2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

GauSem-SLAM: Gaussian Semantic Submaps with Loop Closure for Globally Consistent SLAM

Bowen Zhang†, Yufan Liu†, Lebin Liang, Dong Li, Mingrui Li, and Xuanxuan Zhang*

***Abstract*— 3DGS has shown outstanding performance in multi-view geometry, driving its adoption in visual SLAM. However, real-time semantic 3DGS mapping faces challenges. Current methods typically treat semantics as external priors, making it hard to integrate them into SLAM tracking or loop closure correction. Moreover, traditional semantic SLAM corrects accumulated drift by applying rigid adjustments to dense point clouds, which is costly for 3DGS maps and limits loop closure performance. We propose GauSem-SLAM, which uses a Gaussian semantic submap representation with a progressive allocation strategy, integrating semantics into tracking, mapping, loop detection, and submap management. We fully exploit semantic information by designing a robust loop detection module that combines DINOv2 semantic features with semantic landmarks. Furthermore, we introduce Semantic-Guided Registration (SGR), a method for computing inter-submap loop constraints. Through intra-submap and inter-submap loop correction, followed by a two-stage global map refinement, our system achieves globally consistent pose estimation and mapping. Experiments on three public datasets demonstrate that our method outperforms prior methods in both tracking and mapping.**

I. Introduction

Visual SLAM is crucial for real-time scene perception in robotics, and building semantic maps is important for embodied AI. With the introduction of NeRF [1] and 3DGS [2] into SLAM, real-time reconstruction with fine textures and photorealistic quality becomes possible. However, geometric maps lack scene understanding, while semantic annotations enable optimization of radiance fields or Gaussians for more accurate geometry and rendering.

Although existing semantic SLAM [3], [4], [5] has made some progress, they all lack effective loop closure, which results in global semantic inconsistency. For 3DGS SLAM, Photo-SLAM [6] incorporates the loop closure from ORB-SLAM3 [7], however, they do not perform loop closure correction on Gaussian primitives, resulting in degraded map quality. LoopSplat [8] performs Gaussian-based loop closure through submap registration and pose graph optimization, but its bidirectional keyframe localization with iterative optimization reduces efficiency and limits integration into semantic SLAM. We argue that semantic SLAM should tightly couple loop closure with joint correction of poses and Gaussians to achieve globally consistent tracking and mapping. To this end, we propose GauSem-SLAM. Our method constructs semantic submaps and introduces a spatial-semantic submap allocation strategy to prevent geometric discontinuities of objects at submap boundaries, as shown in Fig. 1. We embed semantic information into Gaussian primitives and tightly couple them with the tracking and mapping system, enabling optimization via local submap bundle adjustment. Furthermore, we introduce a Semantic-Aware Loop Closure, including loop detection based on semantic features and semantic landmarks and both intra-submap and inter-submap loop corrections. To accurately compute inter-submap loop closure constraints, we propose Semantic-Guided Registration (SGR), an efficient and robust registration method for Gaussian submaps. Our main contributions are summarized as follows:

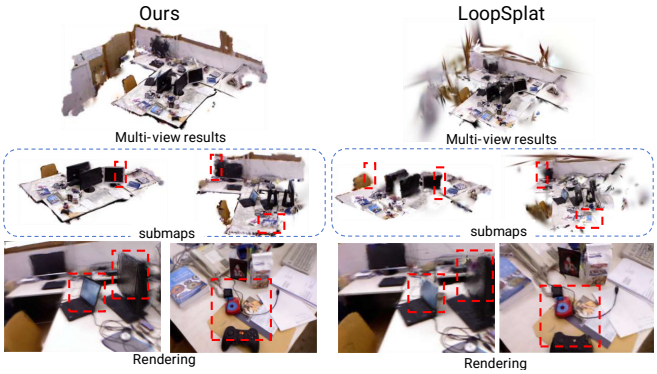

Fig. 1: **Reconstruction results on TUM datasets fr1-desk sequence.** Compared with LoopSplat [8], which assigns submaps based on displacement and rotation thresholds, we incorporate both spatial and semantic information to determine whether a new submap should be created. This effectively prevents geometric fragmentation of objects at submap boundaries and further improves rendering quality.

- We propose GauSem-SLAM, our framework integrates semantic submap construction, tracking and mapping, loop closure, as well as global map refinement into a complete semantic SLAM pipeline.
- We embed semantic information into Gaussians and design a semantic submap management strategy, including spatial-semantic submap allocation, semantic sampling submap initialization, and submap registration.
- We introduce the Semantic-Aware Loop Closure that incorporates 2D semantic map features and 3D semantic landmarks for loop detection and both intra-submap and inter-submap correction strategies. Furthermore, we pro-

†Equal contribution. *Corresponding author. Bowen Zhang is with Hebei University, China. zbw246244@gmail.com. Yufan Liu is with the University of California, Berkeley. Lebin Liang is with the University of Chinese Academy of Sciences, Beijing, China. Dong Li is with the Faculty of Innovation Engineering, Macau University of Science and Technology, Macau, and also with WAYTOUS Inc., Beijing, China. doongli@ieee.org. Mingrui Li is with the Dalian University of Technology, China. 2905450254@mail.dlut.edu.cn. Xuanxuan Zhang is with the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, China. xuanxuanzhang@whu.edu.cn.

pose Semantic-Guided Registration (SGR) for efficient Gaussian submap registration, substantially improving the computation efficiency of loop closure constraints. Finally, we design a two-stage global map refinement and fusion to achieve globally consistent mapping.

II. Related Works

A. Semantic SLAM

Visual SLAM increasingly incorporates semantic mapping to enhance environmental understanding, which is crucial for robotic applications. With the introduction of NeRF, several approaches have combined semantics with NeRF-based SLAM. SNI-SLAM [5] achieves top-down structured semantic mapping through multi-level semantic understanding, leveraging cross-attention to mine inter-attribute relationships, albeit at the cost of higher computational complexity. NIS-SLAM [9] incorporates a pretrained 2D segmentation network and designs a semantic fusion strategy to achieve consistent semantic representation across multiple views. However, NeRF-based SLAM [10] requires extensive sampling and MLP training, limiting real-time capability. The emergence of 3DGS has improved inference efficiency through fast rasterization. SGS-SLAM [11] integrates appearance, geometry, and semantic features to address the over-smoothing issue in NeRF-based SLAM rendering. Hier-SLAM [4] proposes a novel hierarchical representation and leverages the capabilities of large language models to encode semantic information compactly into 3D Gaussian splats. Although the above semantic SLAM methods have achieved some success in semantic segmentation and spatial understanding, they generally lack effective loop closure modules. However, loop closure is equally important for semantic SLAM, this omission leads to accumulated drift in tracking and mapping and causes globally inconsistent semantic mapping. We propose Semantic-Aware Loop Closure that performs both intra-submap and inter-submap loop correction to achieve globally consistent tracking and mapping.

B. Loop Closure in SLAM

Loop closure plays a vital role in improving localization accuracy by recognizing previously visited locations and providing global constraints. Traditional approaches such as ORB-SLAM [7] rely on hand-crafted descriptors and the Bag-of-Words model, a strategy also adopted by Photo-SLAM [6], and NGEL-SLAM [12]. However, BoW requires substantial data and clustering computation, and ORB descriptors are not robust under sparse and repetitive textures, which can lead to incorrect loop closures. Other approaches detect loops through co-visibility. For example, MIPSFusion [13] introduces implicit submaps and computes co-visibility between the current frame and inactive submaps to correct submap loops. However, co-visibility is sensitive to trajectory drift. Spatial misalignment of observed features at the same location can cause co-visibility checks to fail, making it effective only for small-scale loop correction. LoopSplat [8] uses NetVLAD for loop detection, but appearance only scene recognition is not robust, texture sparsity and illumination changes can cause feature degradation, while repetitive textures and structurally similar regions can yield overly similar features. We design a coarse-to-fine loop detection pipeline: first, DINOv2 is used to extract semantic features to assess similarity, then, we propose a semantic landmark scoring method to fully leverage the constructed semantic map for loop frame selection. Compared to LoopSplat, the integration of semantics improves robustness to texture and lighting variations.

III. Method

We propose GauSem-SLAM, a complete SLAM framework for semantic submaps with loop closure. The overall architecture of our method is illustrated in Fig. 2. Section III-A introduces the construction of 3DGS semantic submaps as well as joint tracking and mapping. Subsequently, Section III-B presents Semantic-Guided Registration (SGR), which aligns Gaussian submaps that form loop closures. Finally, Section III-C describes Semantic-Aware Loop Closure, which encompasses loop detection and the correction of both intra-submap and inter-submap loop closures.

A. Semantic 3DGS Submaps for Joint Tracking and Mapping

Unlike LoopSplat, we incorporate semantics into submap representation to prevent geometric fragmentation from partitioning, ensuring artifact-free rendering at boundaries. We represent the scene as a set of semantic 3DGS submaps $\mathcal{M}_i = \{\mathcal{G}_i, \mathcal{K}_i, T_i\}$, where $\mathcal{G}_i$ is the set of isotropic Gaussian kernels within the submap, each defined by opacity $\sigma_j \in [0,1]$, center $\mu_j \in \mathbb{R}^3$, radius $r_j \in \mathbb{R}$, RGB color $c_j \in \mathbb{R}^3$, and semantic color $s_j \in \mathbb{R}^3$. $\mathcal{K}_i$ denotes the keyframes associated with the submap, and $T_i \in \mathbb{R}^{4\times4}$ is its global pose, anchoring the local coordinate system at the submap's first frame.

We perform frame-to-model pose estimation for submap tracking. The current frame pose is initialized using a constant-velocity model and optimized within the submap coordinate system. Following 3DGS rasterization, the submap's Gaussian kernels are projected and rendered:

$$C_{\text{pix}} = \sum_{i=1}^{N} c_i \, \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) \tag{1}$$

with α_i computed from σ and its projected footprint. Depth d_i and semantic color s_i are similarly rendered to obtain D_{pix} and S_{pix}. Pose is optimized via an ℓ_1 loss against ground truth while keeping Gaussian kernels fixed:

$$\mathcal{L}_{\text{tracking}} = \sum_{\text{pix}} (S_{\text{sil}} > 0.99) \Big(\lambda_C \, |C_{\text{pix}} - C| + \lambda_S \, |S_{\text{pix}} - S| + \lambda_D \, |D_{\text{pix}} - D| \Big) \tag{2}$$

C and D are the input RGB-D images, while S represents the segmentation map, which is produced by employing DINOv2 [14] as a feature extractor followed by a segmentation head. S_{sil} is rendered using Eq. (1) with c_i removed, ensuring optimization focuses on well-reconstructed visible regions. For submap initialization, we adopt a semantic-based sampling strategy during initialization. Pixels from

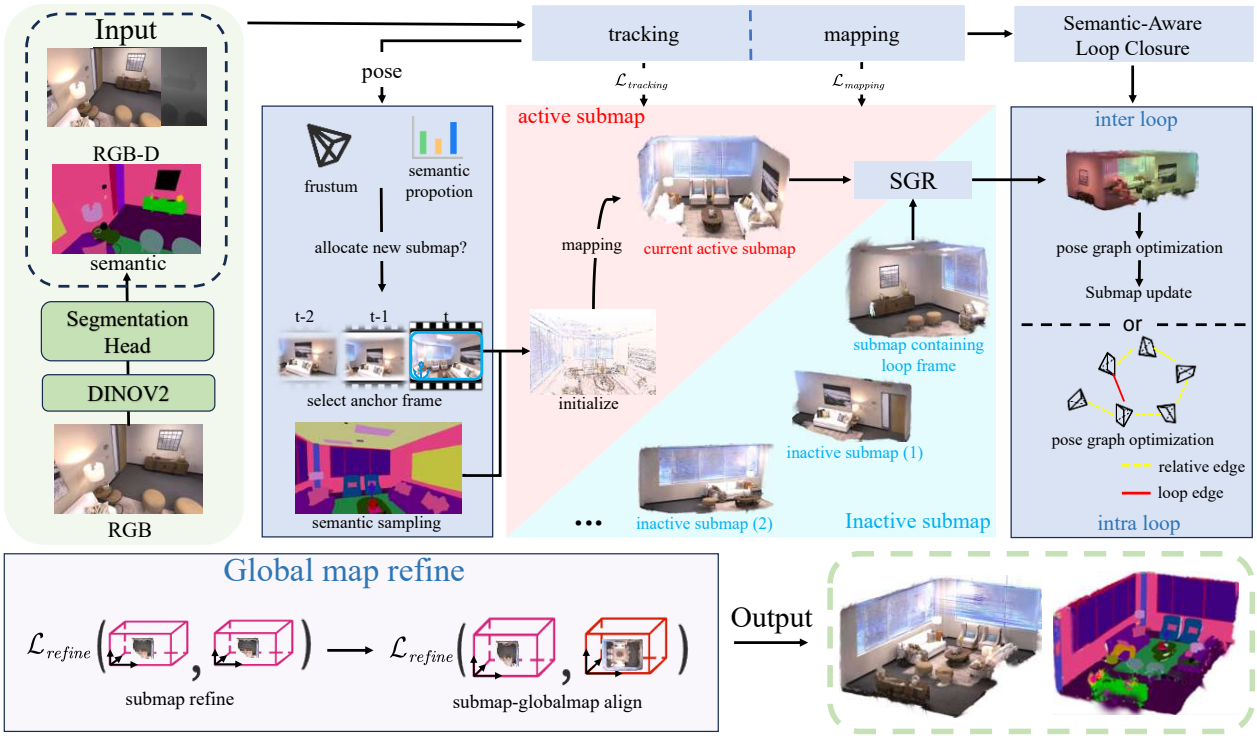

Fig. 2: **System Overview.** Our system takes RGB-D and segmentation map obtained via DINOv2 with a segmentation head as input and mainly consists of three modules: tracking and mapping, loop closure, and global map refinement. We categorize submaps into active (under tracking and mapping) and inactive (waiting for loop closure). After tracking each frame, we perform spatial-semantic submap allocation: if a new submap is created, it is initialized via semantic sampling, and the previous active submap becomes inactive. Each frame is then processed by Semantic-Aware Loop Closure to determine intra- or inter-loop. Intra-loops trigger pose graph optimization on the active submap, while inter-loops use Semantic-Guided Registration to align active submap and inactive submap containing loop frame, correcting poses and Gaussian primitives. Finally, Global Map Refinement is applied: first updating submaps with $\mathcal{L}_{\text{refine}}$, then aligning them with the global map for fusion.

each semantic category are sampled proportionally according to $n_c = \left\lfloor N_{tot} \cdot \frac{A_c^\gamma}{\sum_k A_k^\gamma} \right\rfloor$, with N_{tot} as the image area and $\gamma = 0.5$ to balance small and large categories. A_c denotes the number of pixels in category c. These pixels are back-projected to initialize Gaussian kernels with color c_i, opacity $\sigma_i = 0.5$, and radius $r_i = \frac{d_i}{f}$, where f is the focal length. The kernels are optimized with $\mathcal{L}_{map}$ for 1000 iterations to obtain an initial scene representation. This initialization ensures uniform surface coverage and faster Gaussian convergence. Mapping is then performed every $d_{map} = 5$ frames. Gaussians are densified based on the rendered silhouette; when the silhouette falls below τ_{sil} or rendered depth exceeds ground truth, new kernels are added to capture emerging foreground objects. Local BA optimization is applied within the submap, sampling keyframes each iteration to compute the loss:

$$\mathcal{L}_{mapping} = \sum \lambda_D |D_{pix} - D| + \lambda_S \mathcal{L}_S + \lambda_C \mathcal{L}_C \quad (3)$$

with semantic loss $\mathcal{L}_S$ and appearance loss $\mathcal{L}_C$ are weighted SSIM loss:

$$\mathcal{L}_{\text{ssim}} = \lambda \cdot |\hat{I} - I| + (1 - \lambda) \cdot \left(1 - \text{SSIM}\left(\hat{I}, I\right)\right) \quad (4)$$

All λ_* are hyperparameters. Poses and map are jointly optimized.

For submap allocation, we regard submaps in tracking and mapping as active. A new submap is created when both spatial and semantic conditions are met. Spatially, we assign each submap a bounding volume that encloses the viewing frusta of all its keyframes. This volume expands as more keyframes are added. We compute the overlap ratio r_{olap} between the viewing frustum of the current keyframe and the active submap. Semantically, we construct a histogram $\mathbf{h} \in \mathbb{R}^C$ of class proportions, where C is the number of class and measure the KL divergence r_{kl} between the current and anchor frames. A new submap is initialized when $r_{\text{olap}} < 0.5 \wedge r_{\text{kl}} > 0.4$. This spatial–semantic criterion prevents redundant or fragmented submaps. Newly established anchor frames also serve as boundary frames between submaps. We apply a local BA using $\mathcal{L}_{map}$ to optimize the Gaussian kernels near the boundary, after which the submap becomes inactive and waits for loop detection. The newly created active submap is initialized and then enters tracking and mapping.

B. Semantic-Guided Registration

Current 3DGS-SLAM methods still struggle with submap registration. For example, MAGIC-SLAM [15] performs ICP registration on the input point clouds, instead of directly registering Gaussian kernels. However, deviations between input point clouds and optimized kernels can introduce errors. LoopSplat [8] registers Gaussian directly but at high

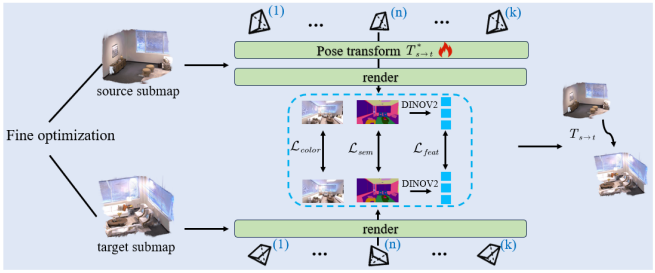

Fig. 3: **Fine optimization in SGR.** After coarse initialization, the source submap is aligned to the target by minimizing a rendering loss that enforces consistency in appearance, semantics, and DINOv2 features, while keeping Gaussian kernel parameters fixed.

cost, since it optimizes bidirectional transformations between source and target submaps, which doubles the computation. We propose Semantic-Guided Registration (SGR), using DINOv2 [14] to extract features from semantic maps for matching image pairs and coarse initialization, followed by a semantic-aware rendering loss for refinement. The entire process is unidirectional and thus more efficient.

Given two Gaussian submaps with a large overlap, we define the one with more keyframes as the target submap $\mathcal{M}_t$ and the other as the source submap $\mathcal{M}_s$. SGR follows a coarse-to-fine process to compute the rigid transformation from source to target. Specifically, for $\mathcal{M}_s$, we extract DINOv2 features $f_s^{(i)} \in \mathbb{R}^{384}$ from each semantic image in the keyframe list $\mathcal{K}_s$ to form a feature set F_s, and similarly for $\mathcal{M}_t$. We compute pairwise ℓ_2 distances and select the top k=10 nearest pairs. Then, the rigid transformation is optimized using Gauss-Newton:

$$T^*_{s\to t} = \arg\min_{T_{s\to t}\in SE(3)} \sum_{i=1}^{k} \left\| \log\left((T_t^i)^{-1} \cdot T_{s\to t} \cdot T_s^i \right) \right\|^2 \quad (5)$$

where $T_t^{(i)}$ and $T_s^{(i)}$ are poses in the target and source submaps, and log is the mapping from $SE(3)$ to $\mathfrak{se}(3)$. The result serves as the coarse initialization. We then transform the poses of the source submap to the target frame $\hat{T}_t^i = T_{s\to t} T_s^i$, and further refine the alignment using appearance, semantic rendering, and DINOv2 feature consistency:

$$\begin{aligned} \mathcal{L} = \sum_{i=1}^{k} & \left| C(\hat{T}_t^i) - C(T_t^i) \right| + \lambda_{sem} \left| S(\hat{T}_t^i) - S(T_t^i) \right| \\ & + \lambda_{feat} \left\| F_{dino}(S(\hat{T}_t^i)) - F_{dino}(S(T_t^i)) \right\|^2 \end{aligned} \quad (6)$$

with $\lambda_{sem} = 1$ and $\lambda_{feat} = 0.5$. The Gaussian kernel parameters are kept frozen. By enforcing consistency in appearance, semantic rendering, and high-level features, our method achieves more robust rigid registration even in blurred or repetitive texture scenes. The fine optimization process is illustrated in Fig. 3.

C. Semantic-Aware Loop Closure

Existing 3DGS SLAM systems lack robust loop closure. LoopSplat [8] uses NetVLAD to extract appearance features, which may fail in environments with sparse textures. HI-SLAM2 [16] relies on optical flow distances, but it is sensitive to illumination changes. We propose semantic-aware loop closure, an online method with scene understanding and stronger generalization. We leverage 2D and 3D semantic information for loop detection, which can correct both poses and Gaussians, enhancing global consistency. Specifically, a new keyframe is created every d_k frames, from which we extract 2D semantic image features $f_k^s \in \mathbb{R}^{384}$ using DINOv2 [14] and store them in a loop database. Every 6 keyframes, we compute the ℓ_2 distances $d_{k,j}^s$ between the current keyframe and all past keyframes j. Keyframes with $d_{k,j}^s < 0.20$ are selected as loop candidates. We further filter loops using semantic landmarks. With semantic color $s \in \mathbb{R}^3$ for each Gaussian, we assign its label by finding the nearest class color in the predefined semantic color mapping via Euclidean distance. For each Gaussian submap, we define Gaussian-IDF: $f(c) = \log\left(\frac{N}{1+N_c}\right)$, where N_c is the number of Gaussians of class c in the submap, and N is the total number. Gaussian-IDF reduces the influence of high-frequency background classes and emphasizes rare semantic classes. For a given frame pose, let M ($M < N$) be the number of observed Gaussians, and M_c the number of observed Gaussians of class c. The landmark score of class c is computed as:

$$w(c) = \frac{M}{M_c} f(c) \quad (7)$$

Then we can construct a vector $w \in \mathbb{R}$ composed of the landmark scores of all classes. The ℓ_2 distance between the landmark score vector of the current frame and those of candidate loop frames is computed, and the closest one is selected as the loop closure. The pseudocode is shown in Algorithm 1.

Unlike LoopSplat, which only optimizes poses between submaps, we consider both intra-submap and inter-submap

Algorithm 1 Semantic-Aware Loop Closure

Require: Input frames $\{\mathcal{I}_k\}$, Gaussian submaps $\{\mathcal{G}\}$
Ensure: Detected loop closure keyframe pair (k, j)

```
Create new keyframe every d_k frames
Extract 2D semantic features f_k^s ∈ R^384 using DINOv2
if every 6 keyframes then
    for all past keyframes feature f_j^s in loop database do
        Compute ℓ2 distance d_{k,j}^s between f_k^s and f_j^s
        if d_{k,j}^s < 0.20 then
            Add j to loop candidate set C
        end if
    end for
end if
Store f_k^s in loop database
for each submap G do
    Compute Gaussian-IDF f(c)
end for
Compute current frame I_k land markscore: w(c) = (M/M_c) f(c),
    where M is observed Gaussians, M_c of class c
Construct landmark score vector w_k ∈ R^c
for each loop candidate j ∈ C do
    Compute landmark score and Construct landmark score
    vector w_j
    Compute ℓ2 distance ||w_k − w_j||_2
end for
Select closest candidate as loop closure (k, j)
return (k, j)
```

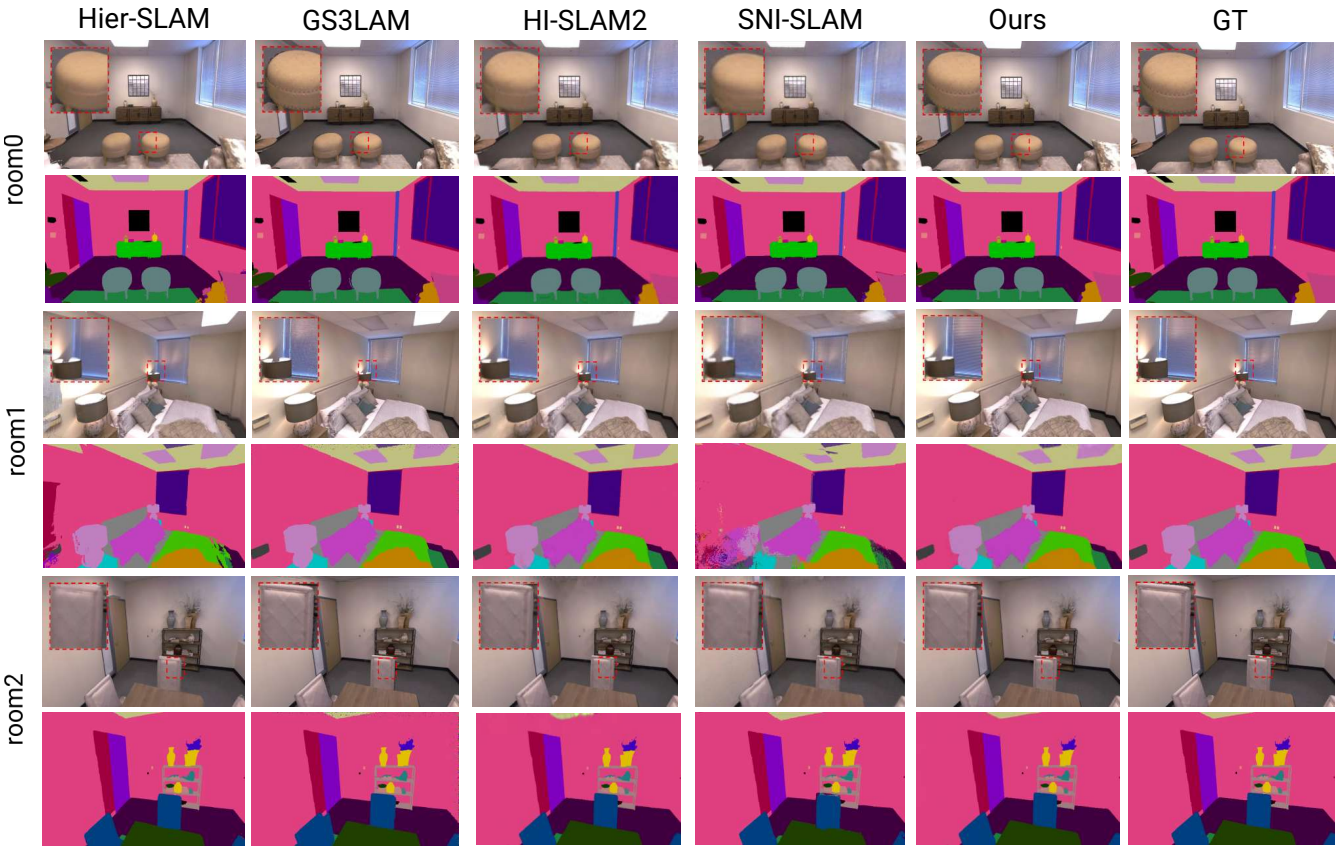

Fig. 4: Rendering and segmentation results on the Replica dataset.

loops. First, we check whether the two loop frames belong to the same submap. If they are in the same submap, we build a local pose graph optimization (PGO) with two constraints: odometry constraints from relative poses between adjacent keyframes, and loop constraints computed from image pairs using Eq. (6). After optimizing keyframe poses, we further update the submap with $\mathcal{L}_{\text{map}}$ as loss function, running 10 iterations per keyframe with fixed poses. If the two loop frames belong to different submaps, we estimate the rigid transformation between submaps using the method in Section III-B, and use it as a loop constraint. Odometry constraints are built from relative poses of anchor frames. After PGO, anchor frames are updated, and keyframe poses in the submap are corrected as $T_i^k \leftarrow (T_i^{a\prime})^{-1} T_i^a T_i^k$ where $T_i^{a\prime}$ is the optimized anchor pose and T_i^k is the keyframe pose. Gaussian centers are also updated $x_i \leftarrow (T_i^{a\prime})^{-1} T_i^a x_i$ where x_i is in homogeneous coordinates.

After processing all frames, we perform a two-stage global map refinement. In the first stage, we jointly optimize keyframe poses and Gaussian within each submap using the loss $\mathcal{L}_{\text{refine}} = \mathcal{L}_{\text{map}} + \lambda_{\text{feat}} \|F_{\text{dino}}(S_{\text{pix}}) - F_{\text{dino}}(S)\|^2$ running 10 iterations per keyframe. The added semantic feature term prevents overfitting in the mapping stage. In the second stage, we align submaps with the global map by minimizing the difference between rendered results from submap coordinates and global coordinates $\mathcal{L}_{\text{refine}}\left((T_i^k, x_i), (T_i^a T_i^k, T_i^a x_i)\right)$ again running 10 iterations per keyframe. Anchor frames are optimized, and Gaussians are aligned to the global map by $x_k = T_i^a x_i$. This two-stage refinement ensures view consistency between submaps and the global map, enabling the construction of a globally consistent representation.

IV. Experiment

A. Experiment Setup

1) Datasets and Baselines: Our experiments are conducted on three datasets: eight sequences from the synthetic Replica [17], six from real-world ScanNet [18], and three from TUM [19]. We compare against 3DGS-based SLAM methods Hier-SLAM [4], GS^{3}SLAM [3], HI-SLAM2 [16], LoopSplat [8] and the NeRF-based SNI-SLAM [5]. Except LoopSplat, all baselines are semantic-oriented, and only LoopSplat and HI-SLAM2 use loop closure.

2) Metrics: We employ PSNR, SSIM, and LPIPS to evaluate the rendering quality of RGB images, and mIoU to assess 2D semantic segmentation performance. To measure tracking accuracy, we use ATE RMSE. All reported results are averaged over five runs under identical experimental settings. The tables highlight the best, second-best, third-best results.

3) Implementation Details: All experiments are conducted on a desktop equipped with an RTX 3090 GPU (24GB memory) and an Intel Core i9-11900K CPU. We use the FAISS vector database to store feature vectors extracted by DINOv2, which enables efficient L2 distance retrieval and

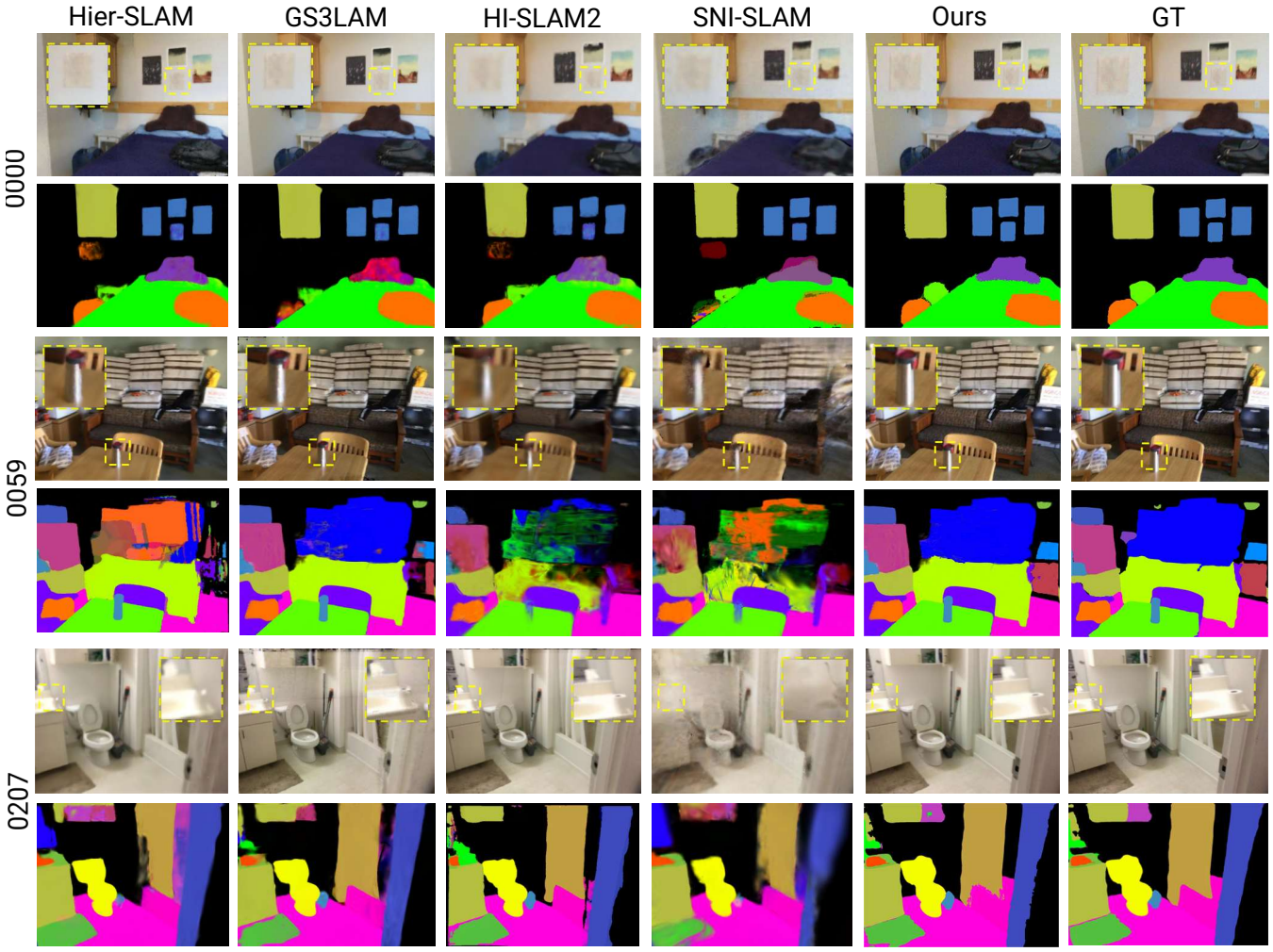

Fig. 5: Rendering and segmentation results on the ScanNet dataset.

TABLE I: Average tracking and mapping results on 8 Replica sequences. – indicates LoopSplat cannot perform semantic segmentation.

Method	Mapping				Tracking
	PSNR ↑	SSIM ↑	LPIPS ↓	mIoU [%]↑	ATE RMSE [cm]↓
Hier-SLAM	35.70	0.980	0.067	94.38	0.33
GS^{3}SLAM	36.26	**0.989**	0.052	94.31	0.37
HI-SLAM2	38.71	0.970	**0.030**	89.35	0.26
LoopSplat	36.63	0.985	0.112	–	0.26
SNI-SLAM	29.43	0.921	0.237	86.05	0.45
Ours	**39.94**	0.970	**0.030**	**95.81**	**0.24**

returns the top-k vectors closest to the query vector. The hyperparameters for the tracking and mapping losses are set to $\lambda_C = 1.0$, $\lambda_D = 0.5$, and $\lambda_S = 0.5$. In the SSIM loss $\mathcal{L}_{\text{ssim}}$, we set $\lambda = 0.8$.

B. Experiment Results

1) Evaluation on Replica [17]: The average tracking and mapping results on Replica are shown in Tab. I, where we achieved the lowest ATE RMSE averaged over 8 sequences. For mapping, as illustrated in Fig. 4, our method surpasses existing semantic SLAM approaches and LoopSplat in PSNR, and achieves higher mIoU than other semantic SLAM systems. This improvement stems from our designed Semantic-Guided Registration and Semantic-Aware Loop Closure, which effectively correct accumulated pose errors and enforce consistency on Gaussian kernels, thereby enhancing both tracking–mapping performance and semantic segmentation accuracy.

2) Evaluation on ScanNet [18]: In terms of tracking accuracy, we achieved a significant advantage, as shown in Tab. III. ScanNet is a real-world handheld dataset, where sensor and image noise pose challenges for tracking, and loop closures occur frequently, making effective loop detection essential. Other semantic SLAM methods lack dedicated loop detection, which leads to accumulated errors. Although LoopSplat incorporates loop detection, it only addresses inter-submap loops and determines submaps merely by thresholds on translational and rotational differences, which is not sufficiently reasonable since it neglects geometric correlations between submaps. This results in geometric discontinuities among submaps and consequently degrades mapping quality. In contrast, we establish submaps by considering both spatial and semantic distributions, and we correct loop closures both within and across submaps. This de-

TABLE II: Rendering performance on ScanNet.

Methods	Metrics	Avg.	0000	0059	0106	0169	0181	0207
Hier-SLAM	PSNR↑	23.00	22.85	21.35	22.26	23.57	24.13	23.85
	SSIM↑	0.853	0.840	0.825	0.868	0.880	0.860	0.843
	LPIPS↓	0.333	0.340	0.305	0.295	0.258	0.385	0.412
	mIoU↑	82.90	84.05	82.78	83.06	84.13	81.28	82.10
GS^{3}SLAM	PSNR↑	22.86	23.02	20.96	22.37	25.85	20.58	24.39
	SSIM↑	0.868	0.852	0.858	0.872	0.890	0.855	**0.878**
	LPIPS↓	**0.222**	0.277	**0.213**	**0.205**	0.189	0.252	**0.195**
	mIoU↑	84.39	83.15	84.06	82.47	85.06	85.27	86.31
HI-SLAM2	PSNR↑	29.28	28.62	27.22	28.13	**31.28**	30.37	30.03
	SSIM↑	**0.880**	0.850	**0.870**	0.900	**0.900**	**0.900**	0.860
	LPIPS↓	0.228	0.280	0.230	0.210	**0.100**	0.250	0.300
	mIoU↑	80.18	78.29	81.07	82.45	79.06	79.87	80.31
SNI-SLAM	PSNR↑	18.77	20.31	19.35	18.58	18.35	17.80	18.20
	SSIM↑	0.787	0.800	0.795	0.778	0.790	0.778	0.781
	LPIPS↓	0.435	0.410	0.480	0.380	0.357	0.510	0.473
	mIoU↑	79.26	77.63	78.35	80.51	79.25	79.75	80.07
LoopSplat	PSNR↑	24.92	24.99	23.23	23.35	26.80	24.82	26.33
	SSIM↑	0.845	0.840	0.831	0.846	0.877	0.824	0.854
	LPIPS↓	0.425	0.450	0.400	0.409	0.346	0.514	0.430
	mIoU↑	–	–	–	–	–	–	–
Ours	PSNR↑	**30.63**	**31.04**	**30.21**	**30.34**	30.89	**30.43**	**30.88**
	SSIM↑	0.873	**0.856**	0.862	**0.911**	0.880	0.861	0.868
	LPIPS↓	0.225	**0.276**	0.245	0.221	0.173	**0.240**	**0.195**
	mIoU↑	**87.15**	**87.25**	**87.80**	**88.45**	**85.70**	**86.68**	**87.00**

TABLE III: The ATE RMSE[cm]↓ results on ScanNet dataset.

Methods	Avg.	0000	0059	0106	0169	0181	0207
Hier-SLAM	11.66	13.64	9.65	17.80	11.53	10.04	7.32
GS^{3}SLAM	8.58	7.41	7.25	8.32	10.05	11.23	7.21
HI-SLAM2	7.16	5.82	7.30	**6.80**	8.25	7.41	7.40
SNI-SLAM	7.91	6.90	7.38	7.19	10.21	11.06	**4.70**
LoopSplat	7.73	6.20	7.10	7.40	10.60	8.50	6.60
Ours	**6.86**	**5.40**	**6.93**	6.82	**8.05**	**7.31**	6.62

TABLE IV: The render results on the TUM dataset.

Methods	Metrics	Avg.	f1/desk	f2/xyz	f3/office	f1/desk2	f1/room
Hier-SLAM	PSNR↑	21.73	21.37	22.41	20.79	21.38	22.69
	SSIM↑	0.859	0.847	0.886	0.871	0.838	0.851
	LPIPS↓	0.242	0.253	0.211	0.258	0.241	0.247
	mIoU↑	82.84	82.59	85.37	82.38	81.57	82.31
GS^{3}SLAM	PSNR↑	23.17	23.22	23.59	24.21	22.17	22.67
	SSIM↑	0.860	**0.851**	0.890	0.852	**0.843**	**0.862**
	LPIPS↓	0.197	0.213	0.147	**0.200**	0.235	**0.190**
	mIoU↑	**83.07**	83.26	84.07	82.27	83.51	82.26
HI-SLAM2	PSNR↑	26.22	25.62	28.83	**26.71**	24.25	25.68
	SSIM↑	0.859	0.845	**0.900**	0.860	0.837	0.852
	LPIPS↓	0.190	0.200	0.117	0.205	0.233	0.195
	mIoU↑	82.82	82.00	84.58	82.25	83.00	82.25
SNI-SLAM	PSNR↑	18.13	17.21	18.56	20.21	16.41	18.25
	SSIM↑	0.711	0.674	0.751	0.689	0.710	0.730
	LPIPS↓	0.366	0.450	0.333	0.356	0.361	0.328
	mIoU↑	79.98	80.06	81.53	78.21	79.31	80.81
LoopSplat	PSNR↑	23.22	22.03	22.68	23.47	23.89	24.05
	SSIM↑	0.858	0.849	0.892	**0.879**	0.822	0.850
	LPIPS↓	0.246	0.307	0.217	0.253	0.240	0.215
	mIoU↑	–	–	–	–	–	–
Ours	PSNR↑	**27.48**	**26.81**	**30.30**	26.59	**26.85**	**26.86**
	SSIM↑	**0.861**	0.850	**0.900**	0.865	0.837	0.855
	LPIPS↓	**0.186**	**0.180**	**0.115**	**0.200**	**0.230**	0.205
	mIoU↑	**85.67**	**85.00**	**86.80**	**86.20**	**84.58**	**85.79**

TABLE V: ATE RMSE[cm]↓ results on TUM dataset.

Methods	Avg.	f1/desk	f2/xyz	f3/office	f1/desk2	f1/room
Hier-SLAM	4.53	3.05	1.21	4.31	5.87	8.21
GS^{3}SLAM	4.25	2.72	1.61	4.06	5.61	7.25
HI-SLAM2	3.17	2.31	1.36	2.52	3.89	5.76
SNI-SLAM	4.35	2.56	**1.12**	**2.27**	4.35	11.46
LoopSplat	3.33	2.08	1.58	3.22	3.54	6.24
Ours	**2.83**	**2.07**	1.21	2.48	**3.19**	**5.18**

sign enables us to achieve superior mapping performance, as reported in Tab. II. In particular, for the long sequence 0000, our approach demonstrates a greater advantage. Ultimately, we realize a more accurate global map reconstruction, as illustrated in Fig. 5.

3) Evaluation on TUM RGB-D [19]: The tracking and mapping results on TUM are shown in Tabs. IV and V, respectively. TUM is a real-world handheld recording dataset, where motion blur frequently occurs and poses challenges to accurate loop closure detection. Compared to the LoopSplat, our ATE is improved by 15%, and PSNR is increased by 18%. This is attributed to our integration of semantic features and semantic landmarks for more robust loop closure detection on the TUM dataset, along with Gaussian and pose correction via SGR.

C. Ablation study

1) Submap Allocation Strategy: We combine spatial and semantic information to determine whether to initialize a new submap. As shown in Fig. 6, compared with the method that triggers submap initialization once the motion or rotation exceeds a predefined threshold, our approach effectively prevents geometric discontinuities at submap boundaries, resulting in artifact-free rendering.

2) Loop Closure: Tab. VII evaluates our loop closure performance against LoopSplat and HI-SLAM2, demonstrating that our introduced semantic landmarks enable more accurate and robust loop detection. For ground-truth loop generation, we partition the frame into fixed-length intervals of 10 frames. For any two intervals $[i_1, i_2]$ and $[j_1, j_2]$, define the admissible set $\mathcal{P} = \{(i,j) \mid i \in [i_1, i_2],\ j \in [j_1, j_2],\ |i-j| > \Delta t_{\min}\}$. If the positional difference between the two intervals satisfies $\max_{(i,j)\in\mathcal{P}} \left\|\mathbf{p}_i^{gt} - \mathbf{p}_j^{gt}\right\| < \tau_{\text{loop}}$, the interval pair is labeled as a ground-truth loop. A loop detection is considered successful if the two frames forming the loop belong to these intervals.

3) Submap Registration: For the ablation study of SGR shown in Tab. VI, we compare our approach with the LoopSplat registration module as well as directly applying FPFH+ICP [20] on Gaussian centers in the Replica dataset, including tracking accuracy and registration runtime. The entire registration process is more efficient than the bidirectional registration of LoopSplat. Moreover, it is demonstrated that the combination of coarse estimation and fine optimization improves tracking accuracy. Coarse estimation based on semantic feature similarity helps avoid local minima, while the fine optimization process incorporating rendering and semantic feature errors further enhances registration accuracy.

The quantitative results of the ablation study in Tab. VIII show that our designed semantic submap strategy, SGR, and the two-stage global adjustment collectively improve both tracking and mapping performance.

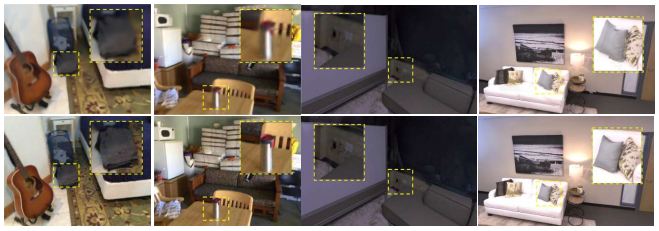

Fig. 6: **Submap allocation ablation.** The first row shows the rendering results of the strategy that establishes new submaps when motion or rotation exceeds a predefined threshold. The second row presents the rendering results of the submap establishment strategy that combines spatial overlap and semantic distribution, which improves rendering quality and eliminates artifacts.

TABLE VI: **Ablation Study on 3DGS Registration**. The numbers are computed based on average performance of 8 scenes on Replica. Corse. Ini. denotes the initial coarse estimation, Fine. Opt. refers to the fine optimization combining rendering and semantic features, and LoopSplat represents the registration module used in LoopSplat.

Corse. Ini.	*Fine. Opt.*	PSNR↑	ATE RMSE↓	Runtime (s)
FPFH+ICP		35.71	0.390	13.3
LoopSplat		36.63	0.243	1.42
✓	✗	36.60	0.268	**0.62**
✗	✓	38.25	0.265	0.93
✓	✓	**39.94**	**0.240**	1.03

V. Conclusions

In this paper, we propose GauSem-SLAM, which adopts 3DGS-based semantic submaps for scene representation. By introducing a semantic submap allocation and initialization strategy, we prevent geometric fragmentation of objects at submap boundaries. Furthermore, we design a more robust semantic loop closure detection and achieve correction on poses and Gaussians through efficient Semantic-Guided Registration. Experimental results demonstrate that our approach outperforms existing SLAM in both tracking and mapping. In the future, we will integrate multi-agent systems and explore applications in large-scale outdoor open environments.

TABLE VII: Loop closure ablation experiments on four ScanNet sequences.

Scene	Method	Precision↑	Recall↑	F1↑	PSNR↑	ATE RMSE↓
0000	w/o 3D semantic landmark	0.79	0.75	0.769	22.39	7.41
	LoopSplat	0.88	0.68	0.767	24.99	6.20
	HI-SLAM2	0.85	0.70	0.768	28.62	5.82
	Ours	**0.90**	**0.85**	**0.874**	**29.04**	**5.40**
0054	w/o 3D semantic landmark	0.77	0.76	0.765	22.87	9.81
	LoopSplat	0.85	0.72	0.780	23.10	16.00
	HI-SLAM2	0.80	0.77	0.785	26.36	8.64
	Ours	**0.86**	**0.80**	**0.829**	**28.51**	**7.79**
0181	w/o 3D semantic landmark	0.85	0.88	0.865	23.10	8.55
	LoopSplat	0.90	0.88	0.890	24.82	8.50
	HI-SLAM2	0.88	**0.89**	0.885	30.37	7.41
	Ours	**0.91**	**0.89**	**0.900**	**30.43**	**7.31**
0233	w/o 3D semantic landmark	0.87	**0.91**	0.890	27.70	5.25
	LoopSplat	0.93	0.90	0.915	28.71	4.70
	HI-SLAM2	0.92	0.90	0.910	28.39	4.93
	Ours	**0.95**	**0.91**	**0.930**	**29.07**	**4.41**

TABLE VIII: **Quantitative results of ablation study.** The values are averaged over six ScanNet sequences.

Method	Metrics				
	PSNR ↑	SSIM ↑	LPIPS↓	mIoU ↑	ATE RMSE ↓
w/o semantic allocation	27.90	0.860	0.247	84.00	7.10
w/o semantic initialize	28.95	0.870	0.227	85.97	6.90
w/o loop closure	27.51	0.850	0.246	82.05	8.21
w/o global refine	27.81	0.852	0.250	83.21	7.42
Ours	**29.58**	**0.873**	**0.225**	**86.65**	**6.86**

References

[1] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, *et al.*, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[2] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *ACM Transactions on Graphics*, vol. 42, no. 4, July 2023.

[3] L. Li, L. Zhang, Z. Wang, and Y. Shen, "Gs3lam: Gaussian semantic splatting slam," in *Proceedings of the 32nd ACM International Conference on Multimedia*, ser. MM '24. Association for Computing Machinery, 2024, p. 3019–3027.

[4] B. Li, Z. Cai, Y.-F. Li, I. Reid, and H. Rezatofighi, "Hier-slam: Scaling-up semantics in slam with a hierarchically categorical gaussian splatting," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[5] S. Zhu, G. Wang, H. Blum, J. Liu, *et al.*, "Sni-slam: Semantic neural implicit slam," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 167–21 177.

[6] H. Huang, L. Li, C. Hui, 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*, 2024.

[7] C. Campos, R. Elvira, J. J. Gómez, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM3: An accurate open-source library for visual, visual-inertial and multi-map SLAM," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[8] L. Zhu, Y. Li, E. Sandström, S. Huang, K. Schindler, and I. Armeni, "Loopsplat: Loop closure by registering 3d gaussian splats," in *International Conference on 3D Vision (3DV)*, 2025.

[9] H. Zhai, G. Huang, Q. Hu, G. Li, H. Bao, and G. Zhang, "NIS-SLAM: Neural implicit semantic RGB-D SLAM for 3D consistent scene understanding," *IEEE Transactions on Visualization and Computer Graphics*, vol. 30, no. 11, pp. 7129–7139, 2024.

[10] M. Li, D. Li, S. Hu, K. Wang, and Z. Zhao, "Slam-x: Generalizable dynamic removal for nerf and gaussian splatting slam," in *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025, pp. 1132–1140.

[11] M. Li, S. Liu, H. Zhou, G. Zhu, *et al.*, "Sgs-slam: Semantic gaussian splatting for neural dense slam," in *European Conference on Computer Vision*. Springer, 2024, pp. 163–179.

[12] Y. Mao, X. Yu, Z. Zhang, K. Wang, *et al.*, "Ngel-slam: Neural implicit representation-based global consistent low-latency slam system," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 6952–6958.

[13] Y. Tang, J. Zhang, Z. Yu, H. Wang, and K. Xu, "Mips-fusion: Multi-implicit-submaps for scalable and robust online neural rgb-d reconstruction," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 6, pp. 1–16, 2023.

[14] M. Oquab, T. Darcet, T. Moutakanni, H. V. Vo, *et al.*, "Dinov2: Learning robust visual features without supervision," 2023.

[15] V. Yugay, T. Gevers, and M. R. Oswald, "Magic-slam: Multi-agent gaussian globally consistent slam," in *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, June 2025, pp. 6741–6750.

[16] W. Zhang, Q. Cheng, D. Skuddis, N. Zeller, D. Cremers, and N. Haala, "Hi-slam2: Geometry-aware gaussian slam for fast monocular scene reconstruction," 2024.

[17] J. Straub, T. Whelan, L. Ma, Y. Chen, *et al.*, "The replica dataset: A digital replica of indoor spaces," *arXiv preprint arXiv:1906.05797*, 2019.

[18] A. Dai, A. X. Chang, M. Savva, M. Halber, *et al.*, "Scannet: Richly-annotated 3d reconstructions of indoor scenes," in *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017.

[19] J. Sturm, N. Engelhard, F. Endres, and W. Burgard, "A benchmark for the evaluation of rgb-d slam systems," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 573–580.

[20] Q.-Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *European conference on computer vision*. Springer, 2016, pp. 766–782.