruction results.

**Depth Propagation** Our goal of depth propagation is to leverage normal priors to obtain a broader range of accurate depth predictions on planes, which contributes to eliminating the uncertainty of Gaussian depth at low-confidence regions. Fig. 4 illustrates the depth propagation algorithm in a local plane. Given a pixel  $p_i$  with coordinates  $(u_i, v_i)$  and a neighboring pixel  $p_j$  with coordinates  $(u_j, v_j)$  and depth  $d_j$ , we define the normalized ray directions from the camera center passing through these two pixels as  $\mathbf{r}_i$  and  $\mathbf{r}_j$ , respectively. If pixel  $p_i$  and its neighboring pixel  $p_j$  belong to the same plane, meaning their normals are identical, then the depth of pixel  $p_i$  is given by:

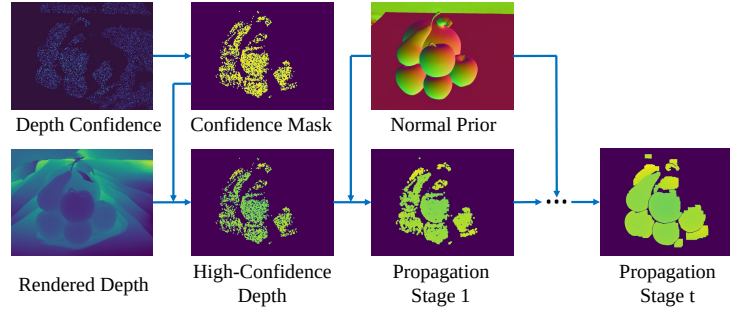
$$d_i = \frac{\mathbf{n}_j \mathbf{r}_j}{\mathbf{n}_i \mathbf{r}_i} d_j. \quad (6)$$

Considering a local neighborhood around pixel  $p_i$ , we propagate the depth values from neighboring pixels to  $p_i$ . We filter the depth values through the depth confidence mask and then do the average to determine the depth of  $p_i$ . This process can be iterated multiple times, as shown in Fig. 5. Therefore, the updated depth map can be represented as,

$$d_j = M_C(u_j, v_j) D, \quad (7)$$



**Fig. 4:** Illustration of the depth propagation.



**Fig. 5:** Illustration of the depth propagation process.

$$M_C = \{C > \tau\}, \quad (8)$$

$$d_i^{t+1} \leftarrow \sum_{j \in N} d_j^t, \quad (9)$$

$$M_C^{t+1} \leftarrow \sum_{j \in N} M_C^t, \quad (10)$$

where  $D$  is the rendered depth,  $M_C$  is the mask of depth confidence and  $\tau$  is the threshold of the mask.  $t$  denotes the iterative update,  $N$  is the number of the nearest neighbors. After obtaining the updated depth map  $d_i^{t+1}$  and mask  $M_C^{t+1}$ , it can be used to supervise the rendered depth, and the loss function is defined below,

$$L_{dp} = M_C^{t+1} \|D - d_i^{t+1}\|. \quad (11)$$

Our method leverages normal priors to extend reliable depth supervision beyond the initially high-confidence regions, providing effective geometric constraints for under-constrained areas such as texture-less surfaces. Specifically, the confidence mask first identifies reliable depth regions as propagation anchors. Depth values are then propagated from these anchors to neighboring regions under the guidance of surface normals, which enforce local geometric consistency during propagation. Meanwhile, the confidence mask is progressively updated to include newly propagated regions, yielding an expanded supervision area. Depth supervision is finally applied within the updated confidence region.

### 3.2 Abnormal Depth Edge-aware Smoothing Loss

The depth propagation method may not extend depth over a large region accurately, due to its reliance on the local planar assumption and the error in the normal prior. In other words, it cannot impose constraints on global depth geometries. In particular, occluded or low-texture regions that are far from high-confidence depth areas cannot be effectively supervised. As a result, under the constraint of normal prior alone, these regions may exhibit generally correct normals but discontinuous depths, leading to abnormal depth edges, as shown in Fig. 2 (b). To address this issue, we propose a loss constraining on abnormal depth edge-aware smoothness.

To extract abnormal depth edges from all depth edges, we first obtain the edge mask of the normal map and the edge mask of the depth, respectively. Then, we multiply the depth edge mask with the inverted edge mask of the normal prior to obtain the mask of abnormal depth edges. Once the edges of the abnormal depth are identified, we apply a smoothing constraint to eliminate these anomalies. Fig. 2 (b) illustrates the specific process. First, we sample at the edge of abnormal depth, obtaining depth  $d_i$ . Then, for each sampled point, we randomly sample another point within its neighborhood, obtaining depth  $d_j$ .

**Table 1:** Quantitative results of Chamfer Distance (CD↓) on DTU dataset with 3 *small-overlapping* images. The methods are divided into three categories, from top to bottom: (1) dense-view reconstruction methods related to ours, (2) the generalizable sparse-view reconstruction methods, and (3) the scene-specific sparse-view reconstruction methods. The best results are in *bold*, the second best are *underlined*. The # indicates that the dense point cloud obtained by MAST3R [10] is used as an initialization.

| Scan ID                       | 24          | 37          | 40          | 55          | 63          | 65          | 69          | 83          | 97          | 105         | 106         | 110         | 114         | 118         | 122         | Mean CD ↓   |
|-------------------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| 2DGS [18]                     | 3.54        | 4.13        | 3.61        | 1.00        | 2.69        | 2.35        | 2.04        | 2.06        | 2.94        | 1.76        | 2.40        | 2.97        | 1.35        | 2.17        | 1.69        | 2.45        |
| PGSR [4]                      | 4.01        | 5.19        | 3.65        | 0.93        | 2.96        | 2.84        | 1.62        | 2.16        | 3.24        | 1.42        | 2.35        | 1.91        | 0.57        | 1.55        | 1.26        | 2.38        |
| SparseNeus <sub>ft</sub> [34] | 4.81        | 5.56        | 5.81        | 2.68        | 3.30        | 3.88        | 2.39        | 2.91        | 3.08        | 2.33        | 2.64        | 3.12        | 1.74        | 3.55        | 2.31        | 3.34        |
| VolRecon [43]                 | 3.05        | 4.45        | 3.36        | 3.09        | 2.78        | 3.68        | 3.01        | 2.87        | 3.07        | 2.55        | 3.07        | 2.77        | 1.59        | 3.44        | 2.51        | 3.02        |
| GenS <sub>ft</sub> [41]       | 7.67        | 8.58        | 5.41        | 5.88        | 7.18        | 5.26        | 4.40        | 6.29        | 6.19        | 4.08        | 6.04        | 5.29        | 4.69        | 4.35        | 4.01        | 5.69        |
| ReTR [33]                     | 3.30        | 3.60        | 3.37        | 2.85        | 2.91        | 3.07        | 2.26        | 2.29        | 2.25        | 2.05        | 2.80        | 2.80        | 1.52        | 2.29        | 2.11        | 2.63        |
| UFORecon [39]                 | 1.51        | <u>2.58</u> | 1.82        | 1.44        | 1.60        | 1.81        | 1.04        | 1.56        | 0.96        | 1.40        | 1.20        | 0.93        | 0.66        | 1.26        | 1.26        | 1.40        |
| MonoSDF [71]                  | 3.47        | 3.61        | 2.10        | 1.05        | 2.37        | 1.38        | 1.41        | 1.85        | 1.74        | 1.10        | 1.46        | 2.28        | 1.25        | 1.44        | 1.45        | 1.86        |
| NeuSurf [20]                  | 1.35        | 3.25        | 2.50        | 0.80        | 1.21        | 2.35        | 0.77        | 1.19        | 1.20        | 1.05        | 1.05        | 1.21        | <b>0.41</b> | 0.80        | 1.08        | 1.35        |
| SparseCraft [69]              | 2.42        | 2.79        | 2.78        | 0.74        | 1.44        | 2.51        | 1.26        | 1.42        | 1.65        | 1.10        | 1.34        | 5.24        | 0.65        | 0.88        | 1.16        | 1.83        |
| SparseRecon [15]              | 1.26        | <b>1.46</b> | <u>1.39</u> | <b>0.71</b> | 1.20        | 2.38        | <u>0.70</u> | 1.23        | 0.92        | 0.80        | <u>0.94</u> | 0.77        | <u>0.44</u> | 0.83        | 0.91        | 1.06        |
| FatesGS [19]                  | 1.32        | 2.85        | 2.71        | 0.80        | 1.44        | 2.08        | 1.11        | 1.19        | 1.33        | <u>0.76</u> | 1.49        | 0.85        | 0.47        | 1.05        | 1.06        | 1.37        |
| Sparse2DGS [57]               | 3.13        | 5.82        | 2.98        | 1.16        | 4.38        | 2.35        | 1.32        | 4.26        | 1.66        | 1.09        | 1.78        | 1.89        | 0.66        | 1.29        | 1.22        | 2.33        |
| MAtCha [13]                   | 1.38        | 2.95        | 1.70        | 1.01        | 1.43        | 1.58        | 1.15        | 1.65        | 1.81        | 1.35        | 1.39        | 1.24        | 0.89        | 1.47        | 1.20        | 1.48        |
| MAtCha <sup>#</sup> [13]      | <u>0.88</u> | 2.79        | 1.40        | 0.83        | 0.93        | 1.35        | 0.87        | 1.25        | 1.51        | 0.95        | 1.15        | 1.02        | 0.56        | 1.01        | 0.92        | 1.16        |
| Ours                          | 1.45        | 2.21        | 2.04        | <u>0.72</u> | <u>0.85</u> | <b>1.09</b> | <b>0.64</b> | <u>1.18</u> | <u>0.76</u> | 0.77        | 0.98        | <b>0.61</b> | 0.46        | <u>0.79</u> | <u>0.77</u> | <u>1.02</u> |
| Ours <sup>#</sup>             | <b>0.57</b> | <u>1.71</u> | <b>1.37</b> | <b>0.71</b> | <b>0.84</b> | <u>1.11</u> | <b>0.64</b> | <b>1.04</b> | <b>0.70</b> | <b>0.74</b> | <b>0.76</b> | <u>0.65</u> | 0.45        | <b>0.66</b> | <b>0.76</b> | <b>0.85</b> |

Finally, we back-project these sampled points into 3D space using camera parameters and their depths, resulting in 3D points  $a_i$  and  $a_j$ , and form a vector  $\mathbf{v}_{ij}$ . According to the local planar assumption, the vector  $\mathbf{v}_{ij}$  should be perpendicular to the normal  $\mathbf{n}_i$  at the sampled point  $i$ . Therefore, the smoothing loss is defined as

$$L_{ds} = \frac{1}{N} \sum_{i,j \in N} \mathbf{n}_i \mathbf{v}_{ij}, \quad (12)$$

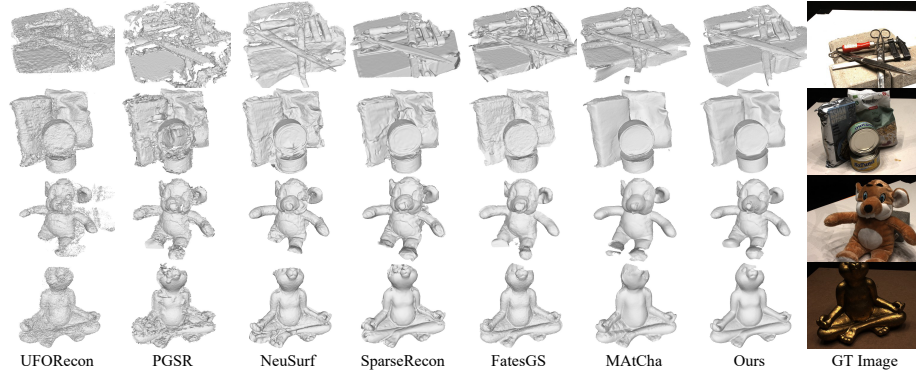
$$\mathbf{v}_{ij} = a_j - a_i. \quad (13)$$

### 3.3 Training Loss

Our loss function consists of depth propagation constraint loss  $L_{dp}$ , abnormal depth edge smoothing loss  $L_{ds}$ , normal prior loss  $L_{normal}$ , and the commonly-used rendering error  $L_{rgb}$ . We define the overall loss function as,

$$L = L_{rgb} + \lambda_1 L_{dp} + \lambda_2 L_{ds} + \lambda_3 L_{normal}. \quad (14)$$

where  $\lambda_1$ ,  $\lambda_2$ , and  $\lambda_3$  are balance weights, the normal prior loss  $L_{normal}$  is computed as the  $\ell_1$  distance between the rendered normal and the normal prior.



**Fig. 6:** Visual comparison of reconstruction with 3 small-overlapping views on the DTU dataset.

## 4 Experiments

### 4.1 Datasets

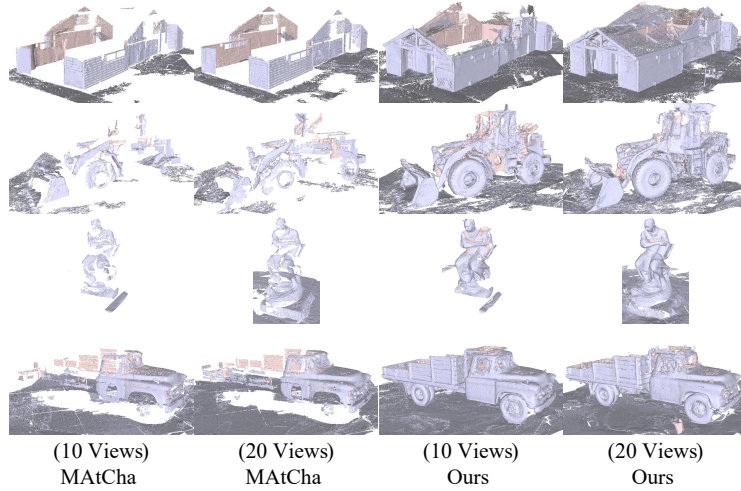
We evaluate the performance of our method on three datasets: DTU [22], Tanks and Temples (TNT) [25] and 5 scenes captured by ourselves. We follow previous methods [15, 19, 20, 71], select 15 scenes from DTU and use 3 small-overlapping views as input. The results on views with large overlap are provided in the supplementary material. On the TNT dataset, we follow MAtCha [13] and use four scenes, excluding the *Courthouse* and *Meetingroom* due to their large scale, which is not suitable for sparse-view reconstruction. We uniformly sample 5, 10, or 20 views from each scene as input. We also capture 5 real-world scenes using a consumer camera. Similarly to the DTU dataset, only 3 images are used in each scene.

### 4.2 Implementation Details

We implement our method based on the 3DGS [24] framework and use the modified differentiable Gaussian renderer from PGSR [4]. The Gaussian opacity decay strategy [16] is also used to reduce redundant Gaussians near the scene surface. We train our model on an RTX3090 GPU, the training iterations are set to 10K. We adopt Metric3Dv2 [17] to generate the normal prior. The confidence threshold  $\tau$  in Eq. (8) is set to 0.6. The iterations  $t$  for depth propagation is set to 10. The hyper-parameters in Eq. (14) are set as  $\lambda_1 = 0.5$ ,  $\lambda_2 = 0.03$ ,  $\lambda_3 = 0.1$ .

### 4.3 Baselines and Metrics

We compare our approach with the latest sparse-view reconstruction methods including the generalizable methods, such as SparseNeuS [34], VolRecon [43], GenS [41], ReTR [33] and UFORcon [39], and the scene-specific optimization



**Fig. 7:** Visual comparison of reconstruction with sparse views on the TNT dataset.

methods, such as MonoSDF [71], NeuSurf [20], SparseCraft [69], SparseRecon [15], MAtCha [13] and FatesGS [19]. We also compare our approach with the latest dense-view reconstruction methods based on 3DGS, such as 2DGS [18] and PGSR [4]. To evaluate the reconstruction quality, we report the Chamfer Distance (CD) on the DTU dataset and F1 score on TNT dataset.

#### 4.4 Comparisons

*Sparse View Reconstruction on DTU.* Tab. 1 presents the CD results on the DTU dataset using 3 small-overlapping views. For a fair comparison, we also evaluate the performance of MAtCha [13] using COLMAP [44] point clouds for initialization. Quantitatively, our method outperforms all baseline approaches regardless of the initialization used. Fig. 6 provides a visual comparison of the reconstruction results. The latest generalizable methods UFORecon [39] produces overly coarse surfaces. PGSR [4], the state-of-the-art dense-view reconstruction method based on 3DGS [24], also struggles to achieve fine details under sparse-view settings. The latest scene-specific optimization methods, NeuSurf [20], SparseRecon [15], FatesGS [19] and MAtCha [13], successfully reconstruct complete geometric surfaces but exhibit noticeable shortcomings. Among all methods, our approach achieves smoother and more accurate surfaces, significantly outperforming the others.

*Sparse View Reconstruction on TNT.* We compare several latest scene-specific optimized 3DGS-based sparse-view reconstruction methods on 360-degree surround-view scenes from the TNT [25] dataset. The evaluated methods include FatesGS [19], MAtCha [13] and ours. Tab. 2 show quantitative comparisons on the TNT



**Table 2:** Quantitative results of F1 score on TNT dataset. The dense point cloud obtained by MAST3R is used as initialization.

| Methods      | 5 Views      | 10 Views     | 20 views     |
|--------------|--------------|--------------|--------------|
| FatesGS [19] | 0.014        | 0.025        | 0.033        |
| MAtCha [13]  | 0.072        | 0.156        | 0.218        |
| Ours         | <b>0.081</b> | <b>0.173</b> | <b>0.293</b> |

**Table 3:** Efficiency comparison of sparse view construction methods. The GPU memory usage is recorded during training.

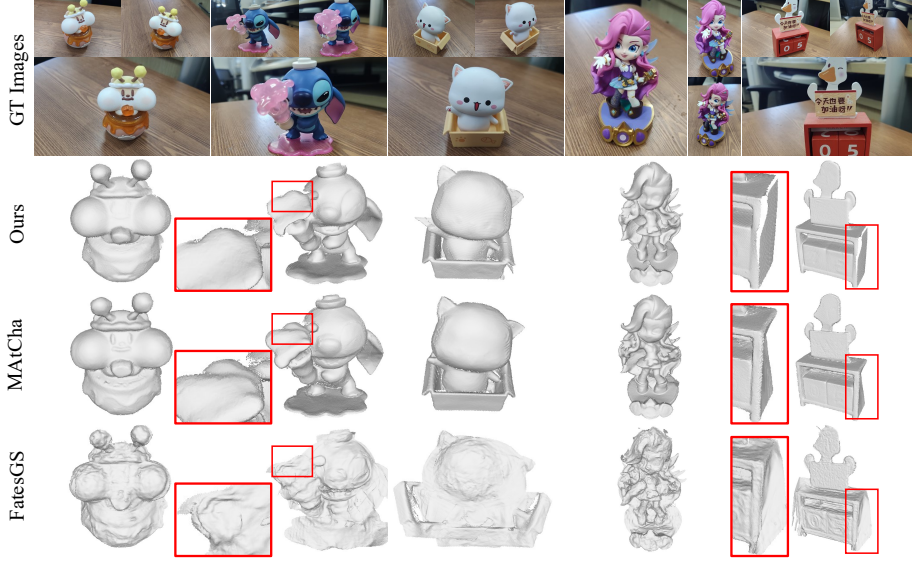
| Methods          | GPU Mem. Usage | Training Time  |
|------------------|----------------|----------------|
| UFORcon [39]     | 23 GB          | 10 days        |
| SparseRecon [15] | 8 GB           | 3.5 hours      |
| FatesGS [19]     | 4 GB           | 14 mins        |
| MAtCha [13]      | <b>3 GB</b>    | <b>10 mins</b> |
| Ours             | <b>3 GB</b>    | <b>10 mins</b> |

dataset with 5, 10 or 20 views. FatesGS fails to reconstruct in almost all scenes and under various input conditions. Our method outperforms all 3DGS-based sparse view reconstruction methods. Fig. 7 provides a visual comparison of reconstruction results with 10 or 20 views. MAtCha exhibits suboptimal performance and struggles to recover meaningful meshes. In contrast, our method achieves more complete and detailed surfaces.

*Sparse View Reconstruction on Self-Captured Data.* On our self-captured dataset, we compare our method with FatesGS [19] and MAtCha [13], as these two methods, specifically designed for scene-specific sparse-view reconstruction, are more comparable to ours in terms of both reconstruction quality and efficiency. As shown in Figure 8, the results of MAtCha exhibit some shortcomings, while the results of FatesGS show noticeable distortions and roughness. In general, our method achieves the best reconstruction performance.

*Efficiency Comparison.* Although generalizable methods can produce reconstructions in minutes, they require extensive pre-training, often spanning several days, and demand more GPU memory. Scene-specific methods based on NeRF involve time-consuming optimization. In contrast, 3DGS-based approaches have demonstrated much higher training efficiency. As shown in Tab. 3, our method achieves comparable memory usage and training time to other 3DGS-based optimization frameworks. Notably, our approach attains superior reconstruction quality under comparable computational efficiency.

#### 4.5 Ablation Study



**Fig. 8:** Visual comparison on self-captured dataset with 3 input views.

To validate the effectiveness of the depth propagation constraint loss  $L_{dp}$  and the anomalous depth edge smoothing loss  $L_{ds}$ , we conduct an ablation study on the DTU [22] dataset and report the average Chamfer Distance (CD) across 15 scenes. As shown in Tab. 4, removing any of our proposed optimization losses results in performance degradation, showing the effectiveness of each component. Since the anomalous depth edges only appear in a small subset of scenes in the DTU dataset, the performance improvement from the anomalous depth edge smoothing loss is relatively limited.

**Table 4:** Ablation studies with 3 small-overlapping views on DTU dataset. O.D. stands for the opacity decay strategy.

| $L_{normal}$ | O.D. | $L_{dp}$ | $L_{ds}$ | Mean CD↓    |
|--------------|------|----------|----------|-------------|
|              |      |          |          | 2.79        |
| ✓            |      |          |          | 2.28        |
| ✓            | ✓    |          |          | 1.55        |
| ✓            | ✓    | ✓        |          | 1.13        |
| ✓            | ✓    |          | ✓        | 1.38        |
| ✓            | ✓    | ✓        | ✓        | <b>1.02</b> |

Fig. 9 presents the visualization of ablation results on scan 24 of the DTU dataset. We use a model that incorporates only the rendering loss as the baseline. Starting from this baseline, we progressively add the normal loss  $L_{normal}$  and the opacity decay strategy [16]. The first row of Fig. 9 shows the corresponding reconstruction results. As observed, the loss  $L_{normal}$  smooths the reconstructed surface, but distortions remain. When the opacity decay strategy is further introduced, these distortions are alleviated to some extent, but noticeable shortcomings persist.

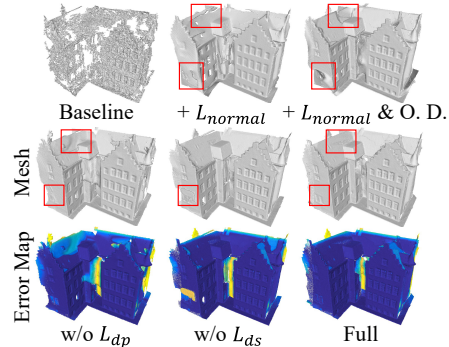
To verify the contribution of either the depth propagation constraint loss  $L_{dp}$  or the depth edge smoothing loss  $L_{ds}$ , we add each of them to the normal loss and the opacity decay strategy. The second and third rows in Fig. 9 illustrate the corresponding reconstruction results and error maps respectively. It can be seen that without  $L_{ds}$ , the surface is not complete, whereas omitting  $L_{dp}$  leads to slight deformations. These deformed regions, characterized by minimal depth variations, are difficult to detect through depth edge, making  $L_{ds}$  ineffective in addressing them. However, the depth propagation constraint loss  $L_{dp}$  effectively eliminates these distortions.

Fig. 10 provides a clearer illustration of the depth anomalies, which lead to mesh fragmentation. The comparison shows that  $L_{ds}$  significantly improves the quality of the reconstructed geometry.

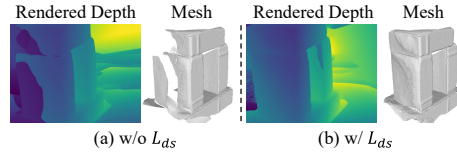
To investigate the impact of the iterations  $t$  of depth propagation on reconstruction quality, we conduct an ablation on the DTU dataset. As shown in Tab. 5, when the iterations  $t$  increases from 4 to 10, the CD consistently decreases, indicating a steady improvement in reconstruction accuracy. However, further increasing the iterations beyond 10 does not yield additional gains and even leads to slight fluctuations in CD values. Therefore, we set the number of propagation iterations to 10 in our implementation. Notably, even when  $t$  is increased to 20, the total computation time remains as low as 0.008 seconds, having almost no impact on the overall performance of our method.

**Table 5:** Effect of propagation iterations on Chamfer Distance (CD) on the DTU dataset.

| $t$  | 4     | 6     | 8     | 10    | 12    | 14    | 16    | 18    | 20    |
|------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| CD ↓ | 1.126 | 1.069 | 1.052 | 1.021 | 1.063 | 1.027 | 1.019 | 1.056 | 1.032 |



**Fig. 9:** Visualization of ablation results on *Scan 24* in the DTU dataset. O.D. stands for the opacity decay strategy.



**Fig. 10:** Effect of the abnormal depth edge-aware smoothing loss.

To evaluate the robustness of our method to the quality of normal priors, we conducted an ablation study on the DTU dataset using three different normal estimators: Omnidata [11], StableNormal [67] and Metric3Dv2 [17]. As shown in Tab. 6, higher-quality normals leads to lower CD. However, it is worth noting that our method still achieves reconstruction quality comparable to the recent MAtCha [13] method even when using the relatively low-quality normals from Omnidata [11]. This demonstrates that our framework is robust to the accuracy of the normal prior.

**Table 6:** Effect of different normal priors on reconstruction quality on the DTU dataset.

| Normal Prior | Omnidata [11] | Metric3Dv2 [17] | StableNormal [67] |
|--------------|---------------|-----------------|-------------------|
| CD ↓         | 1.14          | 1.02            | 1.05              |

## 5 Conclusion

In this paper, we propose a novel 3DGS-based surface reconstruction method, achieving fast and high-fidelity geometric reconstruction from sparse views. We leverage normal priors to guide the propagation of high-confidence depth regions, thereby constraining areas with lower depth confidence. Additionally, we perform anomalous depth edge detection to address depth discontinuities. These constraints ensure the accuracy of rendered depth under sparse input views, ultimately leading to high-quality scene geometry. Extensive experiments on DTU, TNT, and self-captured datasets demonstrate that our method outperforms state-of-the-art sparse view surface reconstruction methods.

*Limitation.* In our method, the depth confidence is estimated based on multi-view geometric and photometric consistency. However, these consistency cues can become unreliable in texture-less regions, leading to inaccurate confidence estimation. As a result, erroneous depths may be propagated to neighboring areas, resulting in local geometric distortions or artifacts in the reconstructed surface.

## Acknowledgements

The corresponding authors are Yu-Shen Liu and Junsheng Zhou. This work was partially supported by Deep Earth Probe and Mineral Resources Exploration – National Science and Technology Major Project (2024ZD1003405), and the National Natural Science Foundation of China (62272263).

## References

1. Bae, G., Budvytis, I., Cipolla, R.: Irondepth: Iterative refinement of single-view depth using surface normal and its uncertainty. In: British Machine Vision Conference (BMVC) (2022) [5](#)
2. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5470–5479 (2022) [4](#)
3. Cai, B., Huang, J., Jia, R., Lv, C., Fu, H.: NeuDA: Neural deformable anchor for high-fidelity implicit surface reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8476–8485 (2023) [4](#)
4. Chen, D., Li, H., Ye, W., Wang, Y., Xie, W., Zhai, S., Wang, N., Liu, H., Bao, H., Zhang, G.: PGSR: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics* **31**, 1–12 (01 2024) [1](#), [4](#), [6](#), [9](#), [10](#), [11](#)
5. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: MVSplat: Efficient 3D gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision. pp. 370–386. Springer (2024) [2](#), [4](#)
6. Dai, P., Xu, J., Xie, W., Liu, X., Wang, H., Xu, W.: High-quality surface reconstruction using gaussian surfels. In: ACM SIGGRAPH 2024 Conference Proceedings (2024) [1](#), [4](#)
7. Darmon, F., Bascle, B., Devaux, J.C., Monasse, P., Aubry, M.: Improving neural implicit surfaces geometry with patch warping. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6260–6269 (2022) [1](#), [4](#)
8. Deng, K., Liu, A., Zhu, J.Y., Ramanan, D.: Depth-supervised NeRF: Fewer views and faster training for free. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12882–12891 (2022) [4](#)
9. Dogaru, A., Ardelean, A.T., Ignatyev, S., Zakharov, E., Burnaev, E.: Sphere-guided training of neural implicit surfaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20844–20853 (2023) [4](#)
10. Duisterhof, B.P., Züst, L., Weinzaepfel, P., Leroy, V., Cabon, Y., Revaud, J.: Mast3R-SfM: a fully-integrated solution for unconstrained structure-from-motion. In: 2025 International Conference on 3D Vision (3DV). pp. 1–10. IEEE (2025) [9](#)
11. Eftekhar, A., Sax, A., Malik, J., Zamir, A.: Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3D scans. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10786–10796 (2021) [15](#)
12. Fu, Q., Xu, Q., Ong, Y.S., Tao, W.: Geo-NeuS: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *Advances in Neural Information Processing Systems* **35**, 3403–3416 (2022) [1](#), [4](#)
13. Guédon, A., Ichikawa, T., Yamashita, K., Nishino, K.: MAtCha Gaussians: Atlas of Charts for High-Quality Geometry and Photorealism From Sparse Views. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6001–6011 (2025) [2](#), [5](#), [9](#), [10](#), [11](#), [12](#), [15](#)
14. Guédon, A., Lepetit, V.: SuGaR: Surface-aligned gaussian splatting for efficient 3D mesh reconstruction and high-quality mesh rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5354–5363 (2024) [1](#), [4](#)

15. Han, L., Zhang, X., Song, H., Shi, K., Liu, Y.S., Han, Z.: SparseRecon: Neural implicit surface reconstruction from sparse views with feature and depth consistencies. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 28514–28524 (2025) [2](#), [4](#), [9](#), [10](#), [11](#), [12](#)
16. Han, L., Zhou, J., Liu, Y.S., Han, Z.: Binocular-guided 3D gaussian splatting with view consistency for sparse view synthesis. In: *Advances in Neural Information Processing Systems* (2024) [2](#), [4](#), [10](#), [13](#)
17. Hu, M., Yin, W., Zhang, C., Cai, Z., Long, X., Chen, H., Wang, K., Yu, G., Shen, C., Shen, S.: Metric3Dv2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024) [4](#), [10](#), [15](#)
18. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2D gaussian splatting for geometrically accurate radiance fields. In: *ACM SIGGRAPH 2024 Conference Proceedings* (2024) [1](#), [4](#), [9](#), [11](#)
19. Huang, H., Wu, Y., Deng, C., Gao, G., Gu, M., Liu, Y.S.: FatesGS: Fast and accurate sparse-view surface reconstruction using gaussian splatting with depth-feature consistency. In: *Proceedings of the AAAI Conference on Artificial Intelligence* (2025) [2](#), [4](#), [5](#), [9](#), [10](#), [11](#), [12](#)
20. Huang, H., Wu, Y., Zhou, J., Gao, G., Gu, M., Liu, Y.S.: NeuSurf: On-surface priors for neural surface reconstruction from sparse input views. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. vol. 38, pp. 2312–2320 (2024) [2](#), [4](#), [9](#), [10](#), [11](#)
21. Jain, A., Tancik, M., Abbeel, P.: Putting NeRF on a Diet: Semantically consistent few-shot view synthesis implementation. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 5885–5894 (2021) [4](#)
22. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanæs, H.: Large scale multi-view stereopsis evaluation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 406–413 (2014) [10](#), [13](#)
23. Ke, B., Obukhov, A., Huang, S., Metzger, N., Daut, R.C., Schindler, K.: Repurposing diffusion-based image generators for monocular depth estimation. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 9492–9502 (2024) [4](#)
24. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics* **42**(4), 1–14 (2023) [1](#), [2](#), [4](#), [5](#), [10](#), [11](#)
25. Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V.: Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)* **36**(4), 1–13 (2017) [10](#), [11](#)
26. Li, J., Zhang, J., Bai, X., Zheng, J., Ning, X., Zhou, J., Gu, L.: DNGaussian: Optimizing sparse-view 3D gaussian radiance fields with global-local depth normalization. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20775–20785 (2024) [2](#), [4](#)
27. Li, Q., Feng, H., Gong, X., Liu, Y.S.: Learning normals of noisy points by local gradient-aware surface filtering. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 28828–28838 (2025) [4](#)
28. Li, Q., Feng, H., Gong, X., Liu, Y.S.: VA-GS: Enhancing the geometric representation of gaussian splatting via view alignment. *Advances in Neural Information Processing Systems* **38**, 173867–173886 (2026) [4](#)
29. Li, Q., Feng, H., Shi, K., Gao, Y., Fang, Y., Liu, Y.S., Han, Z.: PFF-Net: Patch feature fitting for point cloud normal estimation. *IEEE Transactions on Visualization and Computer Graphics* (2025) [4](#)

30. Li, S., Liu, Y.S., Han, Z.: GaussianUDF: Inferring unsigned distance functions through 3D gaussian splatting. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 27113–27123 (2025) [4](#)
31. Li, X., Tan, X., Zhang, Z., Xie, Y., Ma, L.: Point mask transformer for outdoor point cloud semantic segmentation. *Computational Visual Media* (2025) [4](#)
32. Li, Z., Müller, T., Evans, A., Taylor, R.H., Unberath, M., Liu, M.Y., Lin, C.H.: Neuralangelo: High-fidelity neural surface reconstruction. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 8456–8465 (2023) [1](#), [4](#)
33. Liang, Y., He, H., Chen, Y.: ReTR: Modeling rendering via transformer for generalizable neural surface reconstruction. *Advances in Neural Information Processing Systems* **36** (2024) [2](#), [4](#), [9](#), [10](#)
34. Long, X., Lin, C., Wang, P., Komura, T., Wang, W.: SparseNeuS: Fast generalizable neural surface reconstruction from sparse views. In: *European Conference on Computer Vision*. pp. 210–227. Springer (2022) [2](#), [4](#), [9](#), [10](#)
35. Lyu, X., Sun, Y.T., Huang, Y.H., Wu, X., Yang, Z., Chen, Y., Pang, J., Qi, X.: 3DGSr: Implicit surface reconstruction with 3D gaussian splatting. *ACM Transactions on Graphics (TOG)* **43**(6), 1–12 (2024) [1](#), [4](#)
36. Mao, X., Yuan, H., Lu, X., Hamzaoui, R., Gao, W.: PCAC-GAN: a sparse-tensor-based generative adversarial network for 3D point cloud attribute compression. *Computational Visual Media* (2025) [4](#)
37. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* **65**(1), 99–106 (2021) [1](#), [2](#), [4](#)
38. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)* **41**(4), 1–15 (2022) [4](#), [5](#)
39. Na, Y., Kim, W.J., Han, K.B., Ha, S., Yoon, S.E.: UFORecon: Generalizable sparse-view surface reconstruction from arbitrary and unfavorable sets. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5094–5104 (2024) [4](#), [9](#), [10](#), [11](#), [12](#)
40. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N.: RegNeRF: Regularizing neural radiance fields for view synthesis from sparse inputs. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5480–5490 (2022) [4](#)
41. Peng, R., Gu, X., Tang, L., Shen, S., Yu, F., Wang, R.: GenS: Generalizable neural surface reconstruction from multi-view images. In: *Advances in Neural Information Processing Systems*. vol. 36, pp. 56932–56945 (2023) [2](#), [4](#), [9](#), [10](#)
42. Qi, X., Liu, Z., Liao, R., Torr, P.H., Urtasun, R., Jia, J.: GeoNet++: Iterative geometric neural network with edge-aware refinement for joint depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **44**(2), 969–984 (2020) [5](#)
43. Ren, Y., Zhang, T., Pollefeys, M., Süssstrunk, S., Wang, F.: 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*. pp. 16685–16695 (2023) [2](#), [4](#), [9](#), [10](#)
44. Schönberger, J.L., Zheng, E., Pollefeys, M., Frahm, J.M.: Pixelwise view selection for unstructured multi-view stereo. In: *European Conference on Computer Vision* (2016) [11](#)



45. Shen, Z., Liu, Y., Chen, Z., Li, Z., Wang, J., Liang, Y., Yu, Z., Zhang, J., Xu, Y., Schaefer, S., et al.: SolidGS: Consolidating gaussian surfel splatting for sparse-view surface reconstruction. *arXiv preprint arXiv:2412.15400* (2024) [3](#), [4](#), [5](#)
46. Sun, J., Chen, X., Wang, Q., Li, Z., Averbuch-Elor, H., Zhou, X., Snavely, N.: Neural 3D reconstruction in the wild. In: *ACM SIGGRAPH 2022 Conference Proceedings*. pp. 1–9 (2022) [4](#)
47. Truong, P., Rakotosaona, M.J., Manhardt, F., Tombari, F.: SPARF: Neural radiance fields from sparse and noisy poses. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 4190–4200 (2023) [4](#)
48. Wang, G., Chen, Z., Loy, C.C., Liu, Z.: SparseNeRF: Distilling depth ranking for few-shot novel view synthesis. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 9065–9076 (2023) [4](#)
49. Wang, J., Liu, Y., Wang, P., Lin, C., Hou, J., Li, X., Komura, T., Wang, W.: GausSurf: Geometry-guided 3D gaussian splatting for surface reconstruction. *arXiv preprint arXiv:2411.19454* (2024) [3](#), [5](#)
50. Wang, J., Wang, P., Long, X., Theobalt, C., Komura, T., Liu, L., Wang, W.: NeuRIS: Neural reconstruction of indoor scenes using normal priors. In: *European conference on computer vision*. pp. 139–155. Springer (2022) [5](#)
51. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems* (2021) [1](#), [4](#)
52. Wang, W., Liu, X., Zhou, H., Wei, L., Deng, Z., Murshed, M., Lu, X.: Noise4Denoise: Leveraging noise for unsupervised point cloud denoising. *Computational Visual Media* **10**(4), 659–669 (2024) [4](#)
53. Wang, Y., Han, Q., Habermann, M., Daniilidis, K., Theobalt, C., Liu, L.: NeuS2: Fast learning of neural implicit surfaces for multi-view reconstruction. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 3295–3306 (2023) [1](#), [4](#)
54. Wang, Y., Skorokhodov, I., Wonka, P.: HF-NeuS: Improved surface reconstruction using high-frequency details. *Advances in Neural Information Processing Systems* **35**, 1966–1978 (2022) [1](#), [4](#)
55. Wen, X., Xiang, P., Han, Z., Cao, Y.P., Wan, P., Zheng, W., Liu, Y.S.: PMP-Net++: Point cloud completion by transformer-enhanced multi-step point moving paths. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(1), 852–867 (2022) [4](#)
56. Wu, H., Graikos, A., Samaras, D.: S-VolSDF: Sparse multi-view stereo regularization of neural implicit surfaces. *International Conference on Computer Vision* (2023) [2](#), [4](#)
57. Wu, J., Li, R., Zhu, Y., Guo, R., Sun, J., Zhang, Y.: Sparse2DGS: Geometry-prioritized gaussian splatting for surface reconstruction from sparse views. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 11307–11316 (2025) [2](#), [4](#), [9](#)
58. Wu, T., Wang, J., Pan, X., Xu, X., Theobalt, C., Liu, Z., Lin, D.: Voxurf: Voxel-based efficient and accurate neural surface reconstruction. *International Conference on Learning Representations* (2022) [1](#), [4](#)
59. Wu, T., Yuan, Y.J., Zhang, L.X., Yang, J., Cao, Y.P., Yan, L.Q., Gao, L.: Recent advances in 3D gaussian splatting. *Computational Visual Media* **10**(4), 613–642 (2024) [1](#)
60. Xiang, H., Li, X., Cheng, K., Lai, X., Zhang, W., Liao, Z., Zeng, L., Liu, X.: GaussianRoom: Improving 3D gaussian splatting with SDF guidance and monocular

- cues for indoor scene reconstruction. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 2686–2693. IEEE (2025) [5](#)
61. Xiang, P., Wen, X., Liu, Y.S., Cao, Y.P., Wan, P., Zheng, W., Han, Z.: Snowflake point deconvolution for point cloud completion and generation with skip-transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(5), 6320–6338 (2022) [4](#)
  62. Xu, L., Guan, T., Wang, Y., Liu, W., Zeng, Z., Wang, J., Yang, W.: C2F2NeuS: Cascade cost frustum fusion for high fidelity and generalizable neural surface reconstruction. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 18291–18301 (2023) [2](#), [4](#)
  63. Yang, J., Pavone, M., Wang, Y.: FreeNeRF: Improving few-shot neural rendering with free frequency regularization. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 8254–8263 (2023) [4](#)
  64. Yang, L., Kang, B., Huang, Z., Xu, X., Feng, J., Zhao, H.: Depth Anything: Unleashing the power of large-scale unlabeled data. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 10371–10381 (2024) [4](#)
  65. Yang, Y.Q., Guo, Y.X., Xiong, J.Y., Liu, Y., Pan, H., Wang, P.S., Tong, X., Guo, B.: Swin3D: A pretrained transformer backbone for 3D indoor scene understanding. *Computational Visual Media* **11**(1), 83–101 (2025) [4](#)
  66. Yariv, L., Gu, J., Kasten, Y., Lipman, Y.: Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems* **34**, 4805–4815 (2021) [1](#), [4](#)
  67. Ye, C., Qiu, L., Gu, X., Zuo, Q., Wu, Y., Dong, Z., Bo, L., Xiu, Y., Han, X.: StableNormal: Reducing diffusion variance for stable and sharp normal. *ACM Transactions on Graphics (TOG)* **43**(6), 1–18 (2024) [15](#)
  68. Yoo, J.C., Han, T.H.: Fast normalized cross-correlation. *Circuits, systems and signal processing* **28**, 819–843 (2009) [6](#)
  69. Younes, M., Ouasfi, A., Boukhayma, A.: SparseCraft: Few-shot neural reconstruction through stereopsis guided geometric linearization. In: *European Conference on Computer Vision*. pp. 37–56. Springer (2024) [2](#), [4](#), [5](#), [9](#), [11](#)
  70. Yu, M., Lu, T., Xu, L., Jiang, L., Xiangli, Y., Dai, B.: GSDF: 3DGS meets SDF for improved neural rendering and reconstruction. *Advances in Neural Information Processing Systems* **37**, 129507–129530 (2024) [4](#)
  71. Yu, Z., Peng, S., Niemeyer, M., Sattler, T., Geiger, A.: MonoSDF: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in Neural Information Processing Systems* **35**, 25018–25032 (2022) [1](#), [2](#), [4](#), [5](#), [9](#), [10](#), [11](#)
  72. Yu, Z., Sattler, T., Geiger, A.: Gaussian Opacity Fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics* **43**(6), 1–13 (2024) [1](#)
  73. Yuan, Y.J., Lai, Y.K., Huang, Y.H., Kobbelt, L., Gao, L.: Neural radiance fields from sparse rgb-d images for high-quality view synthesis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**(7), 8713–8728 (2022) [4](#)
  74. Zhang, B., Fang, C., Shrestha, R., Liang, Y., Long, X.X., Tan, P.: RaDe-GS: Rasterizing depth in gaussian splatting. *ACM Transactions on Graphics* **45**(2), 1–14 (2026) [4](#)
  75. Zhang, J., Li, J., Yu, X., Huang, L., Gu, L., Zheng, J., Bai, X.: CoR-GS: sparse-view 3D gaussian splatting via co-regularization. In: *European Conference on Computer Vision*. pp. 335–352. Springer (2024) [2](#), [4](#)

76. Zhang, W., Jia, E.Y.t., Zhou, J., Ma, B., Shi, K., Liu, Y.S., Han, Z.: NeRFPrior: Learning neural radiance field as a prior for indoor scene reconstruction. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 11317–11327 (2025) [4](#)
77. Zhang, W., Liu, Y.S., Han, Z.: Neural signed distance function inference through splatting 3D gaussians pulled on zero-level set. *Advances in Neural Information Processing Systems* **37**, 101856–101879 (2025) [1](#), [4](#)
78. Zhang, W., Wang, C., Shi, K., Liu, Y.S., Han, Z.: VRP-UDF: Towards unbiased learning of unsigned distance functions from multi-view images with volume rendering priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2026) [4](#)
79. Zhang, W., Xing, R., Zeng, Y., Liu, Y.S., Shi, K., Han, Z.: Fast learning radiance fields by shooting much fewer rays. *IEEE Transactions on Image Processing* **32**, 2703–2718 (2023) [4](#)
80. Zhang, W., Yang, Y., Huang, H., Han, L., Shi, K., Liu, Y.S., Han, Z.: MonoInstance: Enhancing monocular priors via multi-view instance alignment for neural rendering and reconstruction. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21642–21653 (2025) [4](#)
81. Zhu, Z., Fan, Z., Jiang, Y., Wang, Z.: FSGS: Real-time few-shot view synthesis using gaussian splatting. In: European Conference on Computer Vision. pp. 145–163. Springer (2025) [2](#), [4](#)