

# MMD-SLAM: Structure-Enhanced Multi-Meta Gaussian Distribution-Guided Visual SLAM

Fan Zhu<sup>1,2</sup>, Ziyu Chen<sup>1,2</sup>, Peichen Liu<sup>3</sup>, Yifan Zhao<sup>2</sup>, Zhisong Xu<sup>4</sup>,  
 Hui Zhu<sup>1,2</sup>, Hongxing Zhou<sup>5</sup>, Sixun Liu<sup>6</sup>, and Chunmao Jiang<sup>1,\*</sup>

**Abstract**—3D Gaussian Splatting (3DGS) has significantly boosted novel view synthesis and high-fidelity scene reconstruction, expanding the potential of 3DGS-based Visual Simultaneous Localization and Mapping (SLAM) methods. However, most existing systems fail to fully exploit the underlying structural information, which limits rendering quality and often leads to inconsistent maps. To address these limitations, we propose MMD-SLAM, a structure-enhanced Visual SLAM framework that leverages the Atlanta World (AW) assumption to guide a Multi-Meta Gaussian representation for photorealistic mapping. First, we introduce a point–line fusion strategy for pose optimization, where 3D line segments are incorporated to improve tracking robustness and provide additional constraints for mapping. Second, we design a Multi-Meta Gaussian representation with dominant directions, explicitly encoding structural priors from the AW hypothesis. Finally, we propose a Gaussian evolution strategy that adapts to scene geometry and incorporates structural cues into global optimization. Extensive experiments demonstrate that these innovations enable MMD-SLAM to achieve state-of-the-art performance in both tracking accuracy and mapping quality. e.g., our method achieves a 48.56% reduction in ATE RMSE on ScanNet and a 5.71% improvement in PSNR on Replica, compared with MonoGS.

## I. INTRODUCTION

Visual SLAM aims to estimate camera poses while simultaneously constructing sparse or dense maps of the environment. As a fundamental problem in 3D computer vision, Visual SLAM has been extensively applied in robotics, autonomous driving, and virtual reality (VR) [1]. The traditional Visual SLAM excels at localization, but its limited reconstruction capability falls short for Embodied Artificial Intelligence (Embodied AI). To address this gap, high-fidelity scene reconstruction with new perspective synthesis shows high potential in embodied perception and virtual agent training [2], [3]. Traditional visual SLAM systems primarily focus on camera pose tracking, while the mapping component often relies on sparse representations such as point clouds or voxels. Such low-fidelity maps, which contain holes and lack texture details, are increasingly inadequate for Augmented Reality (AR), VR, and Embodied AI. Consequently, research in Visual SLAM has been shifting toward photorealistic map

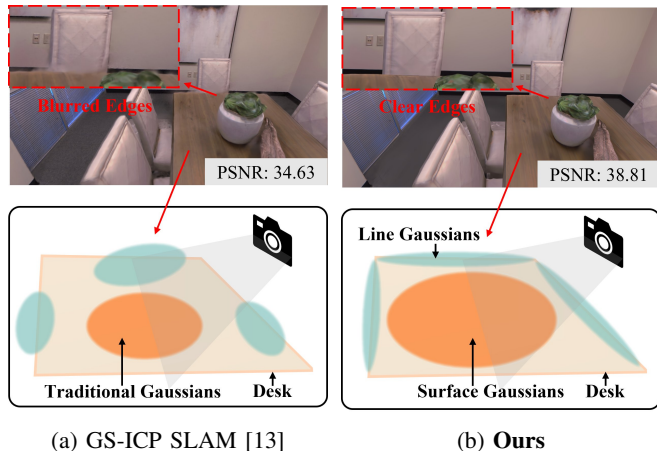


Fig. 1: **Comparison of mapping effect.** (a) illustrates that traditional Gaussian ellipsoids, which do not conform to the underlying structure, interfere with each other and generate blurred artifacts. (b) demonstrates that the Multi-Meta Gaussians used in our method better fit the scene structure after training, exhibiting clear edges.

reconstruction, which is essential for embodied perception and agent training.

The advent of Neural Radiance Fields (NeRF) [4] has introduced a novel approach to scene representation for robotics [5]. Integrating NeRF into the Visual SLAM significantly enhances the system’s high-fidelity reconstruction capabilities. Despite these advances, NeRF-based methods [6], [7] remain limited by over-smoothing and computational inefficiency.

3D Gaussian Splatting (3DGS) [8] is an explicit scene expression method that offers substantial improvements in both training speed and rendering quality. However, most 3DGS-based Visual SLAM methods [9]–[15] explore improvements in rendering quality, efficiency, semantics, and robustness, while ignoring the underlying structural information in the scene. With this in mind, MG-SLAM [16] leverages the Manhattan World assumption and extracts line features for scene optimization. However, its lack of effective integration between discrete features and Gaussian primitives constrains its adaptability. Furthermore, the anisotropic ellipsoid of 3DGS makes it difficult to reasonably describe these geometric structures, which limits the mapping quality of 3DGS-based methods.

To overcome the above limitations, we proposed MMD-SLAM, a structure-enhanced *Multi-Meta* Gaussian

\*This work was supported by China Postdoctoral Science Foundation under Grant 2025M781668; Anhui Province Key Research and Development Plan under Grant 202423k09020037

<sup>1</sup> HFIPS, Chinese Academy of Sciences, Hefei, China.  
<sup>2</sup> University of Science and Technology of China, Hefei, China.  
<sup>3</sup> Aarhus University, Aarhus, Denmark.  
<sup>4</sup> University of Tokyo, Tokyo, Japan.  
<sup>5</sup> Beijing University of Chemical Technology, Beijing, China.  
<sup>6</sup> North China Electric Power University, Beijing, China.  
 \* Corresponding author.

**Distribution-guided Visual SLAM.** Inspired by the Atlanta World (AW) assumption [17], which postulates that man-made indoor environments are dominated by multiple locally orthogonal directions, we leverage structural priors to improve both tracking and mapping. Specifically, we extract line segment features while extracting feature points in the tracking part, and jointly constrain pose estimation and perform global bundle adjustment optimization by reprojecting point features and back-projecting and reprojecting line features. On the other hand, in the mapping part, based on the AW hypothesis, we divide the 3D Gaussian distribution into three categories: points (spheres), lines, and surfaces, and introduce a dominant direction for this Multi-Meta Gaussian, which is very helpful for the accurate description of geometric structure. At the same time, we propose an evolutionary strategy for Multi-Meta Gaussian, further dividing the Gaussian primitives into Weak states that can flexibly convert types and Stable states that stably represent geometric structures, improving the multi-scene adaptability of our method. Finally, we propose a structure-enhanced scene optimization method. By combining the shape loss and dominant direction loss of Multi-Meta Gaussian on the basis of color and depth loss, we construct a complete loss function, further improving the overall scene reconstruction quality, as shown in Fig. 1.

Overall, the main contributions of our work are as follows:

- 1) We propose MMD-SLAM, a visual SLAM system that leverages the AW hypothesis to guide Multi-Meta Gaussians, leading to significantly improved pose estimation and photorealistic mapping quality. To our knowledge, this innovation is unique.
- 2) We propose a structurally enhanced multi-meta representation. By using the AW hypothesis, a set of Multi-Meta Gaussians with dominant directions was innovatively introduced, which can fully utilize the underlying scene structure extracted by the tracking part for mapping. Furthermore, a corresponding optimization strategy are introduced, significantly improving the mapping performance.
- 3) Extensive evaluation on both real-world and large-scale synthetic datasets shows that our method achieves state-of-the-art mapping and rendering quality while maintaining competitive tracking accuracy.

## II. RELATED WORKS

### A. Classical Dense Visual SLAM

Over the past decade, extensive work has used visual SLAM for dense scene reconstruction [18]. Traditional research uses explicit representations like point clouds, surfels, and Truncated Signed Distance Functions (TSDF) to reconstruct scenes [19]. For example, BundleFusion [20] estimates camera poses in real time from sparse features and uses TSDF for online mapping and map adjustment, achieving globally consistent 3D reconstruction. ElasticFusion [21] uses surfel sets for scene representation and periodically optimizes the map through non-rigid surface deformation.

DI-Fusion [22] integrates scene geometry with deep neural networks for incremental reconstruction. These traditional dense SLAM systems primarily focus on tracking and geometric reconstruction, while our approach also considers high-fidelity appearance reconstruction.

### B. Neural Implicit based SLAM

Following NeRF’s rise [4], integrating neural implicit representations into SLAM has attracted significant attention. iMAP [23] first demonstrated the potential of implicit representations in SLAM using a single multilayer perceptron (MLP). Subsequently, various works [5]–[7] explored novel implicit scene representations. For instance, NICE-SLAM [6] reconstructs large-scale scenes using hierarchical feature grids with pretrained MLPs. HS-SLAM [7] combines hash grids, tri-planes, and one-blob networks into a hybrid encoding scheme. Nevertheless, MLP-based implicit methods often suffer from over-smoothing and catastrophic forgetting, limiting their ability to capture fine-grained geometric details.

### C. 3D Gaussian Splatting based SLAM

The recent success of 3D Gaussian Splatting (3DGS) [8] has sparked interest in combining this efficient explicit representation with SLAM systems. Most approaches focus on improving rendering quality [9]–[11] and computational efficiency [12], [13], while others investigate semantic [24] and robustness [14], [25] enhancements. However, they generally overlook structural regularities in man-made environments, restricting both rendering quality and mapping consistency. MG-SLAM [16] addressed this limitation by incorporating structured line features into pose optimization and map refinement, improving reconstruction quality. Yet, it is constrained by the Manhattan World (MW) assumption [26], compromising rendering fidelity and adaptability. In contrast, our method generalizes the MW assumption to the Atlanta World (AW) assumption and introduces a set of multi-modal Gaussian primitives that enhance structural modeling while preserving high-quality appearance reconstruction.

## III. METHOD

Fig. 2 illustrates the overall pipeline of our approach, which is guided by the Atlanta World (AW) assumption. Section III-A describes the tracking module, where point–line constraints are incorporated to extract accurate geometric features, construct a sparse map, and perform global optimization, thereby providing reliable structural constraints for subsequent mapping. In Section III-B, we present the Structure-Enhanced Multi-Meta Gaussian representation as the underlying scene model. By leveraging Gaussian categorization and evolution, we enhance scene reconstruction, and further optimize the map using a combination of appearance and geometry-aware loss functions.

### A. Tracking of Point & Line Fusion Constraints

1) *Geometric primitives and notations:* To clearly describe our method, we first introduce the core mathematical notations and geometric primitives.

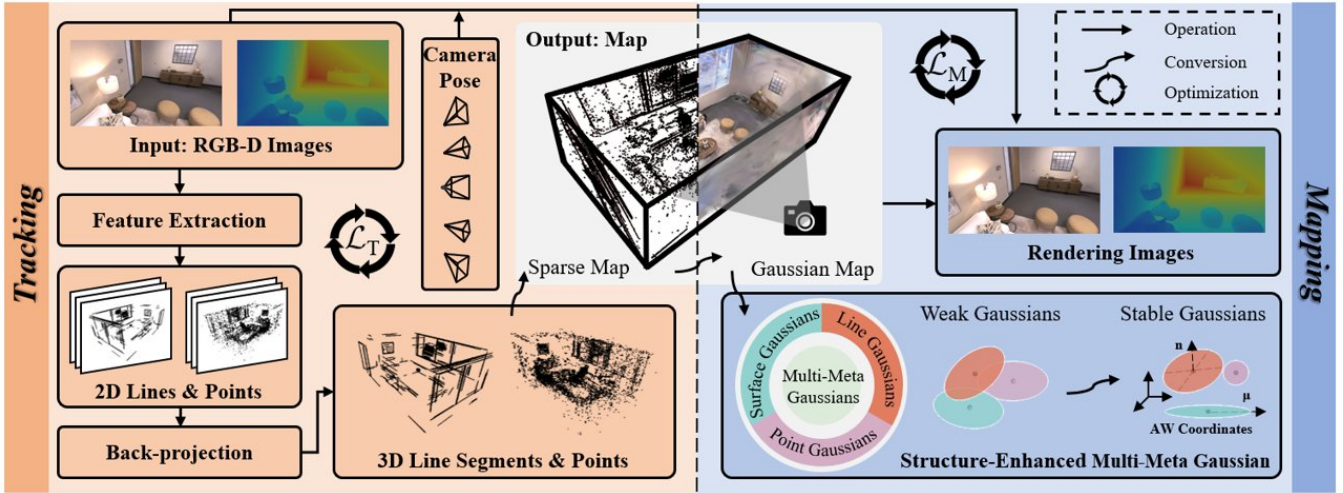


Fig. 2: **Overview.** MMD-SLAM consists of two components: Tracking and Mapping. **Tracking:** First, extracting point and line features from the input RGB-D frame, the camera pose is determined, and a sparse map is constructed. Secondly, the tracking process is optimized by minimizing the reprojection error and backprojection error. **Mapping:** Using accurate point cloud information to initialize a Weak Gaussian, a set of stable Multi-Meta Gaussian distributions that conform to the AW assumption is obtained, in order to precisely fit the scene.

**Camera:** Our system is based on the standard pinhole camera model, with intrinsic matrix  $\mathbf{K} \in \mathbb{R}^{3 \times 3}$  and pose  $(\mathbf{R}, \mathbf{t}) \in SE(3)$ .

**Image:** Each 2D image  $\mathbf{I}$  is associated with a depth map  $\mathbf{D}$ . A 2D point and a line segment in the image are denoted as  $\mathbf{p} = [\mu, \nu]^T \in \mathbb{R}^2$  and  $l = (\mathbf{p}, \mathbf{q})$ , respectively.

**3D Space:** In the 3D world domain, given a three-dimensional point  $\mathbf{P} \in \mathbb{R}^{3 \times 3}$ , we use the projection function  $\pi: \mathbb{R}^3 \rightarrow \mathbf{I}$  to project the three-dimensional point  $\mathbf{P}_i$  to its corresponding two-dimensional point  $\mathbf{p}_i$ . At the same time, there is a back-projection function  $\beta: \mathbf{I} \rightarrow \mathbb{R}^3$  that uses the depth map  $\mathbf{D}$  to back-project the 2D pixels to 3D points in the camera coordinate system. A 3D line segment is composed of a pair of 3D points  $(\mathbf{P}, \mathbf{Q}) \in \mathbb{R}^3 \times \mathbb{R}^3$  in the map.

2) *Point-based pose optimization:* Similar to [27], we extract point features from the 2D image and perform back-projection to optimize the camera pose. For the observation point  $\mathbf{p}_i$  in the key frame  $\mathbf{I}_t$ , there is a reprojection error:

$$\mathcal{L}_{2D}^p = \rho \left( \left\| \mathbf{p}_i - \pi_k(\mathbf{R}_k \mathbf{P}_i + \mathbf{t}_k) \right\|_{\Sigma_{2D}^p} \right) \quad (1)$$

where  $\Sigma_{2D}^p = \alpha_{p_i}^2 \mathbb{I}_2$  is the covariance matrix,  $\alpha_{p_i}^2 \in \mathbb{R}$  is the noise variance of feature points extracted at different scales in the Gaussian image pyramid [28], and we denote by  $\mathbb{I}_2$  the identity matrix of size  $2 \times 2$ .

For RGB-D cameras, we can use depth information to construct a reprojection error of a 3D point. For point  $\mathbf{p}_i$  in the  $k$ th keyframe, we have

$$\mathcal{L}_{3D}^p = \rho \left( \left\| \begin{bmatrix} \mathbf{p}_i \\ \mu_i - \frac{bf_x}{d_{p_i}} \end{bmatrix} - \begin{bmatrix} \pi_k(\mathbf{P}^W) \\ \frac{f_x(x^C - b)}{z^C} + c_x \end{bmatrix} \right\|_{\Sigma_{3D}^p} \right) \quad (2)$$

where, the superscripts  $W$  and  $C$  denote the world and

camera coordinate systems, respectively. The variable  $d$  represents the depth of point  $\mathbf{p}_i$  in image  $\mathbf{I}$ ;  $f_x$  is the focal length in the  $x$  direction;  $z^C$  is the  $z$ -coordinate of point  $\mathbf{P}^C = \mathbf{R}_k \mathbf{P}^W + \mathbf{t}_k$  in the camera frame; and  $c_x$  is the  $x$ -coordinate of the principal point.

3) *Line segment-based pose optimization:* Motivated by [28], we incorporate additional line constraints to enhance the robustness of the system. Line segments are efficiently detected at each level of the image pyramid using the EDLines method [29]. For line matching, we adopt the LBD descriptor [30], where correspondences are evaluated based on the Hamming distance between descriptors. For a 3D line segment  $L \in \mathbb{R}^3$ , its projection on the image plane is represented as  $l = (\mathbf{p}, \mathbf{q})$ . The back-projection of  $l$  is given as  $(\beta(\mathbf{p}), \beta(\mathbf{q}))$ . The distance between the camera projection of a 3D point  $\mathbf{P}^W$  and the 2D line  $l$  is denoted as  $d^l = \frac{\bar{\mathbf{p}} \times \bar{\mathbf{q}}}{\|\bar{\mathbf{p}} \times \bar{\mathbf{q}}\|}$ , where  $\bar{\mathbf{p}} = [\mathbf{p}^T, 1]$ ,  $\bar{\mathbf{q}} = [\mathbf{q}^T, 1]$ . Therefore, the 2D line segment reprojection error is defined as:

$$\mathcal{L}_{2D}^l = \rho \left( \left\| \begin{bmatrix} d_k^l \cdot \pi(\mathbf{R}_k \mathbf{P}^W + \mathbf{t}_k) \\ d_k^l \cdot \pi(\mathbf{R}_k \mathbf{Q}^W + \mathbf{t}_k) \end{bmatrix} \right\|_{\Sigma_{2D}^l} \right) \quad (3)$$

In addition, we exploit the available depth information to overcome the limitations of purely 2D reprojection errors by introducing constraints in 3D space. In the 3D map, the distance between an endpoint  $\mathbf{P}$  of a line segment and its corresponding back-projected point is defined as:

$$d_p^l(\beta(\mathbf{p}), \mathbf{P}^W) = \left\| \mathbf{P}^C - \beta(\mathbf{p}) \right\| \quad (4)$$

where  $\mathbf{P}^C$  is obtained from  $\mathbf{P}^C = \mathbf{R}_k \mathbf{P}^W + \mathbf{t}_k$ . This regularization term constrains the 3D map endpoints around the centroid of their observed back-projected points, preventing drift during optimization.

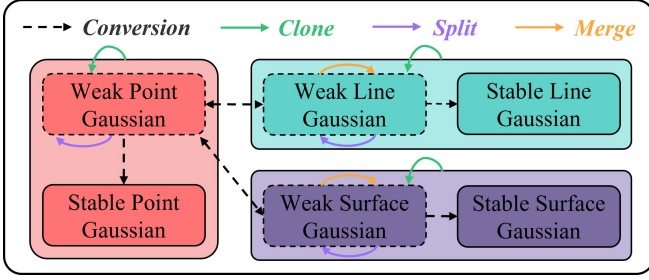


Fig. 3: **Multi-Meta Gaussian Evolution.** The transition from Weak to Stable states is achieved via adaptive type Conversion and density control (Clone, Split, and Merge), a process we term Multi-Meta Gaussian Evolution. Conversion refers to the operation between different types of Gaussians, while densification control is the operation between the same type of Gaussians.

Furthermore, we measure the perpendicular distance from  $P$  to its back-projected 3D line segment:  $L' = (\beta(p), \beta(q))$ , which enforces alignment between the geometric direction of the 3D line segment and that of its observed back-projection. This yields a 3D line back-projection error:

$$d_{3D}^l(L', P^W) = \frac{\|(\mathbf{P}^C - \beta(p)) \times (\mathbf{P}^C - \beta(q))\|}{\|\beta(p) \times \beta(q)\|} \quad (5)$$

This ensures that the geometric direction of the 3D line segment is aligned with the direction of the currently observed back-projection line segment, from which a back-projection error of the 3D line segment can be obtained:

$$\mathcal{L}_{3D}^l = \rho \left( \left\| \begin{array}{l} d_{3D}^l(L', P^W) + \omega d_p^l(\beta(p), P^W) \\ d_{3D}^l(L', Q^W) + \omega d_p^l(\beta(q), Q^W) \end{array} \right\|_{\Sigma_{3D}^l} \right) \quad (6)$$

where  $\omega \in [0, 1]$  is a scalar weighting factor.

4) *Global BA optimization:* Based on the point and line error functions described above, we formulate a global bundle adjustment to minimize the overall objective function as:

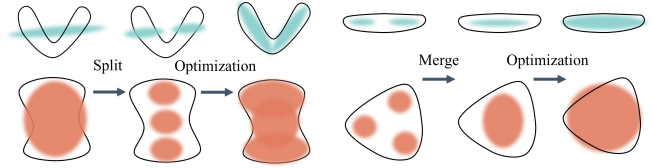
$$\mathcal{L}_T = \sum_{\mathcal{K}} \left( \sum_{\mathcal{P}_{2D}} \mathcal{L}_{2D}^p + \sum_{\mathcal{P}_{3D}} \mathcal{L}_{3D}^p + \sum_{\mathcal{L}_{2D}} \mathcal{L}_{2D}^l + \sum_{\mathcal{L}_{3D}} \mathcal{L}_{3D}^l \right) \quad (7)$$

where  $\mathcal{K}$  denotes the set of keyframes, and  $\mathcal{P}$  and  $\mathcal{L}$  represent the sets of points and lines extracted from a given keyframe  $k$ , respectively.

The optimization is carried out using the Levenberg–Marquardt (L-M) algorithm by iteratively solving the augmented normal as:

$$(\mathbf{J}^T \Sigma^{-1} \mathbf{J} + \lambda \mathbf{I}) \Delta \theta = -\mathbf{J}^T \Sigma^{-1} \varepsilon \quad (8)$$

where the terms  $\varepsilon$  and  $\Sigma$  correspond to the projection errors and covariance contributions in the global objective, respectively.



(a) Gaussian Split

(b) Gaussian Merge

Fig. 4: **Split and Merge.** We improved the original Split and added Merge to make the Density Control module more suitable for our system.

## B. Mapping of Structure-Enhanced Multi-Meta Gaussian

1) *Multi-Meta Gaussian Distribution Guided by AW Assumption:* For 3D Gaussian primitives  $\mathcal{G} = \delta N(\mu^W, \Sigma^W)$ , we largely follow the representation in [8]. Each Gaussian is parameterized by color  $\mathbf{c}$  derived from spherical harmonics  $\mathbf{SH} \in \mathbb{R}^{16}$ , opacity  $\delta \in [0, 1]$ , mean in world coordinates  $\mu^W$ , and covariance  $\Sigma^W \in \mathbb{R}^{3 \times 3}$  decomposed into scale  $\mathbf{s}$  and rotation  $\mathbf{R}$ .

To better fit structured 3D scenes, we draw inspiration from the Atlanta World assumption and extend traditional 3DGS [8] by introducing three types of Gaussian primitives with distinct geometric modalities: **Point Gaussian** ( $\mathcal{G}^P$ , **Sphere Gaussian**), **Line Gaussian** ( $\mathcal{G}^L$ ), and **Surface Gaussian** ( $\mathcal{G}^S$ ). Furthermore, each primitive is classified into either a *Stable* or *Weak* state depending on its reliability. For Weak-state Gaussians, we adopt the unified formulation as:

$$\mathcal{G}(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x} - \mu^W)^T (\Sigma^W)^{-1} (\mathbf{x} - \mu^W)\right) \quad (9)$$

For Stable-state Gaussians, the definitions follow three different patterns: Point Gaussian, Line Gaussian, and Surface Gaussian.

**Point Gaussian:** Similar to SplatTAM [9], we enforce isotropy on Point Gaussians. Their distribution is expressed as:

$$\mathcal{G}^P(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \mu^W\|}{2s^P}\right) \quad (10)$$

Comparing with the Eq. 9, Stable Point Gaussian has fewer learned parameters, which can significantly improve training speed.

**Line Gaussian:** We introduce line Gaussians to model structural edges and other linear features in the scene. For each primitive, among the scale parameters  $\mathbf{s}^L = [s_x^L, s_y^L, s_z^L]$ , the largest scale is defined as the *dominant* direction, while the other two scales are close to each other and much smaller. Let  $x$  denote the dominant direction, then:

$$\mathbf{s}^L = [s_{dir}^L, s_{dir}^L / \mathbf{n}^L, s_{dir}^L / \mathbf{n}^L] \quad (11)$$

**Surface Gaussian:** To represent large planar structures in 3D scenes, we define surface Gaussians. In this case, one axis with the smallest scale is defined as the dominant direction, while the remaining two scales are similar and relatively large. This configuration is analogous to the 2DGS

TABLE I: ATE RMSE (cm) ↓ results on TUM RGB-D [31] and ScanNet [32] datasets. The best results are marked as **best score** and **second best score**. The horizontal lines indicate that this method does not support this dataset.

Datasets	TUM RGB-D [31]				ScanNet [32]					
	fr1/desk	fr2/xyz	fr3/office	Avg.	0000	0059	0106	0169	0207	Avg.
SplaTAM [9]	3.34	1.34	5.21	3.30	12.57	10.15	17.78	12.65	7.98	12.23
MonoGS [11]	1.52	1.58	1.65	1.58	15.94	13.03	19.44	10.44	10.46	14.20
RTG-SLAM [12]	1.62	0.39	1.14	1.06	8.11	6.75	9.03	10.24	9.33	8.69
GS-ICP SLAM [13]	3.12	1.94	3.21	2.76	–	–	–	–	–	–
MG-SLAM [16]	–	–	–	–	5.95	6.41	8.07	7.29	6.14	6.77
<b>MMD-SLAM (Ours)</b>	1.55	0.61	1.12	1.09	5.82	6.52	7.85	6.99	6.51	6.74

representation introduced in [33], and its scale characteristics can be written as:

$$\mathbf{s}^S = [\mathbf{s}_{dir}^S, \mathbf{s}_{dir}^S \mathbf{n}^S, \mathbf{s}_{dir}^S \mathbf{n}^S] \quad (12)$$

2) *Multi-Meta Gaussian Evolution*: Fig. 3 illustrates the operations involved in Multi-Meta Gaussian Evolution. We initialize Gaussian primitives at the 3D points  $\mathbf{P}$  generated during tracking, with all primitives initially assigned to the Weak state. Through iterative updates, these Weak Gaussians gradually adapt to the surrounding 3D scene structure and are progressively classified into different Multi-Meta Gaussians.

**Conversion**: To obtain Stable Gaussians, we reclassify all Weak Gaussians every 1000 iterations and generate new Stable Gaussians. To evaluate this process, we introduce a Type Stability Score (TSS). Specifically, during each classification, if a Weak Gaussian  $\mathcal{G}_{W_i}$  retains the same type, its  $TSS_i$  increases by 1; otherwise, if a type conversion occurs,  $TSS_i$  is reset to 0. When a Gaussian maintains the same type for  $n$  consecutive classifications, i.e.,  $TSS_i = n$ , it is promoted to the Stable state, after which its geometric type remains unchanged.

**Density Control**: Our Clone method largely follows [8], with two additional modifications. First, the cloned Gaussian primitive is assigned the same type as the original one. Second, the cloned Gaussian is initialized in the Weak state. These modifications ensure that our approach remains adaptable to scene structures.

Meanwhile, we employ different types of Gaussians for Split and Merge operations to enhance the representation of complex structures, as illustrated in Fig. 4. During the Split process, an oversized Line Gaussian is divided along its dominant vector  $u$ , splitting the original Gaussian center  $\mu^W$  into two centers:

$$\mu_1^W = \mu^W - \frac{1}{2}\phi\mathbf{u}, \quad \mu_2^W = \mu^W + \frac{1}{2}\phi\mathbf{u} \quad (13)$$

where  $\phi$  is a scaling factor. The newly generated Gaussians are set to the Weak state, with their scales also controlled by  $\phi$ . Similarly, we split the oversized Surface Gaussian into three Weak state Gaussians. The Gaussian centers are distributed triangularly on a plane orthogonal to the principal direction  $\mu_i^W = \mu^W + \phi\mathbf{v}_i, i = 1, 2, 3$ , where  $\mathbf{v}_i$  denotes the in-plane offset vectors.

To reduce computational overhead, we introduce a Merge operation that fuses neighboring Gaussians of the same type

with small scales. Specifically, for merged Line Gaussians, the new dominant scale is defined as the sum of the distances between their centers. For Surface Gaussians, the two in-plane scales are set to the average of the pairwise distances among the three centers. All other attributes are assigned as the mean of the merged Gaussians.

3) *Structure-enhanced scene optimization*: We introduce a structure-enhanced scene optimization method to optimize the learnable parameters, enabling the Gaussian primitives to fully fit the three-dimensional information. Following 3DGS [8], we project a 3D Gaussian primitive  $\mathcal{G}_i(x)$  onto the 2D image to obtain  $\mathcal{G}'_i(x')$ . The projected Gaussians are then sorted by depth, and the appearance is rendered using  $\alpha$ -blending:

$$\mathcal{C}(\mathbf{x}') = \sum_{i \in N} \mathbf{c}_i \delta_i \mathcal{G}'_i(\mathbf{x}') \prod_{j=1}^{i-1} (1 - \delta_j \mathcal{G}'_j(\mathbf{x}')) \quad (14)$$

where  $N$  is the number of Gaussians corresponding to the pixel. We use the same method to gain the rendering depth:  $\mathcal{D}(\mathbf{x}')$ . Broadening this into a single image, and we use L1 loss to optimize:

$$\mathcal{L}_{rgb} = \|\mathcal{C} - \mathcal{C}^{GT}\|, \quad \mathcal{L}_{depth} = \|\mathcal{D} - \mathcal{D}^{GT}\| \quad (15)$$

Meanwhile, we introduce a shape constraint loss  $\mathcal{L}_{shape}$ . For a single Gaussian primitive  $\mathcal{G}_k(\mathbf{x})$  from the set of line or surface Gaussians  $k \in K$ , the scale components along different directions are sorted as  $s_{max} \geq s_{mid} \geq s_{min}$ . The shape error function is constructed by defining the elongation ratio  $e = s_{max}/s_{min}$  and flatness ratio  $f = s_{mid}/s_{min}$  of the Gaussian basis:

$$F_L(e, f) = \frac{1}{e} + (f - 1)^2, \quad F_S(e, f) = \frac{1}{f} + (e - 1)^2 \quad (16)$$

To obtain the shape constraint loss:

$$\mathcal{L}_{shape} = \frac{1}{K} \sum_{k=1}^K (\rho_k^L F_L(e, f) + \rho_k^S F_S(e, f)) \quad (17)$$

Additionally, using the AW assumption and 3D line segments extracted from the tracking part, we can guide the direction of the line Gaussian:

$$\mathcal{L}_{dir} = \frac{1}{H} \sum_{h=1}^H (1 - |\mathbf{u}_h \cdot \hat{\mathbf{t}}_k|) \quad (18)$$

TABLE II: **Quantitative evaluation of our method compared to SOTA methods on Replica datasets [34].** Colored cells indicate **best**, **second**, and **third** performance. FPS (f/s) represents the operating speed of the entire SLAM system.

Methods	Metrics	Room0	Room1	Room2	Office0	Office1	Office2	Office3	Office4	Avg.	FPS $\uparrow$
SplaTAM [9]	PSNR $\uparrow$	32.82	33.90	35.28	38.08	38.78	31.79	30.15	31.61	34.05	0.12
	SSIM $\uparrow$	0.978	0.970	0.984	0.981	0.982	0.965	0.951	0.946	0.970	
	LPIPS $\downarrow$	0.072	0.096	0.070	0.089	0.095	0.098	0.118	0.156	0.099	
MonoGS [11]	PSNR $\uparrow$	33.91	35.73	37.12	40.28	41.58	35.86	35.46	34.04	36.75	1.59
	SSIM $\uparrow$	0.946	0.957	0.964	0.972	0.976	0.962	0.958	0.941	0.960	
	LPIPS $\downarrow$	0.072	0.077	0.067	0.072	0.092	0.098	0.097	0.102	0.085	
RTG-SLAM [12]	PSNR $\uparrow$	30.35	33.68	34.48	39.08	39.21	32.65	32.39	35.57	34.68	7.14
	SSIM $\uparrow$	0.960	0.977	0.981	0.989	0.989	0.980	0.981	0.984	0.980	
	LPIPS $\downarrow$	0.156	0.115	0.122	0.082	0.102	0.143	0.138	0.123	0.123	
GS-ICP SLAM [13]	PSNR $\uparrow$	32.89	35.38	34.63	40.25	40.48	34.32	34.72	36.89	36.20	6.85
	SSIM $\uparrow$	0.943	0.958	0.957	0.976	0.975	0.962	0.957	0.963	0.961	
	LPIPS $\downarrow$	0.075	0.070	0.083	0.046	0.056	0.067	0.061	0.063	0.065	
MG-SLAM [16]	PSNR $\uparrow$	34.67	35.52	37.10	40.04	41.38	35.91	34.85	35.75	36.90	3.76
	SSIM $\uparrow$	0.976	0.978	0.980	0.987	0.988	0.980	0.977	0.978	0.981	
	LPIPS $\downarrow$	0.070	0.084	0.070	0.076	0.083	0.101	0.095	0.112	0.086	
MMD-SLAM (Ours)	PSNR $\uparrow$	35.94	38.27	38.81	42.59	41.98	37.51	36.92	38.78	38.85	3.82
	SSIM $\uparrow$	0.958	0.971	0.964	0.982	0.983	0.974	0.971	0.968	0.972	
	LPIPS $\downarrow$	0.031	0.028	0.033	0.018	0.044	0.036	0.027	0.042	0.032	

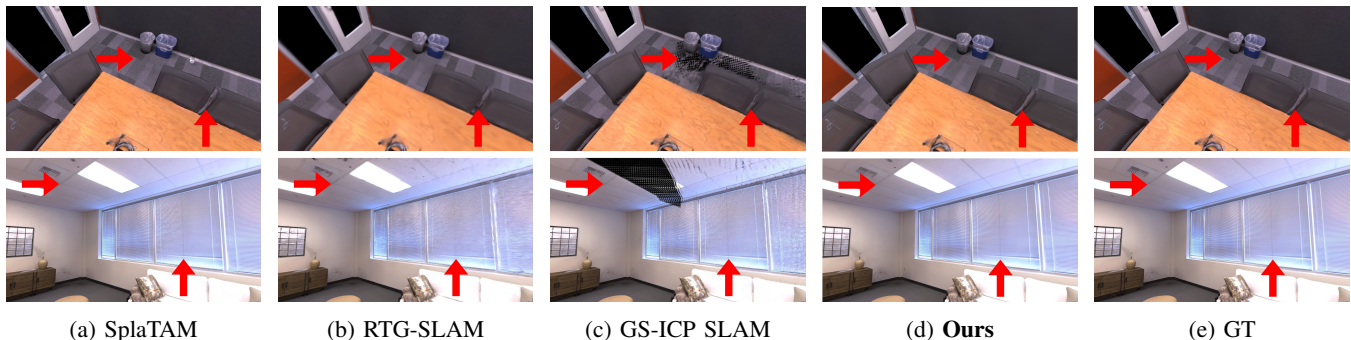


Fig. 5: **Rendering results on the Replica dataset [34].** The red arrow highlight the differences between our approach and baselines. For the details of ceiling, floor and window, our method significantly outperforms baselines.

where  $H$  is the number of line Gaussians,  $\mathbf{u}_h$  is the eigenvector of the dominant direction, and the reference direction provided by the line segment is  $\hat{\mathbf{t}}_k$ . Consequently, the total loss of this mapping module is

$$\mathcal{L}_M = \mathcal{L}_{rgb} + \lambda_{depth} \mathcal{L}_{depth} + \lambda_{shape} \mathcal{L}_{shape} + \lambda_{dir} \mathcal{L}_{dir} \quad (19)$$

where  $\lambda_{depth}$ ,  $\lambda_{shape}$ , and  $\lambda_{dir}$  are the weight parameters.

## IV. EXPERIMENT

### A. Experiment Setup

1) *Datasets*: Following [6], we evaluate our method and baselines on three widely used RGB-D datasets. These include the TUM RGB-D dataset [31] for pose evaluation, the larger and more challenging ScanNet dataset [32], and the synthetic Replica dataset for mapping evaluation. The Replica dataset version follows the setup of [6]. The selection of the sequence in the ScanNet dataset follows [16].

2) *Baselines*: To evaluate the tracking and mapping performance of our method, we compare against SplaTAM [9], MonoGS [11], RTG-SLAM [12], GS-ICP SLAM [13], and MG-SLAM [16], all of which represent state-of-the-art (SOTA) 3DGS-based SLAM systems.

3) *Metrics*: We follow the evaluation protocol in [11]. For pose estimation, we measure the root mean square error (RMSE) of the absolute trajectory error (ATE) across all frames. For photorealistic mapping, we report standard rendering quality metrics, including PSNR, SSIM, and LPIPS. We calculated the FPS of the entire system for real-time performance.

4) *Implementation Details*: Our method is implemented on an NVIDIA RTX A100 GPU (40 GB) and an AMD EPYC 7542 CPU. Except for the non-open-source MG-SLAM [16], all other baselines are run using their official implementations on the same hardware. To ensure fairness, we report the average results over five runs. For the tracking module,

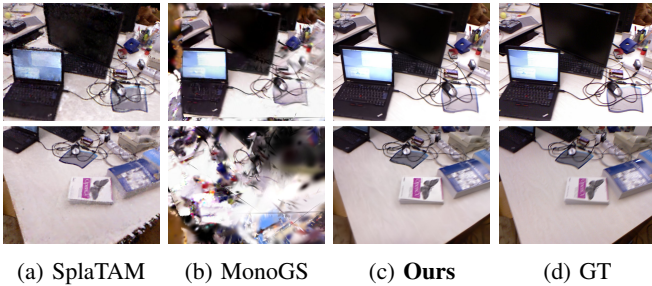


Fig. 6: **Rendering results on the TUM RGB-D dataset [31].** It can be clearly seen that our method can reconstruct a clearer appearance.

TABLE III: **Ablation study on the key components.** SOTA results are highlighted. The data is derived from the mean values of eight sequences in the Replica dataset [34].

PLFC	MMGE	SSO	ATE RMSE (cm) ↓	PSNR (dB) ↑
✗	✗	✗	0.61	35.92
✓	✗	✗	0.47	37.78
✓	✓	✗	0.47	38.31
✓	✓	✓	0.47	38.85

feature extraction parameters follow those used in [35], which is built upon [27]. For ScanNet, the ground-truth (GT) camera poses are obtained using BundleFusion [20]. The weight parameters of the mapping process are  $\lambda_{depth} = 0.1$ ,  $\lambda_{shape} = 0.2$ ,  $\lambda_{dir} = 0.2$ .

### B. Results Analysis

1) *Evaluation on Tracking:* As shown in TABLE I, we report comparisons between MMD-SLAM and SOTA baselines on the TUM RGB-D and ScanNet datasets. Our method achieves competitive pose estimation accuracy on both datasets. On the TUM dataset, which contains cluttered structures, [12] performs better by relying solely on point features for localization. In contrast, on the larger ScanNet dataset with abundant line features, our method together with [16] achieves the best results, owing to the similar point–line fusion tracking strategy adopted by both. Overall, when combining the results from both datasets, our method has the best overall positioning performance. This proves the effectiveness of the point-line dual constraint on pose.

2) *Evaluation on Mapping:* Since the original image quality of the TUM RGB-D and ScanNet datasets is relatively low, we perform quantitative mapping evaluation only on the Replica dataset. TABLE II reports the mapping results of our method compared with baselines, where MMD-SLAM significantly outperforms all competitors and has a competitive running speed. Moreover, to better demonstrate reconstruction quality, we conduct qualitative comparisons on the TUM RGB-D and Replica datasets, as shown in Fig. 6 and Fig. 5. In particular, the comparison on the TUM RGB-D dataset (Fig. 6) clearly shows that our method achieves the best rendering quality. Finally, as illustrated in Fig. 5, SplaTAM, which relies entirely on isotropic Gaussians, often produces holes in the map; RTG-SLAM suffers from generally poor

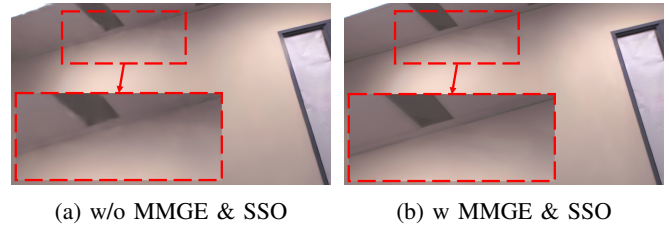


Fig. 7: **Ablation of MMGE & SSO.** The red dashed box highlights the differences between the two methods. Our comprehensive approach has more obvious structure.

rendering quality; and GS-ICP SLAM exhibits significant defects in reconstructed floors and ceilings. In contrast, we employed the Multi-Meta Gaussian model guided by the AW hypothesis, fully leveraging the structural information and achieving the best detail restoration.

3) *Ablation Study:* In order to demonstrate the role of key components in our system, we provide a comprehensive ablation analysis to demonstrate their importance.

**Point & Line Fusion Constraints (PLFC):** The first two rows of TABLE III present the ablation study on the tracking module with respect to Point & Line Fusion Constraints (PLFC). Row 1 corresponds to tracking using only point features, while Row 2 incorporates both point features and line segments. The results show that introducing additional line constraints plays a critical role in pose optimization and contributes to improved mapping quality.

**Multi-Meta Gaussian Evolution (MMGE):** To validate the effectiveness of the Multi-Meta Gaussian Evolution (MMGE) component in MMD-SLAM, we train a standard 3DGS model using the same poses. Row 2 reports the mapping results with single-modal Gaussian primitives, while Row 3 shows the results with our Multi-Meta Gaussian primitives under the same poses. This comparison demonstrates the effectiveness of the proposed Multi-Meta Gaussians.

**Structure-Enhanced Scene Optimization (SSO):** We combine appearance and geometric losses to design a Structure-Enhanced Scene Optimization (SSO) method. We conduct an ablation by controlling whether geometric losses are included. Row 4 in TABLE III corresponds to our full method, showing that each component contributes critically to the overall performance. Fig. 7 visualizes the effects of the key components of our method.

## V. CONCLUSION

In this work, we presented MMD-SLAM, a novel 3DGS based SLAM framework. Our method leverages point and line features as complementary constraints for robust pose estimation and mapping guidance. Guided by the Atlanta World hypothesis, we construct a set of Multi-Meta Gaussian distributions as the underlying scene representation, incorporating Gaussian evolution strategies and structure-enhanced optimization to accurately capture both appearance and geometry. Extensive experiments demonstrate that MMD-SLAM achieves SOTA performance. In future work,

we plan to extend the applicability of our approach and further enhance its robustness in challenging environments.

## REFERENCES

- [1] Changwei Wang, Shunpeng Chen, Yukun Song, Rongtao Xu, Zherui Zhang, Jiguang Zhang, Haoran Yang, Yu Zhang, Kexue Fu, Shide Du, et al. Focus on local: Finding reliable discriminative regions for visual place recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 7536–7544, 2025.
- [2] Yang Liu, Weixing Chen, Yongjie Bai, Xiaodan Liang, Guanbin Li, Wen Gao, and Liang Lin. Aligning cyber space with physical world: A comprehensive survey on embodied ai. *IEEE/ASME Transactions on Mechatronics*, 2025.
- [3] Zhenjun Zhao, Heng Yang, Bangyan Liao, Yingping Zeng, Shaocheng Yan, Yingdong Gu, Peidong Liu, Yi Zhou, Haoang Li, and Javier Civera. Advances in global solvers for 3d vision. *arXiv preprint arXiv:2602.14662*, 2026.
- [4] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2022.
- [5] Yunxuan Mao, Xuan Yu, Zhuqing Zhang, Kai Wang, Yue Wang, Rong Xiong, and Yiyi Liao. Ngel-slam: Neural implicit representation-based global consistent low-latency slam system. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6952–6958. IEEE, 2024.
- [6] Z. Zhu, S. Peng, V. Larsson, W. Xu, H. Bao, Z. Cui, M. R. Oswald, and M. Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12776–12786, 2022.
- [7] Ziren Gong, Fabio Tosi, Youmin Zhang, Stefano Mattoccia, and Matteo Poggi. Hs-slam: Hybrid representation with structural supervision for improved dense slam. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8464–8470. IEEE, 2025.
- [8] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023.
- [9] N. Keetha, J. Karhade, K. M. Jatavallabhula, G. Yang, S. Scherer, D. Ramanan, and J. Luiten. Splatam: Splat track & map 3d gaussians for dense rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21357–21366, 2024.
- [10] H. Huang, L. Li, H. Cheng, and S.-K. Yeung. Photo-slam: Real-time simultaneous localization and photorealistic mapping for monocular stereo and rgb-d cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21584–21593, 2024.
- [11] H. Matsuki, R. Murai, P. H. Kelly, and A. J. Davison. Gaussian splatting slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18039–18048, 2024.
- [12] Z. Peng, T. Shao, Y. Liu, J. Zhou, Y. Yang, J. Wang, and K. Zhou. Rtg-slam: Real-time 3d reconstruction at scale using gaussian splatting. In *ACM SIGGRAPH*, pages 1–11, 2024.
- [13] Seongbo Ha, Jiung Yeon, and Hyeonwoo Yu. Rgbd gs-icp slam. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 180–197. Springer, 2024.
- [14] Mingrui Li, Weijian Chen, Na Cheng, Jingyuan Xu, Dong Li, and Hongyu Wang. Garad-slam: 3d gaussian splatting for real-time anti dynamic slam. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11047–11053. IEEE, 2025.
- [15] Fan Zhu, Yifan Zhao, Ziyu Chen, Biao Yu, and Hui Zhu. Fgo-slam: Enhancing gaussian slam with globally consistent opacity radiance field. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11075–11081. IEEE, 2025.
- [16] Shuhong Liu, Tianchen Deng, Heng Zhou, Liuzhuozheng Li, Hongyu Wang, Danwei Wang, and Mingrui Li. Mg-slam: Structure gaussian splatting slam with manhattan world hypothesis. *IEEE Transactions on Automation Science and Engineering*, 22:17034–17049, 2025.
- [17] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages I–I. IEEE, 2004.
- [18] Zhenjun Zhao. Balf: Simple and efficient blur aware local feature detector. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3362–3372, 2024.
- [19] Shunpeng Chen, Changwei Wang, Rongtao Xu, Peixingtian, yukun Song, Jinzhou Lin, Wenhao Xu, jingyizhang, Li Guo, and Shibiao Xu. SAGE: Spatial-visual adaptive graph exploration for efficient visual place recognition. In *The Fourteenth International Conference on Learning Representations*, 2026.
- [20] 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(4):1, 2017.
- [21] Thomas Whelan, Stefan Leutenegger, Renato F Salas-Moreno, Ben Glocker, and Andrew J Davison. Elasticfusion: Dense slam without a pose graph. In *Robotics: Science and Systems (RSS)*, volume 11. Rome, 2015.
- [22] J. Huang, S. S. Huang, H. Song, and S. M. Hu. Di-fusion: Online implicit 3d reconstruction with deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8928–8937, 2021.
- [23] E. Suvar, S. K. Liu, J. Ortiz, and A. J. Davison. Imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6209–6218, 2021.
- [24] Mingrui Li, Shuhong Liu, Heng Zhou, Guohao Zhu, Na Cheng, Tianchen Deng, and Hongyu Wang. Sgs-slam: Semantic gaussian splatting for neural dense slam. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 163–179. Springer, 2024.
- [25] Mingrui Li, Dong Li, Sijia Hu, Kangxu Wang, Zhenjun Zhao, and Hongyu Wang. Slam-x: Generalizable dynamic removal for nerf and gaussian splatting slam. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 1132–1140, 2025.
- [26] Bangyan Liao, Zhenjun Zhao, Haoang Li, Yi Zhou, Yingping Zeng, Hao Li, and Peidong Liu. Convex relaxation for robust vanishing point estimation in manhattan world. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15823–15832, 2025.
- [27] C. Campos, R. Elvira, J. J. Gomez Rodriguez, J. M. M. Montiel, and J. D. Tardos. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6):1874–1890, 2021.
- [28] Luigi Freda. Plvs: A slam system with points, lines, volumetric mapping, and 3d incremental segmentation. *arXiv preprint arXiv:2309.10896*, 2023.
- [29] Cuneyt Akinlar and Cihan Topal. Edlines: A real-time line segment detector with a false detection control. *Pattern Recognition Letters*, 32(13):1633–1642, 2011.
- [30] Lilian Zhang and Reinhard Koch. An efficient and robust line segment matching approach based on lbd descriptor and pairwise geometric consistency. *Journal of visual communication and image representation*, 24(7):794–805, 2013.
- [31] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of rgb-d slam systems. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 573–580, 2012.
- [32] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5828–5839, 2017.
- [33] B. Huang, Z. Yu, A. Chen, A. Geiger, and S. Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH*, pages 1–11, 2024.
- [34] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, and S. Verma. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019.
- [35] C.-M. Chung, Y.-C. Tseng, Y.-C. Hsu, X.-Q. Shi, Y.-H. Hua, J.-F. Yeh, W.-C. Chen, Y.-T. Chen, and W. H. Hsu. Orbeez-slam: A real-time monocular visual slam with orb features and nerf-realized mapping. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9400–9406. IEEE, 2023.