This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Sparse2DGS: Geometry-Prioritized Gaussian Splatting for Surface Reconstruction from Sparse Views

Jiang Wu Rui Li Yu Zhu† Rong Guo Jinqiu Sun Yanning Zhang†
Northwestern Polytechnical University

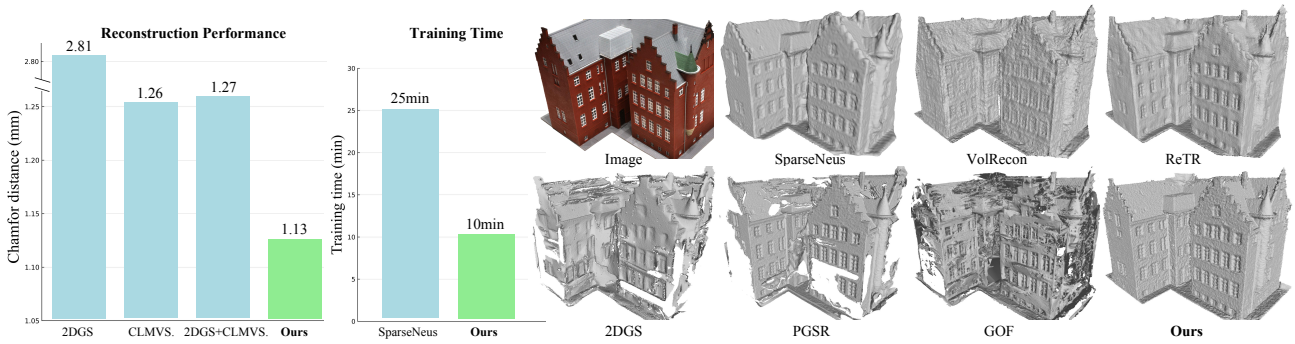

Figure 1. **Sparse2DGS** boosts the strengths of Gaussian Splatting and Multi-view Stereo in sparse-view surface reconstruction, with notable improvement over 2DGS [11], CLMVSNet [31], as well as their plain combination (*left*). Meanwhile, Sparse2DGS is 2× faster than NeRF-based sparse-view method [20] (*middle*) and leads to complete and accurate reconstruction (*right*).

Abstract

We present a Gaussian Splatting method for surface reconstruction using sparse input views. Previous methods relying on dense views struggle with extremely sparse Structure-from-Motion points for initialization. While learning-based Multi-view Stereo (MVS) provides dense 3D points, directly combining it with Gaussian Splatting leads to suboptimal results due to the ill-posed nature of sparse-view geometric optimization. We propose Sparse2DGS, an MVS-initialized Gaussian Splatting pipeline for complete and accurate reconstruction. Our key insight is to incorporate the geometric-prioritized enhancement schemes, allowing for direct and robust geometric learning under ill-posed conditions. Sparse2DGS outperforms existing methods by notable margins while being 2× faster than the NeRF-based fine-tuning approach. Code is available at https://github.com/Wuuu3511/Sparse2DGS.

1. Introduction

Reconstructing 3D surfaces from multiple views is a fundamental task in computer vision and has been widely used in autonomous driving [9, 18] and virtual reality [21]. While typical methods require abundant input images, reconstructing accurate 3D surfaces from sparse views is particularly challenging and is crucial for scenes that are frequently captured with limited images.

NeRF-based surface reconstruction methods [22] have gained popularity by encoding scene geometry into an occupancy field [17, 23] or a signed distance function (SDF) [28, 38]. However, when faced with sparse views, these methods often yield incomplete surfaces with artifacts due to insufficient supervision. Although sparse-view methods [19, 20, 25, 34] use pre-trained feed-forward NeRFs to provide geometric priors, they still exhibit limited generalization to unseen scenes.

Recently, 3D Gaussian Splatting (3DGS) [14] has leveraged Structure-from-Motion (SfM) point cloud to construct Gaussian primitives for fast and high-quality novel view synthesis (NVS). Subsequent works [4, 5, 7, 11, 41, 43] aim to achieve more efficient 3D surface reconstruction using 3DGS. Notably, 2D Gaussian Splatting (2DGS) [11] proposes a 2D variant of Gaussian primitives, with disk-shaped primitives suitable for geometry representation. However, when dealing with sparse views, existing methods [4, 11, 43] struggle to produce faithful reconstructions due to initially over-sparse SfM points.

On the other hand, learning-based multi-view stereo (MVS) [6, 16, 30, 36] predicts dense depth maps from

† indicates corresponding authors.

pixel-wise feature matching, significantly improving geometric completeness even with sparse images. However, MVS depth networks focus on generalizable inference instead of test-time geometry enhancement by design. While combining MVS point cloud with Gaussian Splatting is intuitively effective, their plain combination (see Fig. 1, "2DGS+CLMVS") yields sub-optimal results due to the ill-posed nature of geometry optimization from sparse views.

In this paper, we propose Sparse2DGS, an MVS-initialized Gaussian Splatting method for complete and accurate sparse-view 3D surface reconstruction. The key idea is to boost the strengths of two paradigms through *geometric-prioritized* enhancement against the ill-posed nature of sparse-view reconstruction. Starting from MVS initialization, we first incorporate MVS-derived geometric features into the Gaussian splatting process while keeping feature and color values fixed. This provides geometrically expressive supervision and prevents appearance overfitting, thereby improving geometric reconstruction quality. We further introduce a direct Gaussian primitive regularization, with a reparameterization-based disk sampling strategy that reformulates Gaussian primitives' position, orientation, and scale into point representations. This approach enables a unified optimization of different forms of Gaussian properties within a standard point-based, cross-view consistency loss. Moreover, instead of densifying Gaussian primitives under ill-posed conditions, we propose to selectively update the MVS-initialized Gaussian primitives with rendered geometric cues, allowing for self-guided geometric corrections.

Sparse2DGS significantly outperforms existing Gaussian Splatting methods in sparse-view surface reconstruction. Meanwhile, it demonstrates notable efficiency advantages over NeRF-based test-time optimization. Our contributions are as follows:

- We propose Sparse2DGS, a test-time sparse-view 3D surface reconstruction method that boosts the strengths of Gaussian Splatting and MVS, allowing for complete and accurate reconstruction from sparse images.
- We focus on geometric-prioritized enhancement by proposing geometrically enhanced supervision, direct Gaussian primitive regularization, and selective Gaussian update throughout the Gaussian Splatting pipeline. Sparse2DGS achieves the lowest error (1.13 CD error) compared to 2DGS (2.81) [11], GOF (2.82) [43], and PGSR (2.08) [4] on DTU dataset with 3 views. Meanwhile, Sparse2DGS is $2\times$ faster than fine-tuning NeRF-based SparseNeus [20].

2. Related Works

2.1. 3D Reconstruction via Novel View Synthesis

Recent progresses [11, 12, 25, 43] in novel view synthesis (NVS) have enabled accurate 3D surface reconstruction with only image-based supervision. Neural Radiance Fields [22] (NeRFs) have become a significant technique for representing 3D scenes. It has inspired numerous works on directly extracting surfaces by training a parameterized radiance field through differentiable volume rendering. Neus [28] and VolSDF [38] integrate surface learning into volume rendering by re-parameterizing the signed distance function as volume density, reconstructing more accurate surfaces. To reduce geometric deviations and accelerate training, subsequent works [8, 27, 29, 42] have introduced improvements in various areas.

Recently, 3D Gaussian Splatting [14] has shown impressive results in novel view synthesis (NVS), with many recent studies adopting this approach for surface reconstruction. SuGaR [10] aligns 3D Gaussians with scene surfaces by designing a regularization term and extracts an accurate mesh from the Gaussians using Poisson Reconstruction [13]. 2DGS [11] replaces 3D Gaussian ellipsoids with 2D-oriented planar disks, providing view-consistent geometric representations. RaDe-GS [44] further designs a rasterized approach to render depth and normal, enhancing geometric accuracy while ensuring rendering efficiency. GOF [43] extracts the level set from trained Gaussian primitives by constructing a Gaussian opacity field. PGSR [4] improves mesh quality by introducing single-view and multi-view geometric constraints. Additionally, GSDF [41] and NeusSG [5] reconstruct scenes by combining the advantages of 3D Gaussian Splatting and SDF networks. Despite these methods' success, the extracted geometry tends to be noisy and incomplete when the input number of training viewpoints is limited.

2.2. Learning-based Multi-view Stereo

Deep learning-based Multi-view Stereo (MVS) estimates the reference image's depth from muti-view posed images. MVSNet [36] first introduces the differentiable homography warping to build the cost volume with deep image features and then employs a 3D CNN for cost regularization. Recent follow-up works have introduced improvements in various aspects, including multi-view feature extraction [2, 6], cost aggregation [32, 33, 35, 46], and cost volume regularization [30]. However, these methods require ground-truth depth maps for training, which are challenging to obtain in practice. Unsupervised MVS [3, 15, 31] can be applied in more general, real-world scenarios. MVS methods can infer dense depth maps for each input view and therefore, we select the unsupervised CLMVSNet [31] to initialize the Gaussian point cloud.

2.3. Sparse View Surface Reconstruction

Recent works have explored 3D surface reconstruction from sparse training views. SparseNeuS [20] achieves reasonable mesh reconstruction results by constructing a cascaded

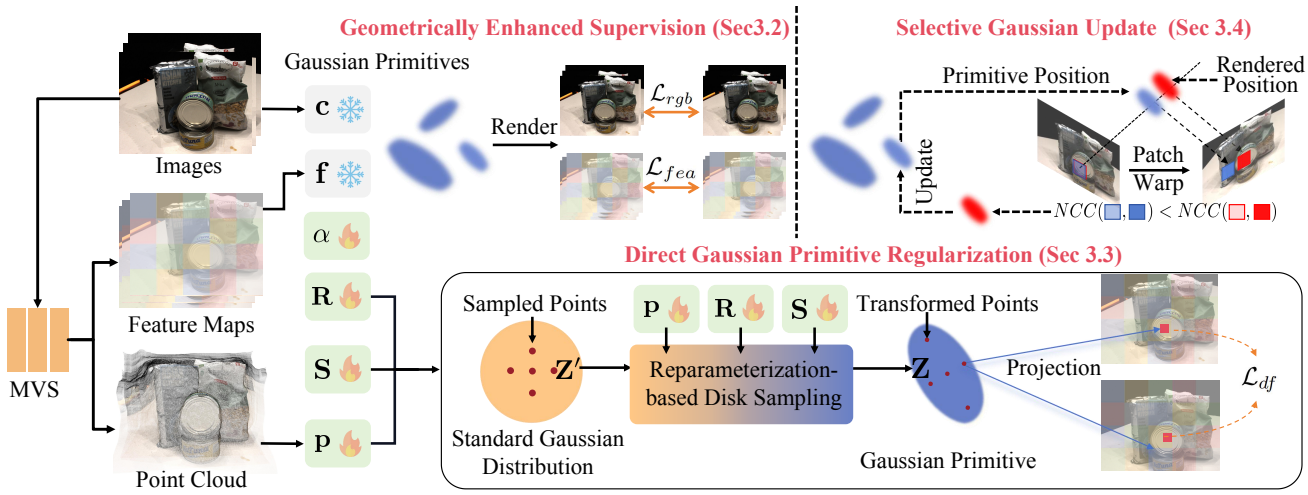

Figure 2. **Overview.** Given sparse posed images, we first leverage the MVS points to initialize the Gaussian position $\mathbf{p}$ (Sec. 3.1). We then leverage the MVS-derived feature for Gaussian Splatting, with fixed feature $\mathbf{f}$ and color $\mathbf{c}$ values to conduct geometrically expressive supervision and avoid appearance overfitting (Sec. 3.2). We then optimize Gaussian primitive properties by reformulating orientation $\mathbf{R}$, scale $\mathbf{S}$, and position $\mathbf{p}$ into sampled points using the reparameterization-based disk sampling, ensuring simultaneously optimizing different forms of Gaussian properties with point-based loss (Sec. 3.3). Finally, we introduce a Selective Gaussian Update strategy that leverages rendered geometry as an alternative to update Gaussian primitive positions, which further improves reconstruction quality (Sec. 3.4).

volume to learn a generalizable radiance field, followed by per-scene fine-tuning. NeuSurf [12] learns dense on-surface priors from sparse points to guide geometric learning. SparseCraft [40] leverages stereopsis cues for geometric regularization based on Taylor expansion. Moreover, VolRecon [25] and ReTR [19] enhance voxel features by designing specialized transformer architectures. C2F2Neus [34] constructs per-view cost frustum for finer geometry estimation. Although these feedforward networks can directly infer scene geometry, their limited generalization to unseen scenes restricts their applicability in real-world scenarios.

3. Methodology

Given sparse view posed images $\{\mathbf{I}_i\}_{i=1}^N$, where $\mathbf{I}_i \in \mathbb{R}^{H\times W\times 3}$ and N denotes the view number, Sparse2DGS aims to reconstruct accurate 3D scene surfaces using Gaussian Splatting. As shown in Fig. 2, we first initialize the Gaussian primitives with the MVS [31] reconstructed point cloud. Then, we sample MVS geometric feature maps into feature splatting with its value and RGB color fixed for supervision. For each Gaussian primitive, we regularize its geometric properties (position, orientation, scale) by a novel reparameterization-based 2D disk sampling strategy that formulates these properties as point-based supervision. Finally, we propose a selective Gaussian update strategy to further enhance Gaussian primitives using rendered geometric cues.

3.1. 2D Gaussian Splatting with MVS initialization

2D Gaussian Splatting [11] (2DGS) represents Gaussian primitives as planar disks suitable for representing and optimizing 3D surfaces. However, it relies on sparse SfM [26] 3D points for initialization, which exhibits limited reconstruction quality under sparse-view inputs. To address this, we initialize 2DGS with an MVS point cloud to provide dense geometric representations.

2D Gaussian Splatting. Each 2D Gaussian primitive is characterized by its central position $\mathbf{p}$, two principal tangential vectors $\mathbf{t_u}$ and $\mathbf{t_v}$, and a scaling vector (s_u, s_v). The rotation matrix of a 2D Gaussian can be represented as $\mathbf{R} = (\mathbf{t_u}, \mathbf{t_v}, \mathbf{t_z})$, where $\mathbf{t_z}$ is defined by two orthogonal tangent vectors $\mathbf{t_z} = \mathbf{t_u} \times \mathbf{t_v}$ and the scale matrix is $\mathbf{S} = \mathrm{diag}(s_u, s_v, 0)$. A 2D Gaussian is then defined in a local tangent plane in world space, which is parameterized:

$$P(u, v) = \mathbf{p} + s_u\mathbf{t}_u u + s_v\mathbf{t}_v v. \tag{1}$$

For the point $\mathbf{u} = (u, v)$ in uv space, its 2D Gaussian value can then be evaluated by standard Gaussian

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

Initializing 2DGS with MVS depth. MVS methods estimate dense depth maps through cross-view matching. In particular, unsupervised MVS [31] reconstructs complete scenes without ground truth for training, offering greater flexibility. Therefore, we use the point cloud reconstructed by the current unsupervised method CLMVSNet [31] to initialize our Gaussian point cloud. Specifically, given sparse views' images $\{\mathbf{I}_i\}_{i=1}^N$, we use the MVS network for depth maps $\{\mathbf{D}_i\}_{i=1}^N$, then back-project these depth maps and fuse into a point cloud $\{\mathbf{p}_{ij} \mid 1 \le i \le N,\ 1 \le j \le HW\}$ as ini-

tialized positions for Gaussian primitives, where i denotes image index and j is the pixel index within each image.

3.2. Geometrically Enhanced Supervision

3D/2D Gaussian Splatting [11, 14] couples color and geometry optimization through image-based loss. However, this leads to severe degradation under sparse views as color will overfit sparse views, leading to a compromised geometry. Meanwhile, the challenges in image areas (*e.g.*, textureless regions) will be amplified with sparse views. To address this issue, we propose geometrically enhanced supervision by incorporating off-the-shelf expressive MVS features for feature splatting and supervision, with the color and feature fixed to avoid overfitting.

In addition to providing dense point clouds from input images $\{\mathbf{I}_i\}_{i=1}^N$, the MVS network [31] offers off-the-shelf feature maps with geometric expressiveness. We leverage MVS features to enhance geometric supervision by incorporating additional feature attribute $\mathbf{f}_{ij}$ into Gaussian primitives. Specifically, for each input image $\mathbf{I}_i$, we extract its feature map $\mathbf{F}_i$ from the Feature Pyramid Network (FPN) of CLMVSNet [31]. Since each 3D point corresponds to an image pixel due to MVS initialization, we sample both RGB color and MVS features into each Gaussian primitive.

$$\mathbf{f}_{ij} = \mathbf{F}_i(\pi(\mathbf{p}_{ij})), \mathbf{c}_{ij} = \mathbf{I}_i(\pi(\mathbf{p}_{ij})), \tag{3}$$

where $\pi(\cdot)$ denotes the projection from 3D space onto 2D image plane. We further render the feature into an image plane using the tile-based renderer together with RGB rendering. We show the rendering process given M Gaussian primitives along a ray, with indices i, j omitted for brevity

$$\{\hat{\mathbf{c}}, \hat{\mathbf{f}}\} = \sum_{m=1}^{M} \{\mathbf{c}_m, \mathbf{f}_m\} \cdot \alpha_m \prod_{o=1}^{m-1} (1 - \alpha_m), \tag{4}$$

where α_m is the opacity of the Gaussian. With rendered feature maps that have enhanced geometric expressiveness for the challenging sparse views, we further supervise the rendered features using cosine similarity loss by

$$\mathcal{L}_{fea} = 1 - \frac{\mathbf{f}_{ij} \cdot \hat{\mathbf{f}}_{ij}}{\|\mathbf{f}_{ij}\| \|\hat{\mathbf{f}}_{ij}\|}, \tag{5}$$

Note that the simultaneous learning of appearance (*i.e.*, color $\mathbf{c}_{ij}$ and feature $\mathbf{f}_{ij}$) and geometry lead to geometric degradation under sparse views (see Tab. 4). To address this issue, both $\mathbf{f}_{ij}$ and $\mathbf{c}_{ij}$ are *fixed* during the optimization to enforce the learning specifically for geometry.

Our method essentially differentiates previous feature-based methods [24, 47] in its specific strategy for geometric enhancement. In our method, the expressive MVS features are incorporated, and features of each Gaussian are fixed, both for geometry enhancement with sparse views. Previous methods leverage splatted features only for segmentation or editing [24, 47] with dense views, while their feature losses do not optimize the scene geometry.

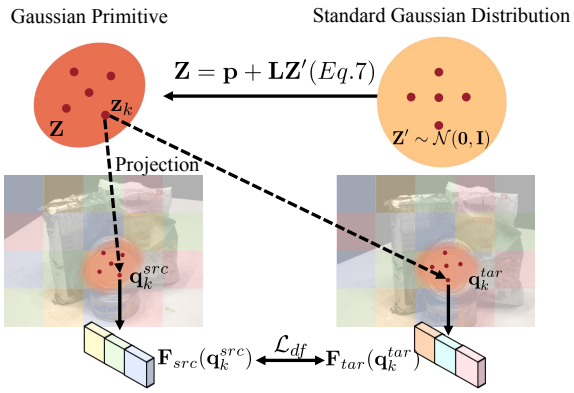

Figure 3. **Direct Gaussian Primitive Regularization.** We sample point set $\mathbf{Z}' \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ from the standard Gaussian distribution and leverage the reparameterization technique to transform $\mathbf{Z}'$ to the point set $\mathbf{Z}$ on each Gaussian primitive. This process allows for representing Gaussian orientation $\mathbf{R}$ and scale $\mathbf{S}$ properties in the form of points easier for geometric supervision. The transformed points $\mathbf{Z}$ are then projected to the source and target views, supervised by cross-view feature consistency loss.

3.3. Direct Gaussian Primitive Regularization

The disk-shaped 2D Gaussian primitives encode latent scene geometry including local shape orientation $\mathbf{R}$, scale $\mathbf{S}$, and position $\mathbf{p}$, which are essential for accurate surface reconstruction. These geometric properties are typically learned from surrogate image render loss supported by abundant multi-view supervision. When facing sparse-view inputs, the primitives tend to degrade due to insufficient geometric constraints.

To address this issue, we incorporate *direct* geometric regularization for 2D Gaussian primitives through multi-view consistency guidance. As multi-view consistency approaches only consider point position, we incorporate broad Gaussian primitive properties (*i.e.*, orientation, scale, and position) by correlating their relationship with point position, through a novel reparameterization-based disk sampling strategy. This regularization aligns 2D Gaussian primitives more closely with the latent scene surfaces, leading to improved geometric learning. In the following illustration, we omit the Gaussian primitive indices i, j for brevity.

Regularizing points with cross-view consistency. Given 3D point sampling positions $\mathbf{Z} = \{\mathbf{z}_k\}_{k=1}^K$ on each Gaussian, we project them to source and target views to obtain the corresponding 2D positions $\mathbf{q}_k^{src}$ and $\mathbf{q}_k^{tar}$, as well as queried MVS features $\mathbf{F}_{src}(\mathbf{q}_k^{src})$, $\mathbf{F}_{tar}(\mathbf{q}_k^{tar})$. We can then optimize the 3D positions by cross-view feature consistency

loss [45] $\mathcal{L}_{df}$:

$$\mathcal{L}_{df} = \frac{1}{K}\sum_{k=1}^{K}(1 - \frac{\mathbf{F}_{src}(\mathbf{q}_k^{src}) \cdot \mathbf{F}_{tar}(\mathbf{q}_k^{tar})}{\|\mathbf{F}_{src}(\mathbf{q}_k^{src})\|\|\mathbf{F}_{tar}(\mathbf{q}_k^{tar})\|}), \quad (6)$$

where K denotes the number of sampled points on each disk-shaped Gaussian primitive.

Reparameterization-based disk sampling. The above operation focuses on position regularization only, and thus, it cannot propagate gradients to other Gaussian primitive properties such as orientation $\mathbf{R}$ and scale $\mathbf{S}$. We propose a reparameterization-based sampling strategy to solve this problem. As shown in Fig. 3, for each Gaussian primitive, we first sample points $\mathbf{Z}'$ from the standard Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$ and then transform the distribution to the target Gaussian primitive through its position $\mathbf{p}$, orientation $\mathbf{R}$ and scale $\mathbf{S}$

$$\mathbf{Z} = \mathbf{p} + \mathbf{L}\mathbf{Z}', \quad (7)$$

$$\mathbf{L} = \mathbf{R}\mathbf{S}, \quad (8)$$

where $\mathbf{Z}' \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This bridges the gap between Gaussian properties (orientation and scale) and point representation, ensuring regularizing Gaussian geometric properties merely through the surrogate point-based loss. Additionally, since the last scale value of the 2D Gaussian primitive [11] is zero, resulting in difficulties in optimizing the normal vector $\mathbf{R}[\ldots, 2]$ in the last dimension, we apply additional supervision using the rendered normal $\mathbf{n}$:

$$\mathcal{L}_{dn} = 1 - \mathbf{R}[\ldots, 2]^\top \mathbf{n}, \quad (9)$$

The final primitive regularization loss can be defined as:

$$\mathcal{L}_{dr} = \mathcal{L}_{df} + \mathcal{L}_{dn}. \quad (10)$$

3.4. Selective Gaussian Update

Gaussian Splatting leverages adaptive density control to densify the Gaussian primitives. However, with sparse views, the densified Gaussian primitives are prone to degrade without sufficient multi-view supervision. To address this issue, as shown in Fig. 2, instead of continually densifying the MVS-initialized dense Gaussian primitives with limited supervision, we propose to selectively update them guided by the alternative rendered geometric cues, *e.g.*, depth and normal.

Evaluating Gaussian primitives via patch warping. Given any Gaussian primitive with its position $\mathbf{p}$ and orientation $\mathbf{R}$, we can warp an image patch from one source view to a target view using its position $\mathbf{p}$ and orientation $\mathbf{R}$ together with the pose, following homography warping techniques [3]. Then, we can evaluate the geometric quality of the Gaussian primitive via the patch-based Normalized Cross-Correlation (NCC) [39] metric.

$$\text{NCC}(x, y) = \frac{\sum_{i=l}^{n}(x_l - \bar{x})(y_l - \bar{y})}{\sqrt{\sum_{l=1}^{n}(x_l - \bar{x})^2}\sqrt{\sum_{l=1}^{n}(y_l - \bar{y})^2}}, \quad (11)$$

where x_l and y_l represent the pixels in the patches x and y of the source and target images, respectively.

Update Gaussian primitive with rendered cues. In addition to using the Gaussian primitive parameter $\mathbf{p}$ and orientation $\mathbf{R}$ for patch warping, we can leverage the rendered pixel-wise depth and normals from tiled Gaussian primitives as an alternative for patch warping. We compute patch-wise NCC over two warping strategies, yielding NCC_G for Gaussian primitive and NCC_R for rendered cues.

If NCC_R is better than NCC_G, we reproject the rendered depth into a 3D point and update the position of the corresponding Gaussian primitive. We update the position of Gaussian primitives every 100 training steps, after which all Gaussian parameters continue to be optimized by gradients. In this way, we improve the initial Gaussian primitive with geometric cues from rendered results, further enhancing the reconstruction quality.

3.5. Training loss

During training, in addition to using the disk regularization loss $\mathcal{L}_{dr}$ and MVS feature splatting loss $\mathcal{L}_{fea}$, we retain the original 2DGS [11] image reconstruction loss $\mathcal{L}_{rgb}$, depth-normal consistency loss $\mathcal{L}_n$, and depth distortion loss $\mathcal{L}_d$. The final loss function is defined as:

$$\mathcal{L} = \mathcal{L}_{rgb} + \lambda_1\mathcal{L}_d + \lambda_2\mathcal{L}_n, +\lambda_3\mathcal{L}_{dr} + \lambda_4\mathcal{L}_{fea}, \quad (12)$$

where $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ denotes the hyperparameters for the weights of each loss term.

4. Experiments

In this section, we first introduce the implementation details and then provide quantitative and qualitative comparisons with current methods on DTU [1] dataset, as well as qualitative results on BlendedMVS [37] dataset. We further analyze the contribution of each module and compare them with different methods through ablation studies.

4.1. Datasets

The DTU [1] dataset consists of 128 scenes captured in controlled lab environments using structured light scanners containing 49 or 64 images. Similar to SparseNeuS [20], we select 15 scenes and use views 23, 24, and 33 for 3-view reconstruction, with an image resolution of 576×768. We perform the quantitative evaluation on the DTU dataset using the official evaluation code to calculate the Chamfer Distance between the reconstructed mesh and the ground truth point cloud. The BlendedMVS [37] dataset includes several large-scale real-world scenes. We also show the qualitative reconstruction comparisons.

Scan	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
Colmap [26]	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
CLMVSNet [31]	1.17	2.52	1.68	0.89	1.51	1.82	0.99	1.29	1.52	0.78	0.88	1.10	0.64	1.04	1.12	1.26
Neus [28]	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 [38]	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
VolRecon [25]	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
SparseNeus [20]	1.29	2.27	1.57	0.88	1.61	1.86	1.06	1.27	1.42	1.07	0.99	0.87	0.54	1.15	1.18	1.27
Retr [19]	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 [34]	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
Sparsecraft [40]	1.17	**1.74**	1.80	**0.70**	1.19	1.53	0.83	**1.05**	1.42	0.78	0.80	0.56	0.44	0.77	**0.84**	1.04
NeuSurf [12]	**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**
2DGS [11]	3.14	3.79	2.31	1.51	5.34	2.22	1.85	2.79	3.62	1.63	3.05	4.11	1.05	3.35	2.44	2.81
GOF [43]	3.27	5.05	2.77	2.04	3.84	2.61	2.23	2.65	3.30	1.65	2.58	4.44	1.16	2.29	2.39	2.82
PGSR [4]	3.36	3.28	2.75	1.27	5.15	1.84	0.88	1.79	3.49	1.19	1.86	1.11	0.61	1.09	1.52	2.08
Ours	**1.05**	**2.35**	**1.38**	**0.83**	**1.37**	**1.45**	**0.84**	**1.16**	**1.43**	**0.74**	**0.85**	**0.84**	**0.57**	**0.95**	**1.01**	**1.13**

Table 1. **Quantitative results on DTU dataset.** Our method achieves the best performance among Gaussian Splatting surface reconstruction approaches and demonstrates competitive performance to previous state-of-the-art NeRF-based methods.

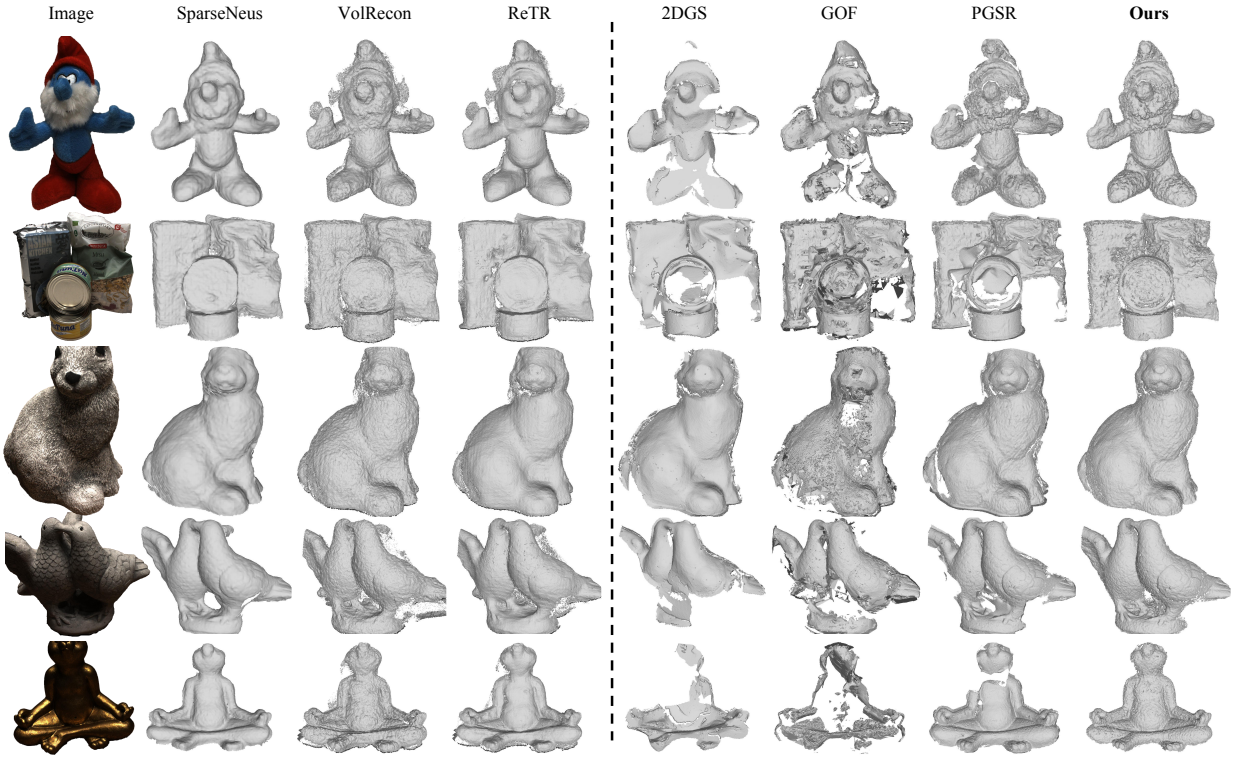

Figure 4. **DTU surface reconstruction results.** Our method achieves more complete reconstructions with finer details.

4.2. Implementation Details

Our method is implemented based on 2DGS [11]. For Gaussian primitive initialization, we use the point cloud back-projected depth maps from CLMVSNet [31]. Instead of using spherical harmonics to represent the color of the Gaussian primitives, we initialize the color using the RGB values from the image. CLMVSNet uses an FPN for feature extraction, and we utilize the full-resolution feature maps with a channel dimension of 8 from the FPN last layer for feature splatting, The remaining parameters are initialized consistently with those in 2DGS. We set the training iteration to 7000 and replaced the original 2DGS's adaptive density control with our proposed Gaussian update strategy. Similar to 2DGS [11], we used truncated signed distance fusion (TSDF) to extract the final mesh. During TSDF fusion, the voxel size and truncated threshold are set to 0.004 and 0.02 respectively. We set the hyperparameters of the loss function, λ_1, λ_2, λ_3, and λ_4, to 1000, 0.05, 1, and 0.2, respectively.

4.3. Benchmark Comparisons

We compare the sparse-view reconstruction performance of different NeRF-based and Gaussian-based methods on

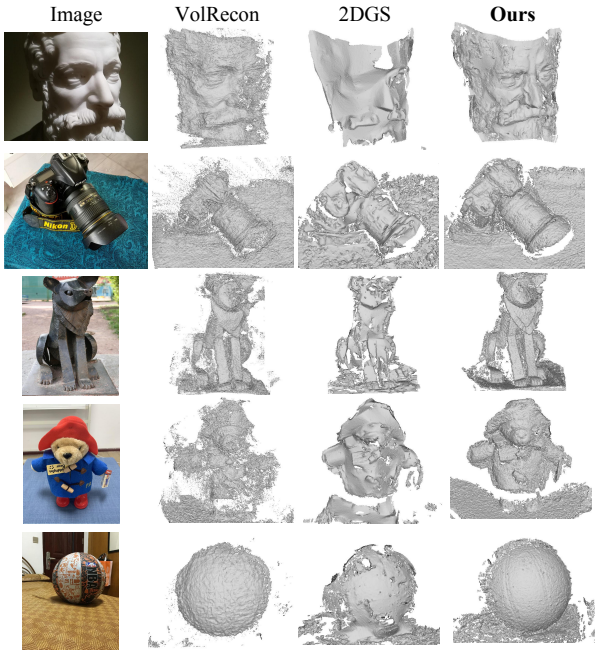

Figure 5. **BlendedMVS reconstruction results.** Our method achieves more complete and detailed reconstructions.

	2DGS	MVS Init.	Fixed Color	Feat. Splat.	DGPR	SGU	CD ↓
(a)	✓						2.813
(b)	✓	✓					1.273
(c)	✓	✓	✓				1.212
(d)	✓	✓		✓			1.224
(e)	✓	✓	✓	✓			1.201
(f)	✓	✓	✓	✓	✓		1.142
(g)	✓	✓	✓	✓		✓	1.152
(h)	✓	✓	✓	✓	✓	✓	**1.125**

Table 2. **Ablation on DTU [1] dataset.** DGPR and SGU represent proposed Direct Gaussian Primitive Regularization and Selective Gaussian Update respectively. Each module improves upon the baseline, leading to the best final results.

the DTU dataset. As shown in Tab. 1, our method significantly improves upon Gaussian-based approaches under sparse views, outperforming baseline (2DGS) and previous Gaussian-based SOTA methods (PGSR) by 67% and 46%, respectively. Our method also outperforms SparseNeus [20], the first sparse-view NeRF-based method with notable margins. Our method performs competitively with the state-of-the-art NeRF-based surface reconstruction methods. The visualization results are shown in Fig. 4. Our method captures more surface details than implicit reconstruction approaches and outperforms Gaussian-based methods in preserving the complete scene structure. In comparison, Gaussian Splatting methods, *i.e.*, 2DGS, GOF, and PGSR, tend to exhibit noise and missing areas. Additionally, we performed a qualitative comparison on the BlendedMVS dataset. As shown in Fig. 5, our method

Methods	2DGS [11]	PGSR [4]	CLMVS. [31]	Ours
3 view SfM Point Cloud	2.81	2.08	-	-
49/64 views SfM Point Cloud	1.65	1.53	-	-
CLMVSNet Point Cloud	1.27	1.38	1.26	**1.13**

Table 3. **Comparison of initialization strategies.** Initializing with MVS point cloud improves reconstruction performance. However, a plain combination (Chamfer Distance, CD:1.27) between 2DGS and CLMVSNet fails to surpass CLMVSNet results (CD: 1.26). Our method enhances both MVS and 2DGS through geometry-prioritized optimization.

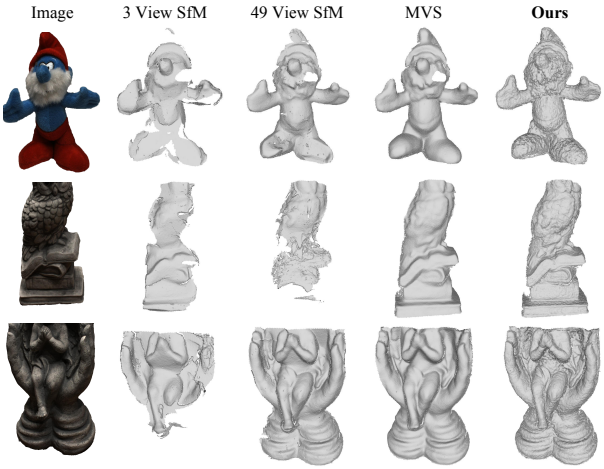

Figure 6. **Comparison between different initialization strategies of 2DGS and our method.** Initializing 2DGS with MVS point cloud improves SfM reconstruction completeness but struggles to capture details. Our method not only enhances completeness but also achieves detailed reconstruction.

achieves higher accuracy in reconstructing meshes for real-world scenes.

4.4. Ablations and Analysis

In this section, we perform ablation studies to analyze the effectiveness of different modules. As shown in Tab. 2, using MVS point cloud initialization notably enhances the performance (row **a**). By fixing the colors and using feature splatting for geometric enhancement supervision, we further improve the accuracy and completeness of the reconstruction (row **e**). The Direct Gaussian Primitive Regularization (DGPR) enhances the surface modeling capability of each Gaussian primitive (row **f**). The Selective Gaussian Update (SGU) leverages rendering cues to enhance the precision of the Gaussian point set, further improving surface accuracy (row **g**).

Comparison of different initialization strategies. We compare the reconstruction results using various initialization strategies for Gaussian Splatting methods. We show the results using 3 different initialization strategies: SfM point cloud from 3 viewpoints, SfM point cloud from 49 viewpoints, and the CLMVSNet reconstructed point cloud.

Settings	Accuracy↓	Completion↓	Average↓
SH=3	0.83	1.71	1.27
SH=2	0.84	1.68	1.26
SH=1	0.84	1.64	1.24
Fixed Color	**0.81**	**1.61**	**1.21**

Table 4. **Ablation study on appearance learning.** Fixing the color helps in learning better geometry.

Methods	Accuracy↓	Compleness↓	Average↓
Baseline	0.790	1.612	1.201
Adaptive Density Control [14]	0.814	1.654	1.234
Selective Gaussian Update	**0.760**	**1.544**	**1.152**

Table 5. **Ablation on DTU [1] dataset.** The proposed Selective Gaussian Update strategy enhances reconstruction accuracy and completeness compared to the generic Adaptive Density Control in sparse views.

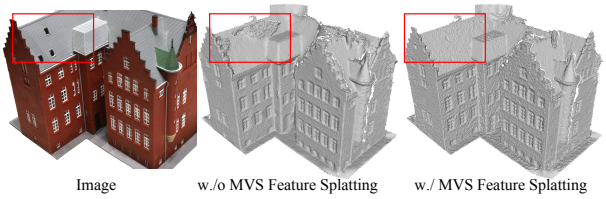

Figure 7. **Visualization of MVS feature splatting.** Incorporating MVS feature splatting aids in achieving higher reconstruction accuracy in challenging regions.

Using MVS point clouds achieves a more complete reconstruction than SfM. Note that as shown in Tab. 3, due to the inherent ill-posedness in sparse-view reconstruction, the plain combination of MVS and 2DGS reconstruction yields suboptimal performance. Even with CLMVSNet point clouds as initialization, the reconstruction quality still falls short of that achieved by CLMVSNet itself. In contrast, our geometric-prioritized enhancement addresses this challenge, with results excelling both CLMVSNet and 2DGS with higher completeness and finer details (Fig. 6).

Geometrically enhanced supervision. In this paper, we mitigate appearance overfitting by fixing the color and feature parameters during optimization. In the original Gaussian splatting, view-independent color is represented by spherical harmonics (SH) with learnable coefficients. Intuitively, reducing SH degrees can achieve a similar goal in alleviating color overfitting. We compare the fixed color and reduced SH degrees in Tab. 4. Results show that reducing the SH degrees also mitigates the degradation in geometry, with only 1 SH degree outperforming other settings (*i.e.*, 2, 3) in surface reconstruction. However, using learnable color coefficients still performs worse than the adopted choice of fixed color during training. Moreover, we enhance geometric supervision using MVS feature splatting. As shown in Fig. 7, MVS features are more expressive and provide more effective constraints, especially in challenging regions such as repetitive texture areas (*e.g.*, the roof of the building).

Ablation on Direct Gaussian Primitive Regularization.

Settings	Accuracy↓	Completion↓	Average↓
K=9	0.746	1.544	1.145
K=25	0.747	1.541	1.144
K=49	0.751	1.532	1.142
K=81	**0.747**	**1.531**	**1.139**

Table 6. **Comparison of different numbers of sampled points.** Increasing the sampling numbers lead to a slight performance improvement.

Methods	SparseNeus [20]	NeuSurf [12]	Ours
Chamfer Distance	1.27	**0.99**	1.13
Training Time	25mins	10 hours	**10mins**

Table 7. **Performance and efficiency.** Our method achieves comparable performance while requiring the least training time.

We analyze the impact of the reparameterization-based disk sampling number K on the final results. As shown in Tab. 6, increasing the number of sampled points can slightly enhance the final reconstruction performance.

Selective Gaussian update strategy. To validate the effectiveness of our selective update strategy, we compared it with the adaptive density control in the original 2DGS. Quantitative results are shown in the Tab. 5. Due to the lack of sufficient multi-view supervision, the densified Gaussian primitives are prone to degradation, leading to declined reconstruction performance (CD:1.23) compared to the baseline (CD: 1.20). The proposed selective Gaussian update strategy enhances the accuracy of the original Gaussian point set by utilizing high-precision points from the rendered cues, which further improves the surface reconstruction quality.

Performance and efficiency. We compared reconstruction performance and training time with previous state-of-the-art surface reconstruction methods. As shown in Tab. 7, our method achieves better performance and $2\times$ faster training speed than SparseNeus [20]. Meanwhile, our method achieves competitive performance while being $60\times$ faster than NeuSurf [12].

5. Conclusion

In this paper, we propose Sparse2DGS, addressing the challenge of ill-posed geometry optimization from sparse views through geometry-prioritized enhancement schemes. Our method not only reconstructs accurate and complete scene surfaces but also achieves higher efficiency than NeRF-based methods. *Limitation*: Despite their effectiveness in dense matching, MVS methods inevitably encounter degradation under occlusions. The error in occluded areas can propagate to our method, leading to potential performance declines.

Acknowledgements Y. Zhang was supported by NSFC (No.U19B2037) and the Natural Science Basic Research Program of Shaanxi (No.2021JCW-03). Y. Zhu was supported by NSFC (No.61901384).

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120:153–168, 2016. 5, 7, 8

[2] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer: Learning robust image representations via transformers and temperature-based depth for multi-view stereo. *arXiv preprint arXiv:2208.02541*, 2022. 2

[3] Di Chang, Aljaž Božič, Tong Zhang, Qingsong Yan, Yingcong Chen, Sabine Süsstrunk, and Matthias Nießner. Rc-mvsnet: Unsupervised multi-view stereo with neural rendering. In *European conference on computer vision*, pages 665–680. Springer, 2022. 2, 5

[4] Danpeng Chen, Hai Li, Weicai Ye, Yifan Wang, Weijian Xie, Shangjin Zhai, Nan Wang, Haomin Liu, Hujun Bao, and Guofeng Zhang. Pgsr: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. *arXiv preprint arXiv:2406.06521*, 2024. 1, 2, 6, 7

[5] Hanlin Chen, Chen Li, and Gim Hee Lee. Neusg: Neural implicit surface reconstruction with 3d gaussian splatting guidance. *arXiv preprint arXiv:2312.00846*, 2023. 1, 2

[6] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8585–8594, 2022. 1, 2

[7] Lue Fan, Yuxue Yang, Minxing Li, Hongsheng Li, and Zhaoxiang Zhang. Trim 3d gaussian splatting for accurate geometry representation. *arXiv preprint arXiv:2406.07499*, 2024. 1

[8] Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *Advances in Neural Information Processing Systems*, 35:3403–3416, 2022. 2

[9] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3354–3361. IEEE, 2012. 1

[10] Antoine Guédon and Vincent Lepetit. 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*, pages 5354–5363, 2024. 2

[11] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 1, 2, 3, 4, 5, 6, 7

[12] Han Huang, Yulun Wu, Junsheng Zhou, Ge Gao, Ming Gu, and Yu-Shen Liu. Neusurf: On-surface priors for neural surface reconstruction from sparse input views. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2312–2320, 2024. 2, 3, 6, 8

[13] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, 2006. 2

[14] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1, 2, 4, 8

[15] Tejas Khot, Shubham Agrawal, Shubham Tulsiani, Christoph Mertz, Simon Lucey, and Martial Hebert. Learning unsupervised multi-view stereopsis via robust photometric consistency. *arXiv preprint arXiv:1905.02706*, 2019. 2

[16] Rui Li, Dong Gong, Wei Yin, Hao Chen, Yu Zhu, Kaixuan Wang, Xiaozhi Chen, Jinqiu Sun, and Yanning Zhang. Learning to fuse monocular and multi-view cues for multi-frame depth estimation in dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21539–21548, 2023. 1

[17] Rui Li, Tobias Fischer, Mattia Segu, Marc Pollefeys, Luc Van Gool, and Federico Tombari. Know your neighbors: Improving single-view reconstruction via spatial vision-language reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9848–9858, 2024. 1

[18] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1477–1485, 2023. 1

[19] Yixun Liang, Hao He, and Yingcong Chen. Retr: Modeling rendering via transformer for generalizable neural surface reconstruction. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 3, 6

[20] Xiaoxiao Long, Cheng Lin, Peng Wang, Taku Komura, and Wenping Wang. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In *European Conference on Computer Vision*, pages 210–227. Springer, 2022. 1, 2, 5, 6, 7, 8

[21] Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *ACM Transactions on Graphics (ToG)*, 39(4):71–1, 2020. 1

[22] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2

[23] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 1

[24] Ri-Zhao Qiu, Ge Yang, Weijia Zeng, and Xiaolong Wang. Feature splatting: Language-driven physics-based scene synthesis and editing. *arXiv preprint arXiv:2404.01223*, 2024. 4

[25] Yufan Ren, Fangjinhua Wang, Tong Zhang, Marc Pollefeys, and Sabine Süsstrunk. Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16685–16695, 2023. 1, 2, 3, 6

[26] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 3, 6

[27] Cheng Sun, Min Sun, and Hwann-Tzong Chen. Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5459–5469, 2022. 2

[28] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 1, 2, 6

[29] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. Improved surface reconstruction using high-frequency details. 2022. 2

[30] Jiang Wu, Rui Li, Haofei Xu, Wenxun Zhao, Yu Zhu, Jinqiu Sun, and Yanning Zhang. Gomvs: Geometrically consistent cost aggregation for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20207–20216, 2024. 1, 2

[31] Kaiqiang Xiong, Rui Peng, Zhe Zhang, Tianxing Feng, Jianbo Jiao, Feng Gao, and Ronggang Wang. Cl-mvsnet: Unsupervised multi-view stereo with dual-level contrastive learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3769–3780, 2023. 1, 2, 3, 4, 6, 7

[32] Jialei Xu, Xianming Liu, Yuanchao Bai, Junjun Jiang, Kaixuan Wang, Xiaozhi Chen, and Xiangyang Ji. Multi-camera collaborative depth prediction via consistent structure estimation. In *Proceedings of the 30th ACM international conference on multimedia*, pages 2730–2738, 2022. 2

[33] Jialei Xu, Wei Yin, Dong Gong, Junjun Jiang, and Xianming Liu. Sdge: Stereo guided depth estimation for 360° camera sets. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 11179–11186. IEEE, 2024. 2

[34] Luoyuan Xu, Tao Guan, Yuesong Wang, Wenkai Liu, Zhaojie Zeng, Junle Wang, and Wei Yang. C2f2neus: Cascade cost frustum fusion for high fidelity and generalizable neural surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18291–18301, 2023. 1, 3, 6

[35] Qingshan Xu, Wanjuan Su, Yuhang Qi, Wenbing Tao, and Marc Pollefeys. Learning inverse depth regression for pixelwise visibility-aware multi-view stereo networks. *International Journal of Computer Vision*, 130(8):2040–2059, 2022. 2

[36] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 1, 2

[37] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020. 5

[38] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021. 1, 2, 6

[39] Jae-Chern Yoo and Tae Hee Han. Fast normalized cross-correlation. *Circuits, systems and signal processing*, 28:819–843, 2009. 5

[40] Mae Younes, Amine Ouasfi, and Adnane Boukhayma. Sparsecraft: Few-shot neural reconstruction through stereopsis guided geometric linearization. In *European Conference on Computer Vision*, pages 37–56. Springer, 2024. 3, 6

[41] Mulin Yu, Tao Lu, Linning Xu, Lihan Jiang, Yuanbo Xiangli, and Bo Dai. Gsdf: 3dgs meets sdf for improved rendering and reconstruction. *arXiv preprint arXiv:2403.16964*, 2024. 1, 2

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

[43] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient and compact surface reconstruction in unbounded scenes. *arXiv preprint arXiv:2404.10772*, 2024. 1, 2, 6

[44] Baowen Zhang, Chuan Fang, Rakesh Shrestha, Yixun Liang, Xiaoxiao Long, and Ping Tan. Rade-gs: Rasterizing depth in gaussian splatting. *arXiv preprint arXiv:2406.01467*, 2024. 2

[45] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6525–6534, 2021. 5

[46] Jingyang Zhang, Shiwei Li, Zixin Luo, Tian Fang, and Yao Yao. Vis-mvsnet: Visibility-aware multi-view stereo network. *International Journal of Computer Vision*, 131(1): 199–214, 2023. 2

[47] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21676–21685, 2024. 4