

AtlasGS: Atlanta-world Guided Surface Reconstruction with Implicit Structured Gaussians

Xiyu Zhang^{1*} Chong Bao^{1*} Yipeng Chen¹ Hongjia Zhai¹ Yitong Dong¹
Hujun Bao¹ Zhaopeng Cui¹ Guofeng Zhang^{1†}
¹State Key Lab of CAD & CG, Zhejiang University

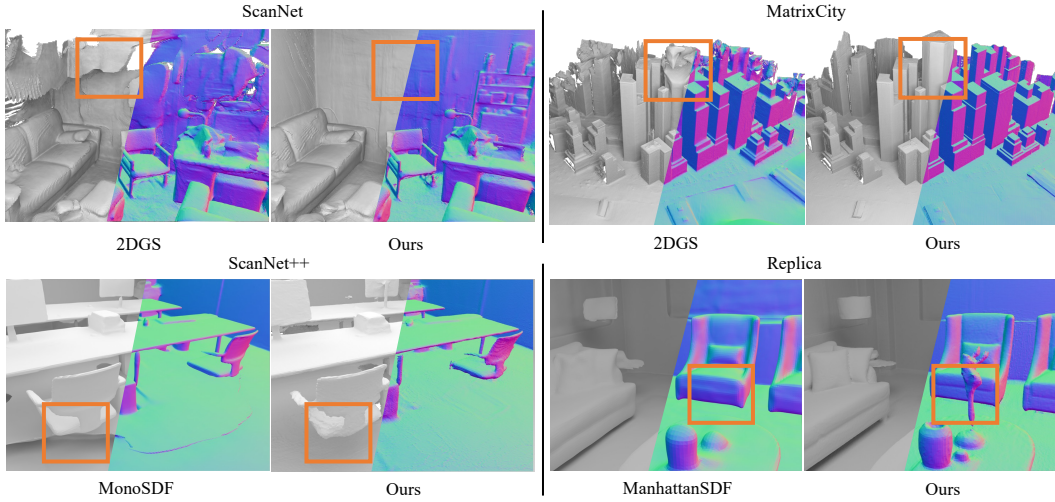


Figure 1: **Qualitative comparison of indoor and outdoor reconstruction.** For each scene, we visualize both the reconstructed geometry and the surface normals. As highlighted by the orange boxes, our method generates significantly smoother surfaces while capturing finer geometric details, clearly outperforming the compared methods.

Abstract

3D reconstruction of indoor and urban environments is a prominent research topic with various downstream applications. However, existing geometric priors for addressing low-texture regions in indoor and urban settings often lack global consistency. Moreover, Gaussian Splatting and implicit SDF fields often suffer from discontinuities or exhibit computational inefficiencies, resulting in a loss of detail. To address these issues, we propose an Atlanta-world guided implicit-structured Gaussian Splatting that achieves smooth indoor and urban scene reconstruction while preserving high-frequency details and rendering efficiency. By leveraging the Atlanta-world model, we ensure the accurate surface reconstruction for low-texture regions, while the proposed novel implicit-structured GS representations provide smoothness without sacrificing efficiency and high-frequency details. Specifically, we propose a semantic GS representation to predict the probability of all semantic regions and deploy a structure plane regularization with learnable plane indicators for global accurate surface reconstruction. Extensive experiments demonstrate that our method outperforms state-of-the-art approaches in both indoor and urban scenes, delivering superior surface reconstruction quality.

*Equal contribution

†Corresponding author

1 Introduction

Recently, indoor and urban 3D reconstruction from multi-view images has emerged as a popular research area, fueled by its applications in digital twins [1–3], robotic navigation [4–6], and augmented reality [7]. These technologies demand accurate and efficient reconstruction of real-world environments. However, man-made scenes often contain large low-texture planar regions, such as floors, ceilings, and unadorned walls, which pose significant challenges for image-based 3D reconstruction. Traditional multi-view stereo methods struggle on these textureless surfaces due to the lack of distinctive visual features, leading to incomplete or distorted geometry. Geometric priors play a crucial role in addressing this challenge. Monocular geometric priors provide locally smooth geometry signals for low-texture regions [8], but they often lack global consistency across different viewpoints, often resulting in inconsistent geometry like bumpy surfaces. Alternatively, the Manhattan-world assumption [9] leverages planar priors to address low-texture regions in man-made scenes but cannot be applied in urban scenes where buildings are not mutually orthogonal structures, such as the building marked by the yellow rectangle in Fig. 1.

Concurrently, 3D representation methods for indoor and urban reconstruction have evolved rapidly and achieved remarkable success. For example, 2DGS [10] employs Gaussian splitting (GS) with surfel primitives to efficiently and effectively reconstruct complex geometry. However, its discrete primitives induce discontinuities in surface reconstruction, resulting in broken surfaces or under-observed regions, as depicted in Fig. 1. Previous implicit SDF representations [11–14] leverage the inductive continuity of coordinate-based multi-layer perceptrons (MLPs) to recover complete surfaces in these challenging regions. However, they are computationally expensive and struggle to represent complex geometries. Some methods [15–17] have explored the simultaneous learning of both representations, using implicit methods to guide GS optimization for smoother outcomes. Unfortunately, this mutual interaction often compromises reconstruction quality.

Based on these observations, we identify two critical challenges: **1)** A globally consistent geometric prior is essential to regularize low-texture regions in both indoor and urban reconstructions; **2)** A 3D representation is needed that retains the efficiency and detail-preserving capabilities of GS while incorporating the smoothness of implicit methods.

In this paper, we propose an Atlanta-world guided implicit-structured Gaussian Splatting for indoor and urban scene reconstruction.

First, the man-made indoor and urban environments can be described as an Atlanta world model [18] where there is one vertical direction aligned with gravity and multiple horizontal directions oriented from walls or urban buildings. To establish globally consistent and accurate geometry in low-texture regions, such as floor, ceiling, and walls, we incorporate this global geometric assumption into GS optimization, thereby regularizing the geometric relationships among these regions. Specifically, we propose a semantic GS representation to predict the semantic probability of floor, ceiling, and wall in the 3D scene. Besides, we design a structure plane regularization with learnable explicit plane indicators for globally accurate surface reconstruction.

Secondly, we propose a novel implicit-structured Gaussian representation, which integrates the continuity of implicit functions with the efficiency and detail preservation of 2DGS. Unlike prior works that simply overlay implicit priors on Gaussian optimization, we embed implicit voxel grids within the Gaussian Splatting framework, allowing implicit geometry to act as a smooth regularizer while maintaining high-frequency representation. Besides, we adopt a view-independent decoding on GS spatial distribution to enhance the geometric consistency across multiple viewpoints. Our representation not only improves the smoothness of surface reconstruction but also facilitates the efficient modeling of intricate geometry, striking a better balance between accuracy and efficiency.

The contribution of our paper can be summarized as: **1)** We propose a novel Atlanta-world guided implicit-structured Gaussian Splatting to achieve smooth indoor and urban scene reconstruction while preserving high-frequency details and rendering efficiency. **2)** To integrate the Atlanta-world assumption, we design a semantic GS representation to predict the semantic probability of low-texture regions such as floor, ceiling, and walls, and propose a structure plane regularization with learnable explicit plane indicators to regularize the global geometry of these low-texture regions. **3)** Extensive experiments in indoor and urban scenes demonstrate the effectiveness of our method. We show the best surface reconstruction quality quantitatively and qualitatively compared to other state-of-the-art methods.

2 Related Works

Neural Implicit Surface Reconstruction. Neural Radiance Fields [19] (NeRF) designs a neural implicit representation to reconstruct the scene from multi-view 2D images. To obtain the scene surface, some NeRF variants [11, 12, 20] combine SDF-based neural representation with volume rendering for better geometry reconstruction. In addition, to obtain better reconstruction results in some challenging scenarios, some research works introduced more prior information during the optimization process, such as geometric regularization [13], monocular depth [21–23], normal [14, 23, 24, 20], and semantics [13, 25, 26]. However, due to the limited representation capacity of MLPs [27], relying solely on MLPs may result in slow optimization and poor reconstruction performance of large scenes. Therefore, additional feature encoding is used to enhance the scene representation ability and speed up the reconstruction, such as dense feature grid [28, 22], hash table [29, 30], sparse voxel [31–33], and tetrahedron [34–38]. However, despite the use of feature encoding, all of these implicit methods still require hours of training and exhibit low inference times and insufficient detail during the reconstruction. In contrast, our method enables efficient surface reconstruction (< 30 minutes) with high-quality details.

Surface Reconstruction with Gaussian Splatting. 3D Gaussian Splatting [39] (3DGS) has emerged as a promising technique for efficient and high-quality novel view synthesis. Starting from SfM points, 3DGS represents the scene with 3D Gaussians and employs fast splatting-based rasterization to accelerate both training and inference. SuGaR [40] extends 3DGS to surface reconstruction by associating 3D Gaussians with the mesh surface and jointly optimizing both the Gaussians and the mesh. PGSR [41] utilizes planar-based Gaussian splatting combined with unbiased depth rendering to maintain global geometric accuracy. To reconstruct complete surfaces in unbounded scenes, GOF [42] leverages a ray-tracing-based volume rendering, which enables a mesh to be extracted directly from the Gaussian representation. Furthermore, 2DGS [10] and Gaussian Surfels [43] argue that the multi-view inconsistency inherent in 3DGS compromises reconstruction quality. To address this, they replace 3D Gaussians with 2D surfels, enabling more precise capture of intricate geometric details. However, the discrete nature of Gaussian Splatting undermines the smoothness of reconstructed surfaces, particularly in regions with low texture or limited observational coverage. DN-Splatter [44] employs depth and normal priors to improve the smoothness of surface reconstruction based on 3DGS. Certain methods [15, 45] simultaneously learn an implicit Signed Distance Function (SDF) field alongside 3D Gaussian Splatting, utilizing the smooth SDF field to regularize the noisy geometry inherent in 3DGS. However, the mutual interaction between these components often compromises reconstruction quality, resulting in suboptimal outcomes. In contrast, our approach integrates the implicit field with Gaussian Splatting under the Atlanta-world assumption, enabling smooth surface reconstruction while preserving high-frequency geometric details.

3 Methods

Our goal is to reconstruct scenes with strong structural priors from posed images. To achieve this, we first introduce the preliminaries of 2DGS [10] in Sec. 3.1, followed by our implicit-structured representation in Sec. 3.2. To leverage structural priors under the Atlanta world assumption, we propose 3D global planar regularization and 2D local surface regularization, as detailed in Sec. 3.3. Finally, we describe the training process in our framework. An overview of our approach is illustrated in Fig. 2.

3.1 Preliminary

Instead of representing scenes with blobs as 3D Gaussian Splatting [39], 2DGS [10] models scenes as surfels distributed around the surfaces. Each surfel is defined in a local tangent plane in world space. To render images, 2DGS rasterizes surfels to the image plane with tile-based rasterization and Ray-Splat intersection to reduce depth bias. Then, 2DGS performs alpha blending [46] to get the rendered attribute A by composing these primitives sorted by depth:

$$A = \sum_i a_i \cdot \alpha_i \cdot \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (1)$$

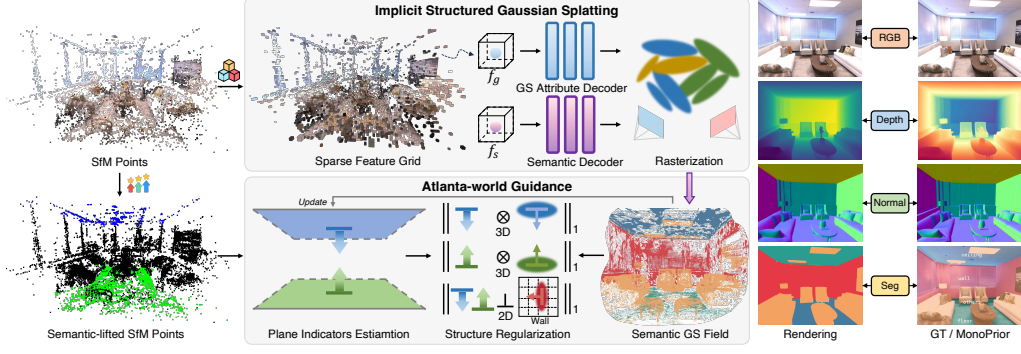


Figure 2: **Overview of AtlasGS.** Given posed images and SfM points, we construct a sparse feature grid and represent scenes as implicit-structured Gaussians. Attributes are decoded using attribute decoder and semantic decoder, followed by rasterization and supervision with RGB images, monocular geometry priors, and semantic maps. To address multiview inconsistency in textureless regions, we introduce learnable explicit plane indicators based on the Atlanta world assumption [18]. The indicators refine the global scene structure by regularizing Gaussian positions and orientations using 3D global planar and 2D local surface losses, ensuring alignment with structural elements such as walls, floors, and ceilings.

where A are the rendered 2D scene information (e.g., color c , depth d , normal $\mathbf{n}$, semantics z), a_i and α_i denote the 3D property and opacity contribution of i -th Gaussian primitive, and $\prod_{j=1}^{i-1} (1 - \alpha_j)$ is the accumulated transmittance. Then, 2DGS optimizes these Gaussians with photometric loss and distortion loss to reduce floaters. The Ray-Splat intersection and well-defined normal introduce multiview consistency depth, providing better reconstruction quality. Though 2DGS produces plausible surfaces and models high-frequency details with explicit Gaussian representation, the discrete nature introduces discontinuities, which leads to broken or protruding texture-less walls.

3.2 Implicit-Structured Gaussian Splatting

In this section, we present our implicit-structured Gaussian representation, which leverages a sparse feature grid and implicit functions to organize discrete Gaussian primitives, ensuring locally coherent geometry while preserving high-frequency details.

Specifically, given the sparse point clouds generated from SfM [47], we first construct a sparse feature grid $\mathcal{V} = \{\mathcal{V}_g^i, \mathcal{V}_s^i, \{\Delta_k^i\}_{k=1}^{\mathcal{K}}, l^i\}_{i=1}^{N_v}$ with a predefined voxel size, including geometry $\mathcal{V}_g^i$ and semantic features $\mathcal{V}_s^i$, offsets of $\mathcal{K}$ local Gaussians $\{\Delta_k^i\}_{k=1}^{\mathcal{K}}$, scaling factor $\{l^i\}_{i=1}^{N_v}$ shared with local Gaussians. Given a voxel located in $\mathbf{v}$, we decode all Gaussian geometry attributes of $\mathcal{K}$ local Gaussians via corresponding geometry MLP $\mathcal{M}_g(\cdot)$ and semantic MLP $\mathcal{M}_s(\cdot)$

$$\begin{aligned} \{\alpha_k, s_k, q_k\}_{\mathcal{K}}, \{z_k\}_{\mathcal{K}} &= \mathcal{M}_g(\mathcal{V}_g(\mathbf{v})), \mathcal{M}_s(\mathcal{V}_s(\mathbf{v})) \\ \{c_k\}_{\mathcal{K}} &= \mathcal{M}_g(\mathcal{V}_g(\mathbf{v}), \mathbf{d}). \end{aligned} \quad (2)$$

Here, $\mathbf{d}$ is view direction which assists in capturing view-dependent appearance, $\alpha \in \mathbb{R}$, $s \in \mathbb{R}^2$, $q \in \mathbb{R}^4$, $c \in \mathbb{R}^3$, and $z \in \mathbb{R}^4$ refer to the opacity, scale, rotation, color, and semantic attributes of each Gaussian, respectively, whose positions $\mathbf{p}$ is defined as $\mathbf{p}_k^i = \mathbf{v}_i + l \cdot \Delta_k^i$. Then we render images with all these Gaussian attributes decoded from the sparse feature grid by surfel rasterization from [10]. In contrast to 2DGS [10], which optimizes Gaussians independently, our decoder predicts local Gaussian attributes, allowing each Gaussian’s optimization to influence its neighbors and capture fine object details through the predicted Gaussian primitives. Consequently, the implicit-structured Gaussian framework combines the strengths of MLPs and Gaussians, achieving smooth local geometry while preserving high-frequency details.

Gaussian Semantic Lifting. To recognize structural regions in scenes, we lift 2D semantics to each Gaussian. We use the 2D pseudo-labels $\hat{Z}$ obtained by the pre-trained semantic segmentation model [48] as supervision and optimize the semantic Gaussians with the rendered semantic probability Z . To obtain the rendered semantics probability Z , we render the 3D semantic attribute z defined

in Eq. (2) into image space with Eq. (1), and optimize the semantic feature grid $\mathcal{V}_s$ and semantic MLP $\mathcal{M}_s$ by minimizing the cross-entropy loss:

$$Z = \sum_i z_i \cdot \mathcal{X}[\alpha_i \cdot \prod_{j=1}^{i-1} (1 - \alpha_j)], \quad (3)$$

$$\mathcal{L}_{\text{sem}} = -\frac{1}{|\mathcal{U}|} \sum_{u \in \mathcal{U}} \hat{Z}(u) \cdot \log Z(u), \quad (4)$$

where $\mathcal{X}[\cdot]$ denotes the stop-gradient operator, used to prevent inconsistent supervision from distorting the geometry, and $\mathcal{U}$ denotes the set of training pixels in each iteration. Here, we define $z \in \mathbb{R}^4$, indicating wall, floor, ceiling, and others.

3.3 Atlanta World Guided Planar Regularization

Man-made indoor and urban environments typically exhibit rich structural information and conform to the Atlanta world assumption [18]. To leverage the globally consistent structural priors inherent to such scenes, we first propose learnable explicit plane indicators to effectively represent scene structural information, such as the ceiling and floor. We then introduce two types of regularization: 3D global planar regularization, which refines Gaussian positions and orientations to align with the plane indicators, and 2D local surface regularization, which provides positional supervision in poorly defined wall regions by aligning them with the plane indicators.

Explicit Plane Indicators The Atlanta world assumption [18] hypothesizes that such a scene can be approximated by a combination of a dominant horizontal plane, like the floor for indoor scenes and ground for outdoor scenes, and multiple vertical planes, such as walls and buildings. Based on this assumption, we define explicit plane indicators with a gravity direction and two distance offsets to represent the floor and ceiling of an indoor scene. Specifically, the floor plane and ceiling plane can be represented with $\pi_f = (\mathbf{n}_g, d_f)$, $\pi_c = (-\mathbf{n}_g, d_c)$, and their plane parametric equations are as follows:

$$d_f + \mathbf{n}_g \cdot \mathbf{p} = 0, \quad d_c - \mathbf{n}_g \cdot \mathbf{p} = 0, \quad (5)$$

where $\mathbf{p} \in \mathbb{R}^3$ is the location of 3D points, d_f and d_c are the distances from the origin to their respective planes, and $\mathbf{n}_g$ is the gravity direction. For the urban scenes, we omit the ceiling plane.

To achieve this, we extract ceiling points and floor points, either from semantic Gaussians or semantic lifted sparse points. We then apply RANSAC [49] to regress the plane coefficients and determine the gravity direction to initialize plane indicators. Finally, we optimize the plane indicators alongside the Gaussians. Please refer to our supplementary for further details.

3D Global Planar Regularization. According to the associated probability distribution $z \in \mathbb{R}^4$, we can obtain the semantic label of each Gaussian. We enforce two critical geometric regularizations: normal alignment and planar constraint. The normal alignment enforces normals to be perpendicular to the gravity direction in wall regions while ensuring they remain parallel in ceiling and floor regions. Besides, the planar constraint ensures that the Gaussian positions $\mathbf{p}_i$ lie in their corresponding ceiling or ground plane. Thus, the 3D global planar regularization is formulated as:

$$\begin{aligned} \mathcal{L}_{3D} = & \sum_{i \in M_{\parallel}} p_{\parallel} (1 - |\mathbf{n}_g^{\top} \mathbf{n}_i|) \\ & + \sum_{i \in M_{\perp}} p_{\perp} |\mathbf{n}_g^{\top} \mathbf{n}_i| + \sum_{i \in M_c} p_c |d_c - \mathbf{n}_g^{\top} \mathbf{p}_i| + \sum_{i \in M_f} p_f |d_f + \mathbf{n}_g^{\top} \mathbf{p}_i|, \end{aligned} \quad (6)$$

where M_f , M_c , $M_{\parallel}$ and $M_{\perp}$ are the floor, ceiling, parallel, perpendicular sets, $p_{\parallel}$, $p_{\perp}$, p_c , p_f are the corresponding probabilities, $\mathbf{n}_i$ and $\mathbf{p}_i$ are the normal and position of i -th Gaussian.

2D Local Surface Regularization. Previous methods [14, 11, 13] with implicit representation derive normals from the gradient of the given points, while optimizing normals can also optimize the

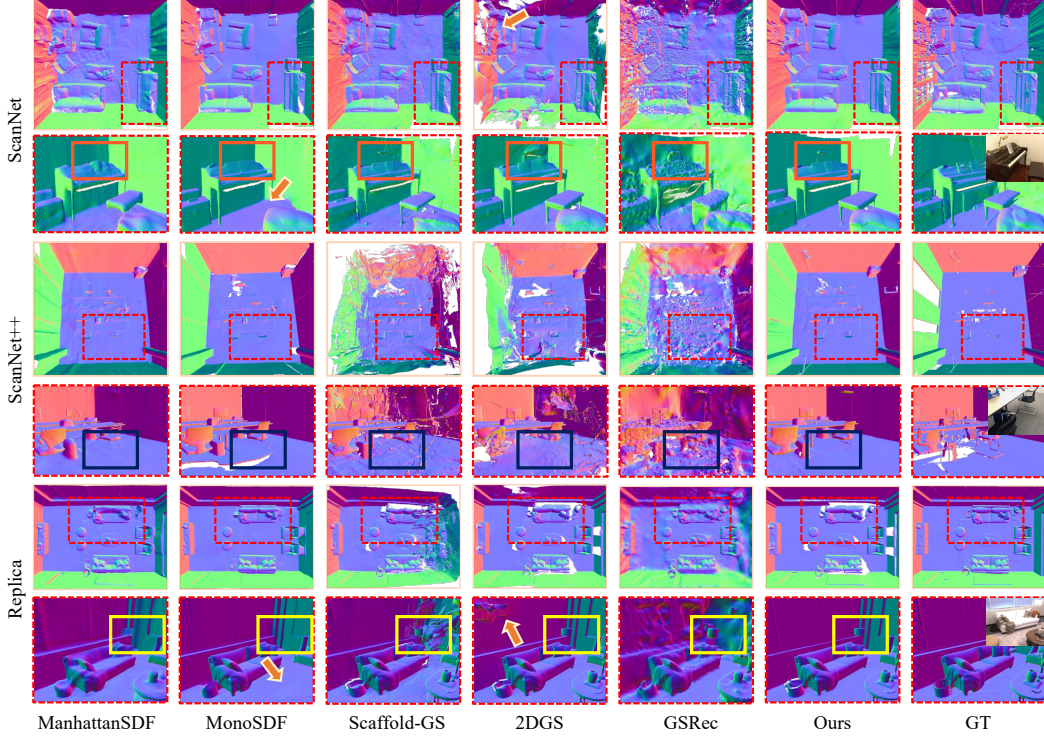


Figure 3: **Qualitative comparison of indoor scene reconstruction.** We show the reconstruction performance of the baselines and our approach on ScanNet [50], ScanNet++ [51], and Replica [52] datasets. As highlighted in the boxes, our approach maintains local smoothness and preserves high frequency. The red dashed boxes mark regions that are zoomed in below for closer inspection.

local surface. However, the Gaussian representation explicitly decouples the positions and normals of Gaussians. It poses a challenge that only optimizing Gaussian orientations in 3D space does not directly affect their spatial distribution. Without explicit positional regularization, wall Gaussians may not lie on the same plane, resulting in misalignment between the surface and the plane indicators.

To address this issue, we introduce 2D local surface regularization by regularizing the normal $\mathbf{N}_d$ from the rendered depth D . With the semantic Gaussians, we obtain coherently rendered semantics and depth, from which we derive local surface normals in wall regions. Then, we align the surface normal with our plane indicators, optimizing Gaussian positions more directly. Besides, to mitigate misclassification introduced by the semantic segmentation model [48], we weigh the loss according to the probabilities. Thus, our 2D local surface regularization loss is formulated as:

$$\mathcal{L}_{2D} = \sum_{i \in M_{\perp}} p_w (|\mathbf{N}_d \cdot \mathbf{n}_g|) + p_{c,f} (1 - |\mathbf{N}_d \cdot \mathbf{n}_g|), \quad (7)$$

$$\mathbf{N}_d = \frac{\nabla_x \mathbf{P} \times \nabla_y \mathbf{P}}{|\nabla_x \mathbf{P} \times \nabla_y \mathbf{P}|}, \quad (8)$$

where $p_{c,f}$ is the sum of the probabilities of the floor and ceiling, p_w is the probability of the wall, $\mathbf{P}$ are the points backprojected from depth, ∇ is the gradient operator. So, our structural regularization loss is:

$$\mathcal{L}_{\text{reg}} = \mathcal{L}_{3D} + \mathcal{L}_{2D}. \quad (9)$$

3.4 Training

We optimize Gaussians and structure with the following:

$$\mathcal{L} = \mathcal{L}_{\text{rgb}} + \lambda_1 \mathcal{L}_{\text{depth}} + \lambda_2 \mathcal{L}_{\text{normal}} + \lambda_3 \mathcal{L}_{\text{reg}} + \lambda_4 \mathcal{L}_{\text{sem}} + \lambda_5 \mathcal{L}_{\text{dist}} + \lambda_6 \mathcal{L}_{\text{nc}}, \quad (10)$$

where $\mathcal{L}_{\text{rgb}}$ is photometric loss proposed in original 3DGS, the distortion loss $\mathcal{L}_{\text{dist}}$ and normal consistency loss $\mathcal{L}_{\text{nc}}$ from 2DGS [10]. $\lambda_i, i \in \{1 \dots 6\}$ are the loss weight, $\mathcal{L}_{\text{sem}}$ is the semantic

Table 1: **Quantitative comparison on Replica[52] and ScanNet++ [51].** We report accuracy (Acc) and completeness (Comp) in cm, others in percentage with a 5cm threshold. The first three results are highlighted in red, orange, and yellow, respectively.

Category	Methods	Replica [52]					ScanNet++ [51]				
		Acc↓	Comp↓	Prec↑	Recall↑	F-score↑	Acc↓	Comp↓	Prec↑	Recall↑	F-score↑
Implicit	ManhattanSDF [13]	4.76	5.59	68.80	66.40	67.57	3.96	4.98	77.30	76.16	76.67
	MonoSDF [14]	4.14	5.38	75.50	70.89	73.08	4.05	5.57	76.25	75.16	75.65
Explicit	Scaffold-GS [54]	8.58	11.27	63.53	54.91	58.89	23.11	15.36	28.18	31.75	29.78
	2DGS [10]	4.76	6.34	74.54	65.37	69.64	16.53	17.91	22.84	20.79	21.71
	DN-Splatter [44]	16.97	15.52	32.67	30.90	31.75	16.77	15.05	22.42	21.97	22.16
	GSRec [17]	4.90	6.89	73.32	67.69	70.37	9.37	9.13	47.12	53.81	50.14
	Ours	2.25	4.08	93.18	82.22	87.35	3.22	4.09	87.59	87.47	87.48

Table 2: **Quantitative comparison on ScanNet [50].** We report accuracy (Acc) and completeness (Comp) in cm, others in percentage with a 5cm threshold. The first three results are highlighted in red, orange, and yellow, respectively.

Category	Methods	Acc↓	Comp↓	Prec↑	Recall↑	F-score↑	Time ↓	FPS ↑
Implicit	ManhattanSDF [13]	4.25	5.23	72.39	63.18	67.25	> 7 h	< 10
	MonoSDF [14]	4.25	4.76	73.53	69.18	71.21	> 7 h	< 10
Explicit	Scaffold-GS [54]	9.47	7.99	51.41	49.08	50.17	12 mins	279
	2DGS [10]	11.46	13.89	43.15	36.17	39.27	11 mins	118
	DN-Splatter [44]	13.54	14.77	21.71	18.94	20.22	12 mins	145
	GSRec [17]	6.71	5.40	60.36	66.63	63.30	35 mins	261
	Ours	3.62	3.93	80.31	75.85	77.98	27 mins	70

loss mentioned in Sec. 3.2. To provide the smoothness of local surfaces, we incorporate monocular geometry priors from pre-trained models [8, 53] for indoor reconstruction during training.

$$\mathcal{L}_{\text{depth}} = \sum ||(wD + q) - \hat{D}||^2, \quad (11)$$

$$\mathcal{L}_{\text{normal}} = \sum \left(1 - \mathbf{N} \cdot \hat{\mathbf{N}}\right) + \left(1 - \mathbf{N}_d \cdot \hat{\mathbf{N}}\right), \quad (12)$$

where $\mathbf{N}$, $\hat{\mathbf{N}}$, $\hat{D}$ are the rendered normal, the normal prior and depth prior from [8, 53], respectively. w , q are the scale and shift computed from least-squares to align rendered depth D to monocular depth $\hat{D}$.

4 Experiments

Implementation Details. We implement our approach in PyTorch [55], incorporating a custom surfel rasterization module for semantic learning, and optimize the parameters with Adam optimizer [56]. The hyperparameter $\mathcal{K}$ is set to 10, and the voxel size is fixed at 0.01. During training, the loss weights $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5, \lambda_6$ are set to 0.25, 0.1, 0.1, 1.0, 100, and 0.05, respectively. The explicit plane indicator is initially derived from the semantic lifted SfM points and is reinitialized using the semantic Gaussians if the discrepancy exceeds a predefined threshold. Surfaces are extracted using TSDF Fusion [57]. All experiments are performed on a single NVIDIA 4090D GPU. Additional implementation details are provided in the supplementary material.

Datasets. Our method leverages the Atlanta world assumption, making it well-suited for both indoor and urban scenes. We evaluate our method using well-known datasets for both indoor and outdoor scene reconstruction. For indoor environments, we use ScanNet [50], ScanNet++ [51], and Replica [52]. For outdoor settings, we employ the MatrixCity [58] dataset for surface reconstruction. ScanNet and ScanNet++ feature large-scale RGB-D images and 3D surfaces akin to real-world scenarios, while Replica offers a synthetic dataset with high-quality images from 3D meshes. MatrixCity provides a synthetic, city-scale dataset for neural rendering. In line with previous studies [13, 14], we select four scenes from ScanNet, seven from Replica, and four from ScanNet++, sampling images

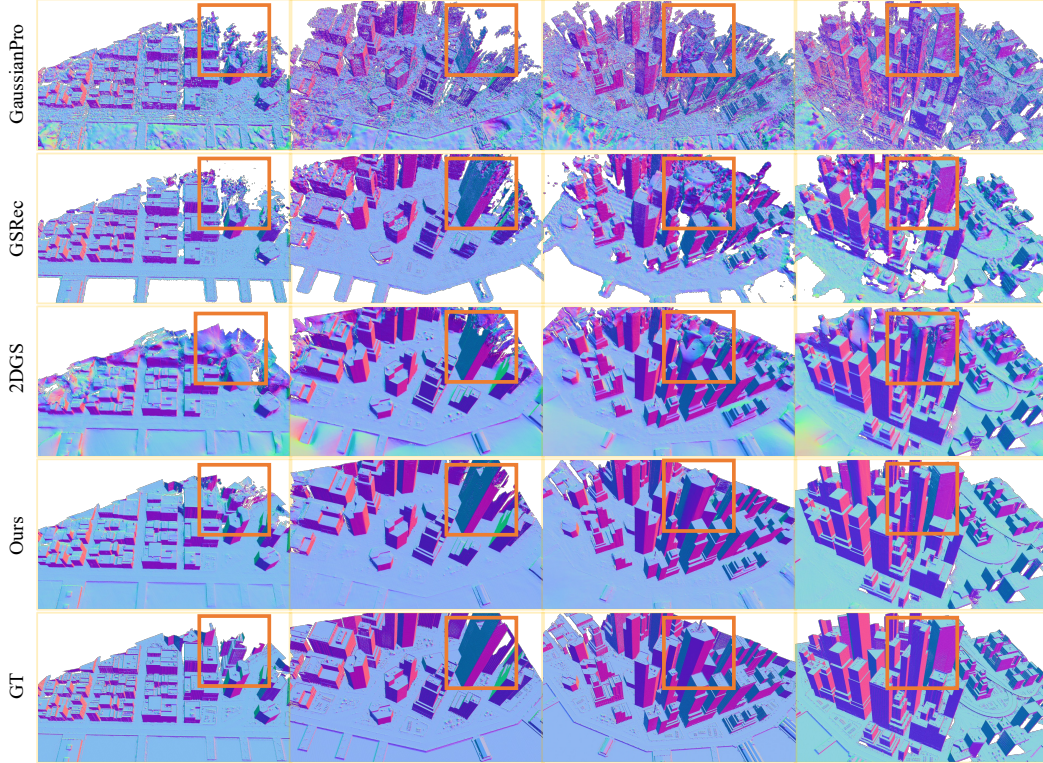


Figure 4: **Qualitative comparison of outdoor scene reconstruction.** As highlighted in the boxes, our approach can produce detailed and noise-free surfaces in textureless regions.

uniformly from the sequences. For outdoor evaluation, four city blocks from MatrixCity are used. Additional details are provided in the supplementary material.

Baselines. For indoor scenes, we compare our approach against two types of methods: 1) Neural implicit representations: ManhattanSDF [13] and MonoSDF [14]; 2) Gaussian-based representations: Scaffold-GS [54], 2DGS [10], DN-Splatter [44], and GSRec [17]. Additional supervision with monocular depth and normals, as noted in Eq (11) and Eq (12), is integrated into Scaffold-GS and 2DGS for indoor scenes. The quantitative evaluation for indoor scenes includes accuracy, completeness, precision, recall, and F-score. For outdoor scenes, comparisons are made with 2DGS, GSRec, Scaffold-GS, and GaussianPro [59]. In this context, the geometric evaluation relies on accuracy, completeness, and chamfer distance.

4.1 Comparison

Indoor Surface Reconstruction. We evaluate surface reconstruction on ScanNet [50], ScanNet++ [51], and Replica [52]. As shown in Tab. 1 and 2, our method outperforms both implicit and explicit baselines, achieving state-of-the-art results. SDF-based implicit methods like ManhattanSDF [13] and MonoSDF [14] ensure smooth surfaces but struggle with fine details, e.g., “lamp” in orange box of Fig. 3) and suffer from multiview inconsistency in textureless regions, e.g., there is a discontinuity on the floor in MonoSDF.

Table 3: **Quantitative comparison on Matrix-City [58].**

Methods	Acc↓	Comp↓	CD↓
GaussianPro [59]	0.102	0.081	0.091
Scaffold-GS [54]	0.328	0.303	0.316
GSRec [17]	0.048	0.175	0.112
2DGS [10]	0.115	0.098	0.106
Ours	0.022	0.034	0.028

Additionally, the time-consuming ray sampling in NeRF-based frameworks is to blame for the training time exceeding 7 hours and the inability to achieve real-time rendering. Gaussian-based explicit methods, such as Scaffold-GS and 2DGS, are fast but produce discontinuous surfaces due to

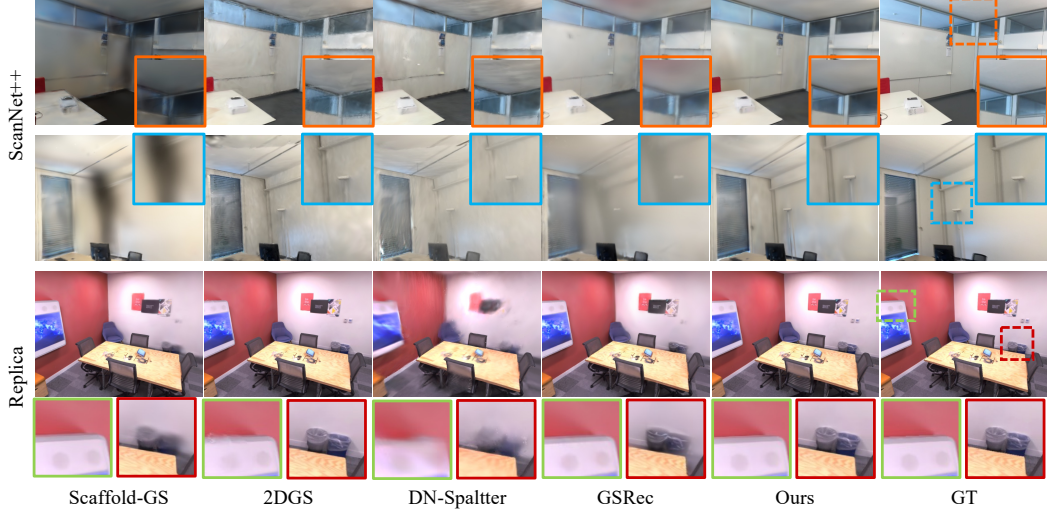


Figure 5: **Qualitative comparison of novel view synthesis.** We show the novel view synthesis results of different Gaussian splatting-based approaches on ScanNet++ [51] and Replica [52] datasets. Our method can obtain higher-fidelity rendering results with less noisy information than the baselines.

Table 4: **Quantitative comparison of novel view synthesis.** We perform experiments on Replica [52] and ScanNet++ [51] datasets.

Methods	Replica [52]			ScanNet++ [51]		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
ScaffoldGS [54]	38.08	0.9660	0.0961	18.25	0.7749	0.2764
2DGS [10]	41.59	0.9823	0.0464	21.87	0.8114	0.3060
DN-Splatter [44]	29.02	0.8967	0.2312	22.76	0.8226	0.2971
GSRec [17]	36.00	0.9574	0.1205	22.96	0.8314	0.2708
Ours	39.58	0.9756	0.0766	22.51	0.8321	0.2517

independent primitive optimization or view-dependent geometry. GSRec [17] improves geometry via IMLS but still yields noisy results. In contrast, our implicit-structured Gaussians combine locally coherent geometry with high-frequency detail preservation, enabling smoother and more accurate reconstructions. While slower than prior Gaussian methods due to decoding all Gaussians when rendering, our approach remains much faster than implicit ones and delivers superior quality.

Urban Surface Reconstruction. Structural priors are common in man-made environments, including both indoor scenes and urban buildings. To evaluate our method under such priors, we use the MatrixCity dataset [58]. We compare against GaussianPro [59], Scaffold-GS [54], GSRec [17], and 2DGS [10]. As shown in Tab. 3, our method yields more accurate and complete surfaces by leveraging the Atlanta world assumption. In Fig. 4, GaussianPro suffers from noisy surfaces and inconsistent depth despite normal propagation. GSRec reduces noise via IMLS regularization but produces sparse reconstructions in textureless regions, and its Poisson surface reconstruction trims low-density areas, resulting in missing geometry. 2DGS also struggles in textureless regions like the sea and building facades, leading to artifacts such as protrusions or holes. In contrast, our approach delivers smoother and more accurate surfaces in these challenging regions.

Novel View Synthesis. We evaluate novel view synthesis on the Replica [52] and ScanNet++ [51] datasets. As demonstrated in Tab. 4 and Fig. 5, our method achieves superior quantitative results, rendering photorealistic views with accurate geometry while avoiding the artifacts common in other approaches. While Scaffold-GS and 2DGS perform well on synthetic datasets without significant lighting variations, they struggle to render photorealistic novel views in real scenes. Scaffold-GS models significant lighting variations using view-dependent geometry, which can lead to overfitting in scenes with substantial lighting changes, such as those in [51]. This overfitting results in inaccurate lighting environment and geometry, as shown in Fig. 5. 2DGS [10] achieves a higher quantitative result on Replica with a discrete representation, however, the discrete representation exhibits protrud-

ing surfaces and results in noisy images on real scenes. GSRec [17] improves geometric accuracy but produces a blurry background and objects, lacking detailed modeling. In contrast, with its precise geometry, our method effectively models lighting variations across views while accurately capturing the appearance of the background and objects.

4.2 Ablation Study

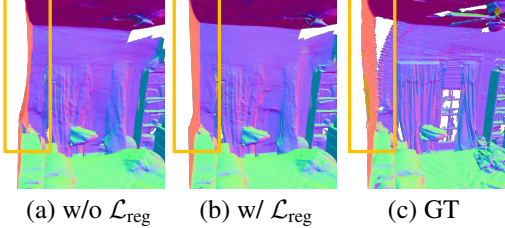


Figure 6: **Qualitative ablation on ScanNet [50].** (a) Ours w/o $\mathcal{L}_{\text{reg}}$ (Row b in Tab. 5); (b) Ours w/ $\mathcal{L}_{\text{reg}}$ (Row f in Tab. 5); (c) GT.

Table 5: **Quantitative ablation** on ScanNet [50].

Method	$\mathcal{L}_{\text{depth}}$	$\mathcal{L}_{\text{normal}}$	$\mathcal{L}_{3D}$	$\mathcal{L}_{2D}$	CD↓	F-score↑
a) 2DGS	✓	✓	✗	✗	12.68	39.27
b) Ours	✓	✓	✓	✓	3.77	77.98
c) w/o $\mathcal{L}_{2D}$	✓	✓	✓	✗	3.97	75.52
d) w/o $\mathcal{L}_{\text{reg}}$	✓	✓	✗	✗	4.10	74.23
e) w/o $\mathcal{L}_{\text{normal}}$	✓	✗	✓	✓	3.89	76.30
f) w/o $\mathcal{L}_{\text{depth}}$	✗	✓	✓	✓	4.23	74.22

We conduct an ablation study on the ScanNet [50] dataset to evaluate the contribution of each component in our method, including the implicit-structured Gaussian, depth and normal priors, as well as the proposed 3D global planar and 2D local surface regularization terms. We report geometric evaluation metrics including Chamfer Distance (CD) and F-score (5 cm threshold), as summarized in Tab. 5.

As shown in Tab. 5, each component contributes positively to surface reconstruction quality. Compared to 2DGS (Row a) which uses normal and depth priors, our method with implicit-structured Gaussians (Row d) significantly improves geometry, as it provides locally coherent structures and better leverages the guidance from geometry priors. By comparing Rows b, c, and d, we observe that our 3D global planar regularization and 2D local surface regularization enhance geometry quality and yield superior geometric metrics. These regularization terms ensure globally consistent geometric supervision under the Atlanta world assumption, mitigating the inconsistencies and inaccuracies inherent in depth and normal priors. Furthermore, Fig. 6 illustrates the geometric comparison of our structural regularization, demonstrating straighter wall regions. Additionally, we perform ablations on the geometry priors themselves. As seen in rows e and f, removing either the normal prior or the depth prior leads to noticeable degradation, confirming that both types of geometric supervision are critical for high-quality reconstruction. Our full model (Row b) integrates all components and achieves the best performance with the lowest CD and highest F-score.

5 Conclusion & Limitations

This work presents a novel framework, **AtlasGS**, which reconstructs structured scenes using implicit-structured Gaussians under the Atlanta world assumption. We introduce our implicit-structured Gaussian representation as a hybrid approach that provides locally coherent geometry via MLPs and preserves high-frequency details through explicit Gaussian representation. To resolve the global inconsistency of geometric priors, we propose 3D and 2D regularization strategies based on the Atlanta world assumption, effectively correcting textureless regions. As a result, our method achieves fast and accurate surface reconstruction, and extensive experiments demonstrate that it achieves state-of-the-art performance. However, there are also some limitations in our work. First, the training and rendering speed are slower than the previous Gaussian-based methods [54, 10, 39]. Second, our method is primarily based on the Atlanta world assumptions, which depend on a pretrained semantic segmentation model for a limited set of elements. The future extension of our work is to speed up the training and rendering speed and facilitate SAM [60] and geometry priors to provide planar information, which can provide better performance and broader applicability.

Acknowledgement

This work was partially supported by NSF of China (No. 62425209).

References

- [1] Xiaoliang Ju, Zhaoyang Huang, Yijin Li, Guofeng Zhang, Yu Qiao, and Hongsheng Li. Diffind-scene: Diffusion-based high-quality 3d indoor scene generation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4526–4535, 2024.
- [2] Shougao Zhang, Mengqi Zhou, Yuxi Wang, Chuanchen Luo, Rongyu Wang, Yiwei Li, Zhaoxiang Zhang, and Junran Peng. Cityx: Controllable procedural content generation for unbounded 3d cities. *arXiv preprint arXiv:2407.17572*, 2024.
- [3] Chong Bao, Yinda Zhang, Yuan Li, Xiyu Zhang, Bangbang Yang, Hujun Bao, Marc Pollefeys, Guofeng Zhang, and Zhaopeng Cui. Geneavatar: Generic expression-aware volumetric head avatar editing from a single image. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8952–8963, 2024.
- [4] Michal Adamkiewicz, Timothy Chen, Adam Caccavale, Rachel Gardner, Preston Culbertson, Jeannette Bohg, and Mac Schwager. Vision-only robot navigation in a neural radiance world. *IEEE Robotics and Automation Letters*, 7(2):4606–4613, 2022.
- [5] Obin Kwon, Jeongho Park, and Songhwai Oh. Renderable neural radiance map for visual navigation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9099–9108, 2023.
- [6] Timothy Chen, Ola Shorinwa, Joseph Bruno, Aiden Swann, Javier Yu, Weijia Zeng, Keiko Nagami, Philip Dames, and Mac Schwager. Splat-nav: Safe real-time robot navigation in gaussian splatting maps. *IEEE Transactions on Robotics*, 2025.
- [7] Hongjia Zhai, Xiyu Zhang, Boming Zhao, Hai Li, Yijia He, Zhaopeng Cui, Hujun Bao, and Guofeng Zhang. Splatloc: 3d gaussian splatting-based visual localization for augmented reality. *IEEE Transactions on Visualization and Computer Graphics*, 2025.
- [8] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2025.
- [9] James Coughlan and Alan L Yuille. The manhattan world assumption: Regularities in scene statistics which enable bayesian inference. *Advances in Neural Information Processing Systems*, 13, 2000.
- [10] 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.
- [11] 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. *Advances in Neural Information Processing Systems*, 34:27171–27183, 2021.
- [12] 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.
- [13] Haoyu Guo, Sida Peng, Haotong Lin, Qianqian Wang, Guofeng Zhang, Hujun Bao, and Xiaowei Zhou. Neural 3d scene reconstruction with the manhattan-world assumption. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5511–5520, 2022.
- [14] 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.

- [15] 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.
- [16] Xiaoyang Lyu, Yang-Tian Sun, Yi-Hua Huang, Xiuzhe Wu, Ziyi Yang, Yilun Chen, Jiangmiao Pang, and Xiaojuan Qi. 3dgsr: Implicit surface reconstruction with 3d gaussian splatting. *ACM Transactions on Graphics*, 43(6):1–12, 2024.
- [17] Qianyi Wu, Jianmin Zheng, and Jianfei Cai. Surface reconstruction from 3d gaussian splatting via local structural hints. In *European Conference on Computer Vision*, pages 441–458, 2024.
- [18] Grant Schindler and Frank Dellaert. Atlanta world: An expectation maximization framework for simultaneous low-level edge grouping and camera calibration in complex man-made environments. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*, volume 1, pages I–I, 2004.
- [19] 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. In *European Conference on Computer Vision*, pages 405–421, 2020.
- [20] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. Neuris: Neural reconstruction of indoor scenes using normal priors. In *European Conference on Computer Vision*, volume 13692, pages 139–155, 2022.
- [21] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022.
- [22] Jingwen Wang, Tymoteusz Bleja, and Lourdes Agapito. Go-surf: Neural feature grid optimization for fast, high-fidelity RGB-D surface reconstruction. In *International Conference on 3D Vision*, pages 433–442, 2022.
- [23] Zihan Zhu, Songyou Peng, Viktor Larsson, Zhaopeng Cui, Martin R. Oswald, Andreas Geiger, and Marc Pollefeys. Nicer-slam: Neural implicit scene encoding for rgb slam. In *International Conference on 3D Vision*, pages 42–52, 2024.
- [24] Xu Cao and Takafumi Taketomi. Supernormal: Neural surface reconstruction via multi-view normal integration. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20581–20590, 2024.
- [25] Ruibo Wang, Song Zhang, Ping Huang, Donghai Zhang, and Wei Yan. Semantic is enough: Only semantic information for nerf reconstruction. In *IEEE International Conference on Unmanned Systems*, pages 906–912, 2023.
- [26] Abhijit Kundu, Kyle Genova, Xiaoqi Yin, Alireza Fathi, Caroline Pantofaru, Leonidas J. Guibas, Andrea Tagliasacchi, Frank Dellaert, and Thomas A. Funkhouser. Panoptic neural fields: A semantic object-aware neural scene representation. In *IEEE/CVF on Computer Vision and Pattern Recognition*, pages 12861–12871, 2022.
- [27] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J. Davison. imap: Implicit mapping and positioning in real-time. In *IEEE/CVF International Conference on Computer Vision*, pages 6209–6218, 2021.
- [28] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12786–12796, 2022.
- [29] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*, 41(4):102:1–102:15, 2022.
- [30] Hengyi Wang, Jingwen Wang, and Lourdes Agapito. Co-slam: Joint coordinate and sparse parametric encodings for neural real-time SLAM. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13293–13302, 2023.

- [31] Hai Li, Xingrui Yang, Hongjia Zhai, Yuqian Liu, Hujun Bao, and Guofeng Zhang. Vox-surf: Voxel-based implicit surface representation. *IEEE Transactions on Visualization and Computer Graphics*, 30(3):1743–1755, 2024.
- [32] Tong Wu, Jiaqi Wang, Xingang Pan, Xudong Xu, Christian Theobalt, Ziwei Liu, and Dahua Lin. Voxurf: Voxel-based efficient and accurate neural surface reconstruction. In *International Conference on Learning Representations*, 2023.
- [33] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *Advances in Neural Information Processing Systems*, 2020.
- [34] Jonás Kulháněk and Torsten Sattler. Tetra-nerf: Representing neural radiance fields using tetrahedra. In *IEEE/CVF International Conference on Computer Vision*, pages 18412–18423, 2023.
- [35] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8280–8290, 2022.
- [36] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *Advances in Neural Information Processing Systems*, pages 6087–6101, 2021.
- [37] Radu Alexandru Rosu and Sven Behnke. Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.
- [38] Hongjia Zhai, Gan Huang, Qirui Hu, Guanglin Li, Hujun Bao, and Guofeng Zhang. NIS-SLAM: Neural implicit semantic RGB-D SLAM for 3D consistent scene understanding. *IEEE Transactions on Visualization and Computer Graphics*, 30(11):7129–7139, 2024.
- [39] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkuehler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023.
- [40] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5354–5363, 2024.
- [41] 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. *IEEE Transactions on Visualization and Computer Graphics*, 2024.
- [42] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 43(6):271:1–271:13, 2024.
- [43] Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using gaussian surfels. In *ACM SIGGRAPH Conference Papers*, page 22, 2024.
- [44] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splatter: Depth and normal priors for gaussian splatting and meshing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.
- [45] Haodong Xiang, Xinghui Li, Xiansong Lai, Wanting Zhang, Zhichao Liao, Kai Cheng, and Xueping Liu. Gaussianroom: Improving 3d gaussian splatting with sdf guidance and monocular cues for indoor scene reconstruction. *arXiv preprint arXiv:2405.19671*, 2024.
- [46] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995.

- [47] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.
- [48] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1280–1289.
- [49] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.
- [50] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2432–2443, 2017.
- [51] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023.
- [52] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard Newcombe. The Replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.
- [53] Chongjie Ye, Lingteng Qiu, Xiaodong Gu, Qi Zuo, Yushuang Wu, Zilong Dong, Liefeng Bo, Yuliang Xiu, and Xiaoguang Han. Stablenormal: Reducing diffusion variance for stable and sharp normal. *ACM Transactions on Graphics*, 43(6):1–18, 2024.
- [54] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20654–20664, 2024.
- [55] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.
- [56] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [57] Angela Dai, Matthias Nießner, Michael Zollhöfer, Shahram Izadi, and Christian Theobalt. BundleFusion: Real-time globally consistent 3D reconstruction using on-the-fly surface reintegration. *ACM Transactions on Graphics*, 36(3):24:1–24:18, 2017.
- [58] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *IEEE/CVF International Conference on Computer Vision*, pages 3205–3215, 2023.
- [59] Kai Cheng, Xiaoxiao Long, Kaizhi Yang, Yao Yao, Wei Yin, Yuexin Ma, Wenping Wang, and Xuejin Chen. Gaussianpro: 3d gaussian splatting with progressive propagation. In *International Conference on Machine Learning*, 2024.
- [60] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: We claim our contributions in the abstract and introduction.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: We discuss the limitations in the Sec. 5.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[NA\]](#)

Justification: The paper has no theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: The paper contains implementation details for reproducibility in Sec. 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We use official datasets in experiments, and code will be released.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We claims the data splits in supplementary.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We follow existing related works for the setting of error bars.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We have provided that in the Sec. 4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We have reviewed that and claim we conform that Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: There is no social impact of the work performed.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper does not have such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We have cited them in the references.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: We use and cite existing datasets in this work. Other assets including code/model will be released after submitting.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: We do not include such experiments.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: We do not include such experiments.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: We do not include such experiments.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.

A Appendix

A.1 Planar Indicator Initialization

As described in Sec. 3.3, we initialize the plane indicator using ceiling and floor points derived from semantically lifted SfM points or semantic Gaussian primitives. The SfM points are generated by triangulating posed images through 2D feature matching, establishing 2D-3D correspondences. Utilizing these correspondences, we aggregate semantic labels for each 3D point from 2D semantic maps across multiviews and apply a voting procedure to identify the most prevalent semantic label, including those for ceiling and floor. The Gaussian semantic lifting module, mentioned in Sec. 3.2, lifts 2D semantic maps to each Gaussian primitive, and each primitive contains a semantic probability of the wall, floor, ceiling, or other categories. Consequently, SfM points and Gaussian primitives are assigned structural semantic labels such as wall, floor, or ceiling, allowing us to extract ceiling and floor points.

Subsequently, we conduct plane fitting to identify the floor plane $(\mathbf{n}_f, d_f)$ using RANSAC [49] applied to the extracted floor points. The normal vector $\mathbf{n}_f$ is chosen as the gravity direction $\mathbf{n}_g$. The offset of the ceiling plane, d_c , is calculated based on the ceiling points and the gravity direction as follows:

$$d_c = \text{mean}_{\mathbf{p} \in \mathbf{P}_{\text{ceiling}}} (\mathbf{n}_g \cdot \mathbf{p}), \quad (13)$$

where $\mathbf{P}_{\text{ceiling}}$ represents the set of ceiling points. The plane indicator is initially determined using the semantic lifted points. If the angle deviation or the offset discrepancy surpasses a threshold, the plane indicator is reinitialized using semantic Gaussian primitives to minimize inaccuracies in textureless regions. Fig. 7 further illustrates plane indicators derived from both semantic lifted sparse points and semantic Gaussian primitives, demonstrating that both approaches can provide reliable structural priors.

A.2 Additional Implementation Details

Our implementation is based on PyTorch, utilizing customized surfel rasterization techniques for semantic learning. Parameters are optimized using the Adam optimizer. Most of the training learning rates are similar to those used in [54]. We set the hyperparameter $\mathcal{K}$ to 10 for indoor scenes and 5 for urban scenes, with a voxel size of 0.01, and the feature dim is 32 in our sparse feature grid. For all scenes, the implicit-structured Gaussian is trained for 40,000 steps. Voxels grow between steps 1,500 and 20,000, provided the gradients of the Gaussians exceed $2e-4$ and are pruned if the opacities of all local Gaussians fall below 0.005. During training, we start our 3D global planar regularization from step 7000 and 2D local surface regularization from 20000. After completing training, surfaces are extracted using TSDF-Fusion [57], following the approach described in [10].

A.3 Additional Experimental Details

Similar to previous works for indoor scene reconstruction [14], we select four scenes in ScanNet [50], including *scene0050_00*, *scene0084_00*, *scene0580_00*, *scene0616_00* and seven scenes

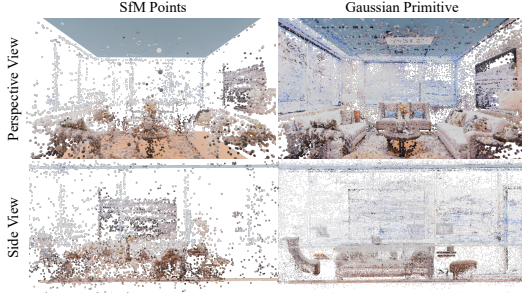


Figure 7: **Plane Indicator Visualization.** We visualize the plane indicators derived from both the semantic lifted SfM points and the semantic Gaussian primitives from both the perspective and side views. In the visualization, the ceiling plane is colored in blue, while the floor plane is colored in orange.

Table 6: **Defination of metrics.** P and P^* are the 3D points from the predicted and the GT mesh.

Metric	Definition
Acc	$\text{mean}_{\mathbf{p} \in P} (\min_{\mathbf{p}^* \in P^*} \ \mathbf{p} - \mathbf{p}^*\)$
Comp	$\text{mean}_{\mathbf{p}^* \in P^*} (\min_{\mathbf{p} \in P} \ \mathbf{p} - \mathbf{p}^*\)$
CD	$\frac{\text{Acc} + \text{Comp}}{2}$
Prec	$\text{mean}_{\mathbf{p} \in P} (\min_{\mathbf{p}^* \in P^*} \ \mathbf{p} - \mathbf{p}^*\ < 0.05)$
Recall	$\text{mean}_{\mathbf{p}^* \in P^*} (\min_{\mathbf{p} \in P} \ \mathbf{p} - \mathbf{p}^*\ < 0.05)$
F1-score	$\frac{2 \times \text{Prec} \times \text{Recall}}{\text{Prec} + \text{Recall}}$

in Replica [52], *office0~office3*, *room0~room2*, and as for ScanNet++ [51], we select four scenes, *8b5caf3398*, *b20a261fdf*, *f34d532901*, *f6659a3107*.

As described in Sec. 4, we uniformly sample images on the indoor scenes due to redundant images in the original dataset. For each scene in ScanNet [50] and Replica [52], we select one out of every 10 images in the original image sequence. For ScanNet++ [51], we use the image sequence from the iPhone and select one out of every 60 images. All the images are cropped and resized, and center-cropped to 640×480 . For MatrixCity [58], we use all the provided images and make the image resolution 960×540 . The SfM points are triangulated by COLMAP [47] with given images and corresponding poses.

A.4 Evaluation Metrics

Following previous methods [13, 14], we evaluate accuracy (Acc), completeness (Comp), Chamfer Distance (CD), precision (Prec), recall (Recall), and F1-score on ScanNet [50], ScanNet++ [51], and Replica [52]. Tab. 6 shows the definition of these metrics.

A.5 Additional Indoor Experiments

Semantic Segmentation. We evaluate the semantics from the rendered and the pre-trained segmentation model Mask2Former [48] on Replica [52] and ScanNet++ [51]. As shown in Tab. 7, ours achieves better results across all three classes on both datasets. By leveraging Gaussian semantic lifting, our model effectively aggregates multi-view information into 3D space and renders view-consistent semantic maps. In contrast, the 2D semantic segmentation model is more susceptible to image noise, leading to misclassifications, as illustrated in Fig. 8. The joint optimization scheme also helps correct semantic misclassifications, particularly around the boundaries between floors and walls.

Table 7: **Quantitative comparison of structural layout segmentation on Replica [52] and ScanNet++ [51] dataset.**

Methods	Replica [52]			ScanNet++ [51]		
	IoU _w ↑	IoU _f ↑	IoU _c ↑	IoU _w ↑	IoU _f ↑	IoU _c ↑
Mask2Former [48]	0.628	0.823	0.900	0.684	0.780	0.767
Ours	0.701	0.846	0.927	0.732	0.858	0.777



Figure 8: **Qualitative comparison of structural layout segmentation.**

A.6 Additional Qualitative Results

We present qualitative top-view results for ScanNet, ScanNet++, and Replica in Figs. 9 to 11, respectively. For additional comparisons, please refer to our accompanying video.

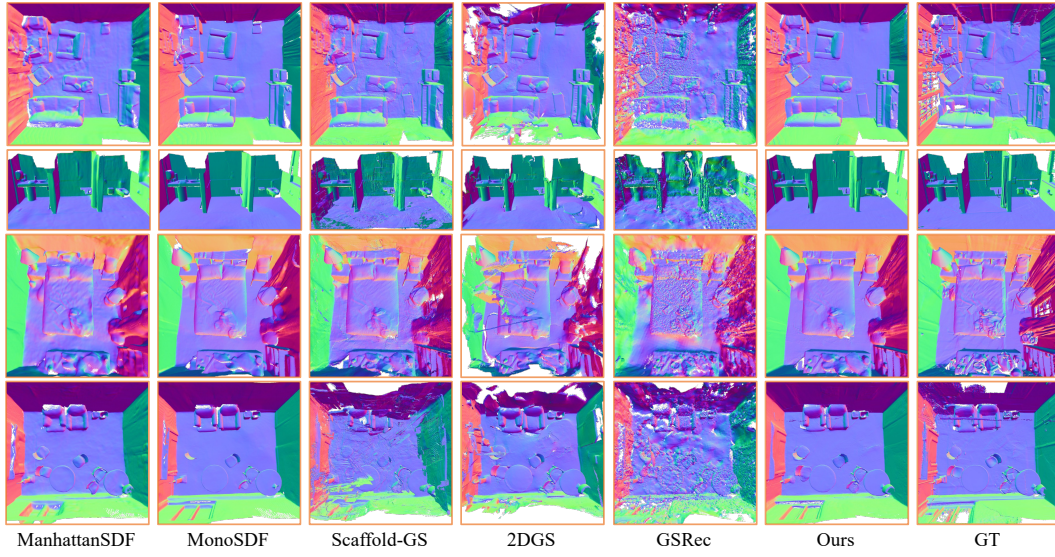


Figure 9: **Qualitative comparison of surface reconstruction on ScanNet [50].**

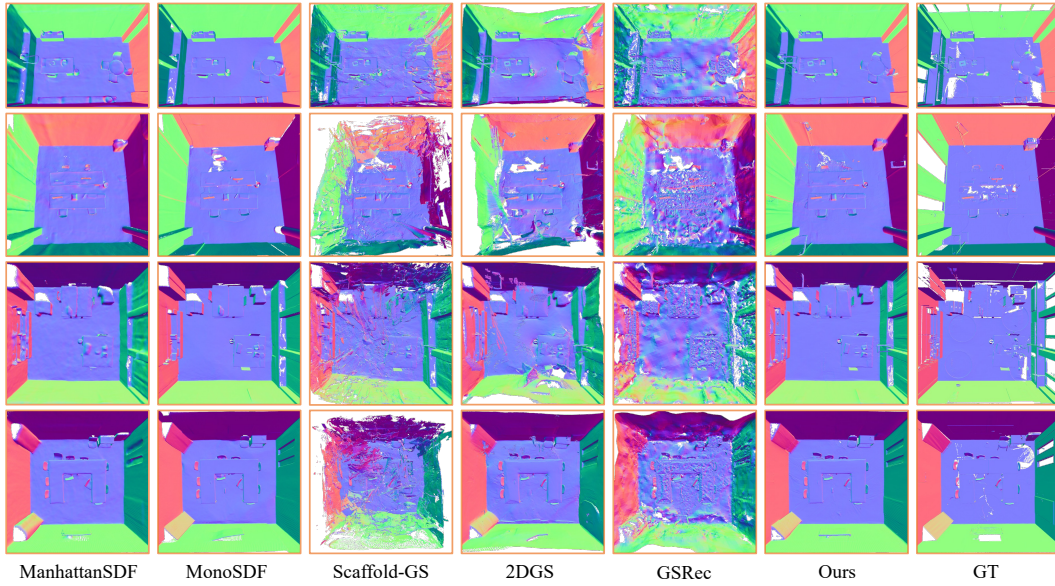


Figure 10: **Qualitative comparison of surface reconstruction on ScanNet++ [51].**

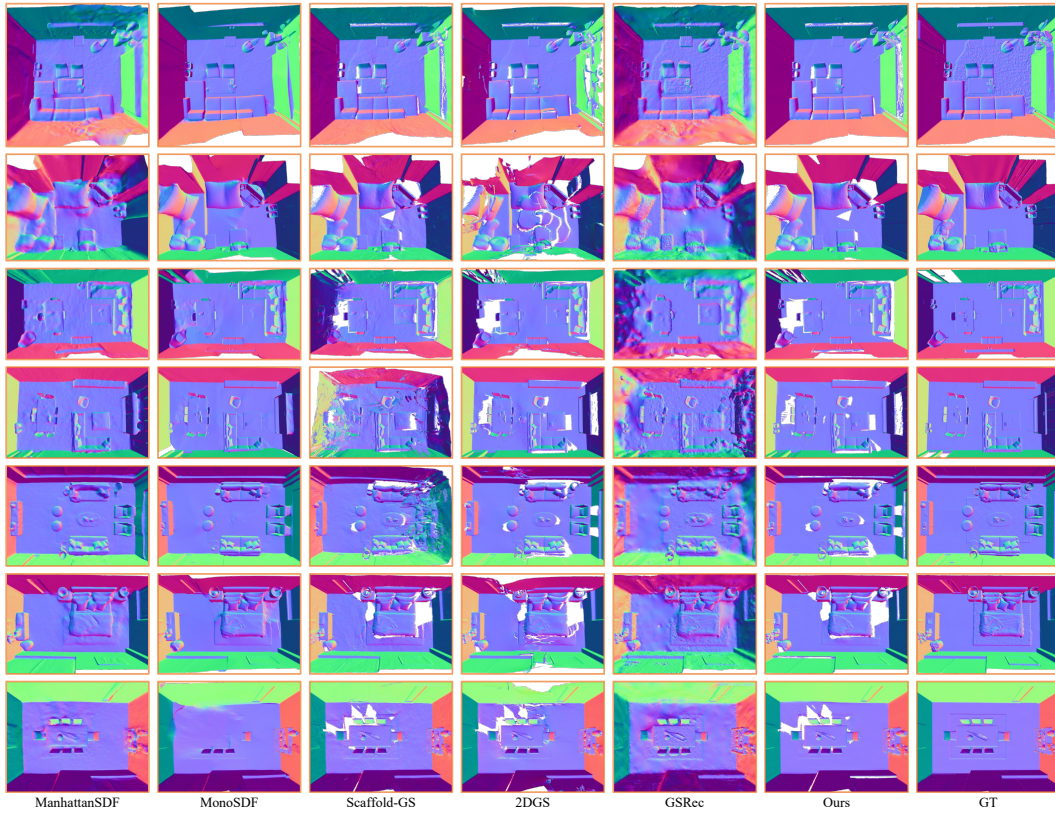


Figure 11: **Qualitative comparison of surface reconstruction on Replica [52].**