

OpenMonoGS-SLAM: Monocular Gaussian Splatting SLAM with Open-set Semantics

Jisang Yoo¹, Gyeongjin Kang¹, Hyun-kyu Ko¹, Hyeonwoo Yu¹, Eunbyung Park²

Abstract—**Simultaneous Localization and Mapping (SLAM)** is a foundational component in robotics, AR/VR, and autonomous systems. With the rising focus on spatial AI in recent years, combining SLAM with semantic understanding has become increasingly important for enabling intelligent perception and interaction. Recent efforts have explored this integration, but they often rely on depth sensors or closed-set semantic models, limiting their scalability and adaptability in open-world environments. In this work, we present OpenMonoGS-SLAM, the first monocular SLAM framework that unifies 3D Gaussian Splatting (3DGS) with open-set semantic understanding. To achieve our goal, we leverage recent advances in Visual Foundation Models (VFM), including MAST3R for visual geometry and SAM and CLIP for open-vocabulary semantics. These models provide robust generalization across diverse tasks, enabling accurate monocular camera tracking and mapping, as well as a rich understanding of semantics in open-world environments. Our method operates without any depth input or 3D semantic ground truth, relying solely on self-supervised learning objectives. Furthermore, we propose a memory mechanism specifically designed to manage high-dimensional semantic features, which effectively constructs Gaussian semantic feature maps, leading to strong overall performance. Experimental results demonstrate that our approach achieves performance comparable to or surpassing existing baselines in both closed-set and open-set segmentation tasks, all without relying on supplementary sensors such as depth maps or semantic annotations.

I. INTRODUCTION

Simultaneous Localization and Mapping (SLAM) plays a fundamental role in robotics, augmented/virtual reality (AR/VR), and autonomous driving, where understanding spatial layouts and maintaining accurate localization are crucial. In recent years, the focus has expanded beyond purely geometric reconstruction, as spatial AI [1], [2] increasingly emphasizes semantic understanding of 3D environments, an aspect critical for enabling intelligent behavior, meaningful interaction, and effective decision-making.

3D Gaussian Splatting (3DGS) [3] has emerged as a powerful representation for fast and high-fidelity 3D rendering. It has laid the foundation for several novel SLAM systems that leverage differentiable rasterization for fast and efficient optimization [4]–[10].

While recent 3DGS-based SLAM methods have shown promising results, many still depend on depth sensors to achieve high-quality reconstruction [4]–[7]. This reliance limits their deployment in scenarios where depth input is unavailable, such as lightweight or low-cost platforms with

only RGB cameras. Monocular SLAM offers a more flexible and widely applicable alternative, but existing approaches often struggle to match the geometric accuracy and completeness of their RGB-D counterparts [8]–[10]. Bridging this performance gap remains a key challenge for making 3DGS-based SLAM practical in real-world, monocular settings.

On the other hand, there is a growing interest in incorporating semantic understanding into 3DGS-based SLAM systems to enable richer scene interpretation beyond geometry. Several recent methods [11]–[15] have explored this direction, aiming to augment 3D representations with semantic labels. However, most of these approaches are limited to closed-set recognition [11]–[13], relying on fixed taxonomies and curated training data, which restricts their ability to handle novel or open-set environments. Furthermore, they often rely on dense depth input [14], [16] for accurate reconstruction and semantic integration, which limits their deployment in monocular or resource-constrained scenarios.

In this work, we present a novel *monocular* SLAM framework that unifies *3D Gaussian splatting* and *open-set semantic understanding*, named as *OpenMonoGS-SLAM*. First, building upon the efficient and high-fidelity rendering capabilities of 3DGS, our framework enables accurate and fast 3D reconstruction and camera tracking. Second, relying solely on a monocular RGB input, the proposed method eliminates the need for depth sensors, making it broadly applicable. Third, we move beyond closed-set limitations by integrating open-vocabulary semantic reasoning. To the best of our knowledge, this combination has not been realized in any existing SLAM systems, despite being a direction that aligns closely with the long-term goals of the community.

To achieve our goal, we leverage recent Visual Foundation Models (VFM) [17]–[21] that offer promising zero-shot capabilities and generalization across diverse domains. More specifically, our method begins with a monocular setup that utilizes MAST3R-derived features [21] for camera tracking and 3D point reconstruction. To obtain semantic cues, we utilize a foundational segmentation model (SAM) [17] to generate 2D segmentation maps, which are then projected into 3D space. We enforce multi-view consistency to ensure coherent and stable semantic mapping across different viewpoints. Additionally, we employ CLIP [19] features to integrate high-level semantic understanding. In summary, our method harnesses the rich generalization capabilities of VFM to generate semantic features, which are embedded into the 3D Gaussian field for accurate and expressive scene representation.

To further improve the efficiency of *OpenMonoGS-SLAM*,

¹Sungkyunkwan University, Suwon, Republic of Korea.

²Yonsei University, Seoul, Republic of Korea. Corresponding author: Eunbyung Park.

we propose using a high-dimensional memory bank. We maintain a compact Gaussian feature map throughout the mapping process, and dynamically retrieve relevant high-dimensional CLIP features from the memory bank using an attention mechanism. This enables efficient integration of rich semantic information without inflating the per-Gaussian feature dimensionality. This allows for efficient and expressive representation of semantic information in 3D. Finally, we combine multiple self-supervised learning objectives to effectively learn semantic information without relying on any 3D semantic ground truth, achieving improved performance in both segmentation and mapping.

Our key contributions are summarized as follows:

- We propose a monocular 3D open-set semantic SLAM framework that integrates VFM-derived semantic features into 3DGS representation. By expressing these features in a multi-scale manner, our approach enables scale-aware and open-set semantic mapping without requiring expensive depth or annotation.
- We design an efficient and compact semantic fusion strategy that constructs a low-dimensional Gaussian semantic map by embedding 2D features via an attention mechanism over a memory bank of CLIP features. This structure allows efficient semantic retrieval and fusion during mapping, while self-supervised losses enable open-set training without requiring dense annotations.
- We validate our approach across standard benchmarks, showing that our method achieves high-quality semantic consistency and photorealistic rendering, comparable or superior to prior SLAM systems that rely on more restrictive inputs.

II. RELATED WORK

A. Visual Foundation Model

Recent advances in visual foundation models have significantly impacted a broad range of computer vision tasks by enabling strong generalization and transferability. These models are typically pretrained on large-scale image datasets using self-supervised or weakly supervised objectives, and are later adapted to a variety of downstream applications. Prominent 2D vision foundation models such as DINO [22], [23] and CroCo [24], [25], learn dense visual representations through contrastive learning or masked image modeling. Trained without explicit human supervision, these models capture rich semantic and geometric information and have been successfully applied to a wide range of tasks, including object discovery [26], [27], segmentation [17], dense matching [28], [29], and 3D reconstruction [20], [21], [30]. Their dense features serve as general-purpose image descriptors, often used as backbones in more specialized frameworks.

Our method builds upon this line of work by integrating visual representations, promptable segmentation models [17], [19] and dense matching frameworks [21], into a unified system for open-set semantic SLAM. By leveraging the generalization ability of visual foundation models, we aim to minimize task-specific supervision while achieving high performance in complex real-world environments.

B. Visual SLAM

Dense 3D reconstruction from multi-view images is a long-standing problem in computer vision, fundamental to mapping and scene understanding. The field has progressed from traditional methods, such as Structure-from-Motion [31], [32] and Multi-View Stereo [33], [34], to neural implicit [35] and explicit [3] approaches, offering improvements in accuracy and representation. However, these methods typically rely on offline processing, limiting their applicability in real-time scenarios.

To address this, recent work has explored integrating neural representations into incremental SLAM pipelines [4]–[10], enabling online reconstruction while retaining high-quality geometry and appearance. While these approaches have advanced real-time dense mapping, most remain focused on geometry and lack semantic understanding, especially in open-set or unfamiliar environments. Incorporating semantics into SLAM is challenging due to domain shifts, limited annotations, and poor generalization to novel object categories, which restricts the system’s adaptability in real-world scenarios.

Our method addresses these limitations by leveraging visual foundation models for open-set semantic SLAM, unifying promptable semantic segmentation and dense matching representations within a single framework.

III. METHOD

In this section, we detail the framework and training strategy of our proposed OpenMonoGS-SLAM. Our system builds upon MASt3R-SLAM, and incorporates semantic knowledge from Visual Foundation Models (VFs) into a 3D Gaussian Splatting (3DGS) representation for expressive and scalable scene understanding. The overall pipeline is illustrated in Fig. 1.

A. 3DGS-based Monocular SLAM

1) *MASt3R-SLAM*: We adopt MASt3R-SLAM [36] as our backbone model for camera tracking and pointmap estimation. MASt3R employs dense per-pixel features, enabling robust tracking under challenging conditions. We can obtain reliable associations by iteratively optimizing the ray errors. Given the output pointmaps $\mathbf{X}_i^i, \mathbf{X}_i^j \in \mathbb{R}^{HW \times 3}$, the optimal pixel coordinate in the reference frame i , for each target point $\mathbf{x} \in \mathbf{X}_i^j$ can be found with the following optimization objective,

$$\mathbf{p}^* = \arg \min_{\mathbf{p}} \|\psi([\mathbf{X}_i^i]_{\mathbf{p}}) - \psi(\mathbf{x})\|^2, \quad (1)$$

where $\mathbf{X}_i^j \in \mathbb{R}^{H \times W \times 3}$ denotes the output pointmap of image j in the camera i ’s coordinate frame, $\psi([\mathbf{X}_i^i]_{\mathbf{p}}), \psi(\mathbf{x})$ are a queried ray (i ’s frame) and the target ray (j ’s frame), normalized to unit norm, respectively. Based on the established correspondences, the camera tracking is achieved by minimizing the following ray error loss:

$$E_r = \sum_{m,n \in \mathbf{m}_{f,k}} \left\| \frac{\psi(\tilde{\mathbf{X}}_{k,n}^k) - \psi(\mathbf{T}_{kf}\mathbf{X}_{f,m}^f)}{w(\mathbf{q}_{m,n}, \sigma_r^2)} \right\|_{\rho}, \quad (2)$$

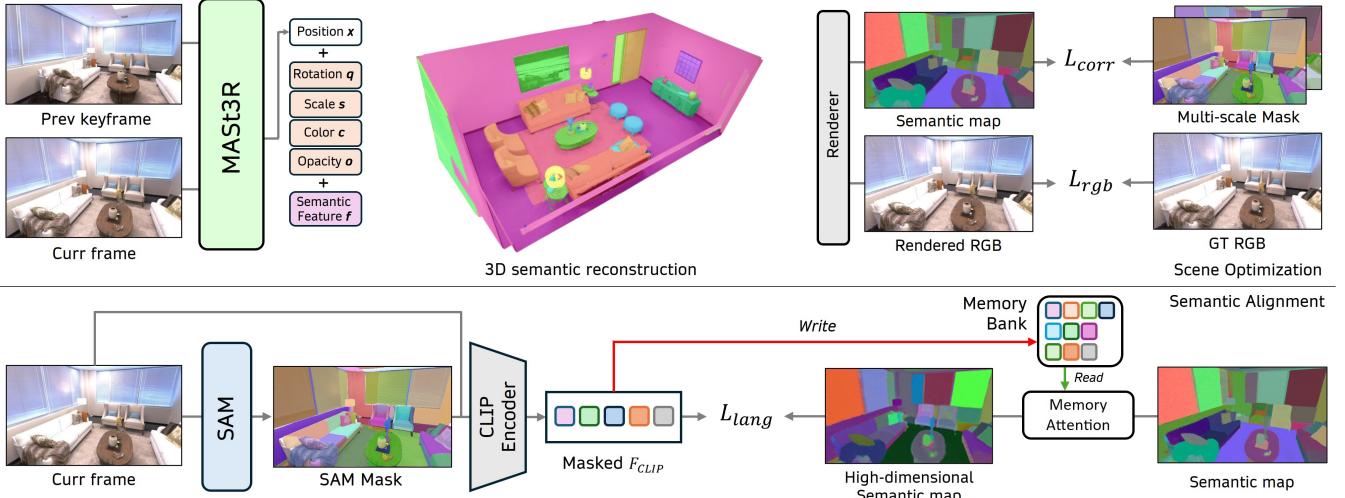


Fig. 1: Overview of Our Method. **Top:** Given the previous keyframe and the current frame, MAS3R estimates a point map. We reconstruct a 3D semantic map by augmenting each point with Gaussian attributes and a semantic feature vector. Rendering the 3D Gaussians yields an RGB color and a semantic feature map, which are supervised by the ground truth RGB image and multi-scale masks generated by SAM, respectively. **Bottom:** When the current frame is selected as a new keyframe, SAM generates instance masks, and masked CLIP features are extracted by applying the masks to the RGB image. These masked CLIP features are used to update the memory bank online and serve as the supervision target for the language-guided loss. The semantic map is further enhanced by memory attention to obtain a high-dimensional semantic map.

where $\tilde{\mathbf{X}}_{k,n}^k, \mathbf{X}_{f,m}^f \in \mathbb{R}^3$ denote the 3D coordinates of matching features between a frame f and a keyframe k , respectively. \mathbf{T}_{kf} represents the relative transformation from frame f to keyframe k . $w(\mathbf{q}_{m,n}, \sigma_r^2)$ is a confidence-based weighting function where $\mathbf{q}_{m,n} = \sqrt{\mathbf{Q}_{f,m} \mathbf{Q}_{f,n}^k}$ is the match confidence score. The Huber norm $\|\cdot\|_\rho$ is applied to improve robustness against outliers. Please refer to the original paper for further details [36].

2) 3D Gaussian Representation: The camera poses estimated through this tracking procedure are used to project MAS3R-generated 3D points into a global coordinate frame. We then initialize a set of 3D Gaussians at these positions. Each Gaussian maintains color attributes as well as learnable semantic features.

$$\mathcal{G}_i = \{\mathbf{x}_i, \mathbf{q}_i, \mathbf{s}_i, \alpha_i, \mathbf{c}_i, \mathbf{f}_i\}, \quad (3)$$

where \mathbf{x}_i denotes the 3D position, \mathbf{q}_i the quaternion orientation, \mathbf{s}_i the scale, α_i the opacity, \mathbf{c}_i the RGB color, and $\mathbf{f}_i \in \mathbb{R}^d$ the learnable semantic feature.

Since Gaussians are initialized using a pixel-aligned 3D pointmap from MAS3R, they can be accurately placed in 3D space, enabling high-quality rendering without the need for additional densification. Once initialized, the rendering of RGB and semantic features for each pixel coordinate \mathbf{p} is performed via depth-ordered alpha blending as follows,

$$\mathbf{C}_\mathbf{p} = \sum_{i=1}^N \mathbf{c}_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (4)$$

$$\mathbf{F}_\mathbf{p} = \sum_{i=1}^N \mathbf{f}_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j). \quad (5)$$

Similar to [5], we supervise 3D Gaussians not only with keyframes but also with additional mapping frames, which are used exclusively for optimization. This is necessary since the keyframes selected by MAS3R-SLAM are too sparse to ensure consistent updates across all regions. This representation yields a compact and expressive structure capable of both photorealistic rendering and semantic reasoning.

B. Multi-Scale Semantic Learning

Recent works have demonstrated the importance of modeling the multi-scale nature of semantic objects for effective 3D scene understanding [14], [37]–[39]. Building on this insight, we aim to guide 3D learning using 2D segmentation outputs from VFM. To improve generalization to various object sizes and avoid dependence on fixed semantic taxonomies, we adopt a scale-aware strategy. We leverage SAM to produce 2D object masks from each input image. These masks vary in size, reflecting the multi-scale nature of the objects. These 2D masks are then lifted into 3D using depth estimates from MAS3R-SLAM and the known camera intrinsics. We use these semantic information for enhancing multi-view semantic consistency, we encourage the readers to refer to SAGA for more details [39].

C. Language-embedded Semantic Memory Bank

While SAM provides instance-level segmentation, these masks lack semantic meaning without language grounding. [37] has demonstrated the effectiveness of embedding natural language into radiance fields for semantic understanding. Inspired by this, we produce embeddings for each segmented region using CLIP features to inject language priors into our 3D representation. However, storing raw CLIP embeddings for all Gaussians is prohibitively memory-intensive.

To address this, following the approach used in M3 [40], we implement a scalable memory bank that stores representative CLIP embeddings. While M3 proposes this mechanism in an offline setting, directly applying it to SLAM is impractical due to the need for online updates. To adapt this idea, we introduce an online mechanism: when a new keyframe is selected, we compute cosine similarity between the current CLIP embedding and the memory bank entries, which store representative CLIP embeddings from previous keyframes. If the similarity is lower than τ_m , which means sufficiently different, we append it to the memory bank. This ensures a diverse but compact coverage of the semantic space.

During mapping, the rendered 2D semantic features serve as queries and attend to the memory bank via an attention mechanism to retrieve high-dimensional semantic features. Given a rendered feature $\mathbf{F}_p \in \mathbb{R}^d$, the detailed formulation is as follows:

$$\hat{\mathbf{F}}_p = \text{softmax}((\mathbf{W}_{proj}\mathbf{F}_p)\mathcal{M}^\top)\mathcal{M}, \quad (6)$$

where $\mathcal{M} \in \mathbb{R}^{M \times D}$ is the memory bank composed of M high-dimensional CLIP features, and $\mathbf{W}_{proj} \in \mathbb{R}^{D \times d}$ is a linear projection layer that aligns the query space with the memory space, and $\text{softmax} : \mathbb{R}^M \rightarrow [0, 1]^M$ returns attentional scores. The output $\hat{\mathbf{F}}_p \in \mathbb{R}^D$ contains the attended high-dimensional features for each Gaussian.

This allows us to enrich each Gaussian with semantic information while maintaining a compact and efficient Gaussian map. At the same time, we fully leverage high-dimensional language features to maximize semantic expressiveness.

D. Self-Supervised Learning Objectives

We employ a combination of self-supervised losses to jointly optimize geometry, appearance, and semantics. This synergy between losses allows our model to build expressive 3D maps that are not only photorealistic but also semantically meaningful.

a) Photometric Supervision: We render novel views from the current 3D Gaussian representation and minimize a weighted combination of \mathcal{L}_1 and the structural similarity index loss \mathcal{L}_{SSIM} between the rendered and ground truth images. This encourages the learned Gaussians to maintain accurate spatial distribution and photorealistic visual fidelity.

$$\mathcal{L}_{rgb} = (1 - \lambda)\mathcal{L}_1 + \lambda(1 - \mathcal{L}_{SSIM}). \quad (7)$$

b) Multi-View Semantic Consistency. To enforce semantic consistency across different views, we apply a multi-view contrastive learning objective that pulls semantically similar Gaussians closer and pushes dissimilar ones apart in the feature space. Following SAGA [39], the corresponding contrastive loss between two sampled pixel coordinates $\mathbf{p}_1, \mathbf{p}_2$

is written as,

$$\begin{aligned} \mathcal{L}_{corr} = & \frac{1}{S|\mathcal{P}|^2} \sum_{s=1}^S \sum_{\mathbf{p}_1 \in \mathcal{P}} \sum_{\mathbf{p}_2 \in \mathcal{P}} (1 - 2 \cdot \text{Corr}_m(s, \mathbf{p}_1, \mathbf{p}_2)) \\ & \cdot \max(\text{Corr}_f(s, \mathbf{p}_1, \mathbf{p}_2), 0), \end{aligned} \quad (8)$$

where s denotes the scale index, and $\text{Corr}_m, \text{Corr}_f$ represent mask and feature correspondence, respectively. We sample $|\mathcal{P}| = 2,000$ points and use $S = 4$ scales. For more details, please refer to SAGA [39].

c) Language-Guided Semantic Alignment. To integrate open-set semantic priors from VFM, we supervise the learned features with CLIP embeddings via a regression objective:

$$\mathcal{L}_{lang} = \frac{1}{|\mathcal{P}|} \sum_{\mathbf{p} \in \mathcal{P}} \left[\left(1 - \cos(\hat{\mathbf{F}}_p, \mathbf{F}_p^{\text{CLIP}}) \right) + \left\| \hat{\mathbf{F}}_p - \mathbf{F}_p^{\text{CLIP}} \right\|_2^2 \right]. \quad (9)$$

d) Final Loss. We combine all losses into a unified training objective:

$$\mathcal{L}_{total} = \lambda_{rgb}\mathcal{L}_{rgb} + \lambda_{corr}\mathcal{L}_{corr} + \lambda_{lang}\mathcal{L}_{lang}. \quad (10)$$

By leveraging these complementary loss terms in a self-supervised manner, our model effectively balances geometric accuracy, photometric fidelity, semantic consistency, and language grounding. This enables OpenMono-SLAM to construct robust and interpretable 3D maps from monocular inputs, even in challenging open-set environments.

IV. EXPERIMENTS

A. Experiments Setup

Implementation Details. Our OpenMonoGS-SLAM system is built upon the MASt3R-SLAM framework. We initialize the 3D Gaussian attributes from dense 3D point maps generated from the pretrained MASt3R and adopt the gsplat framework [42] for efficient and scalable differentiable rendering of the 3D scene.

For multi-scale learning, we use $S = 4$ scale levels and the memory threshold is set to $\tau_m = 0.9$. The loss weight parameters are $\lambda_{photo} = 1.0$, $\lambda_{corr} = 0.05$, and $\lambda_{lang} = 0.05$. For visual foundation models (VFM), we utilize MASt3R, SAM (ViT-H), and CLIP (ViT-B) with their default configurations. All experiments are conducted on a single NVIDIA RTX 4090 GPU with 24GB VRAM. Mapping is performed for 30K iterations. Following MASt3R-SLAM’s preprocessing pipeline, we resize the image width to 512 pixels for both our model and baselines for evaluation.

To address open-set semantic segmentation, we incorporate Grounded-SAM [18], which enables segmentation via text prompt guidance without requiring predefined class labels. For fair comparisons with baseline methods, all offline models are trained using a consistent setting—selecting every 10th frame as a training view and optimizing the models for 30K iterations.

Dataset and Evaluation Metrics. We evaluate our method on the Replica [43] dataset, which contains ground-truth

Methods	Input	Metrics	Avg.	room0	room1	room2	office0	office1	office2	office3	office4
Visual SLAM											
MonoGS [8]	RGB	PSNR \uparrow	27.77	26.23	25.28	27.87	31.28	33.61	23.78	27.97	26.15
		SSIM \uparrow	0.858	0.821	0.776	0.869	0.886	0.916	0.830	0.891	0.878
		LPIPS \downarrow	0.203	0.179	0.304	0.184	0.181	0.142	0.262	0.136	0.237
Photo-SLAM [9]	RGB	PSNR \uparrow	30.37	29.73	26.65	31.97	35.27	28.74	30.72	29.25	30.62
		SSIM \uparrow	0.904	0.874	0.828	0.930	0.941	0.878	0.937	0.912	0.931
		LPIPS \downarrow	0.161	0.156	0.246	0.115	0.125	0.227	0.143	0.139	0.138
SEGS-SLAM [10]	RGB	PSNR \uparrow	33.54	31.25	27.27	35.09	38.56	38.64	34.42	33.90	29.35
		SSIM \uparrow	0.927	0.901	0.826	0.953	0.969	0.957	0.955	0.952	0.908
		LPIPS \downarrow	0.104	0.092	0.203	0.056	0.064	0.108	0.082	0.074	0.153
Ours (SAM1.0)	RGB	PSNR \uparrow	34.47	33.53	30.64	35.16	38.41	38.59	32.10	33.49	33.86
		SSIM \uparrow	0.957	0.953	0.914	0.967	0.969	0.971	0.959	0.959	0.963
		LPIPS \downarrow	0.086	0.066	0.153	0.075	0.065	0.084	0.085	0.067	0.093
Semantic SLAM											
DNS-SLAM [41]	RGB-D	PSNR \uparrow	22.89	22.45	24.61	25.27	24.09	25.28	21.39	21.87	18.20
		SSIM \uparrow	0.851	0.844	0.882	0.900	0.891	0.809	0.880	0.870	0.832
		LPIPS \downarrow	0.231	0.231	0.230	0.231	0.195	0.259	0.251	0.256	0.283
SGS-SLAM [13]	RGB-D	PSNR \uparrow	32.11	29.83	31.60	32.68	36.75	37.28	29.72	28.63	30.36
		SSIM \uparrow	0.922	0.891	0.915	0.941	0.959	0.949	0.916	0.897	0.907
		LPIPS \downarrow	0.343	0.153	0.168	0.153	0.136	0.185	0.176	0.186	0.213
Hier-SLAM [14]	RGB-D	PSNR \uparrow	33.57	30.16	31.15	33.46	37.36	39.12	31.27	30.83	33.19
		SSIM \uparrow	0.933	0.896	0.902	0.947	0.961	0.965	0.932	0.925	0.936
		LPIPS \downarrow	0.061	0.085	0.077	0.048	0.041	0.034	0.066	0.060	0.079
Ours (GT)	RGB	PSNR \uparrow	35.12	33.29	31.20	36.27	38.68	39.98	32.80	34.42	34.33
		SSIM \uparrow	0.960	0.952	0.918	0.972	0.970	0.976	0.963	0.966	0.966
		LPIPS \downarrow	0.080	0.067	0.151	0.066	0.064	0.073	0.081	0.058	0.084

TABLE I: Quantitative comparisons on the Replica dataset. Best results are highlighted as **FIRST**, **SECOND**. In the semantic SLAM setting, all baselines utilize RGB-D input, whereas our method relies solely on RGB. Despite this, it achieves comparable or better performance, highlighting the effectiveness of our approach.



Fig. 2: Qualitative comparisons of novel view synthesis on the Replica dataset.

semantic labels provided by [44]. We additionally evaluate on the TUM RGB-D benchmark [45] under a monocular setting, reporting tracking and reconstruction quality. For camera tracking evaluation, we compute the absolute trajectory error (ATE) using the root mean square error (RMSE) metric, implemented via the EVO [46] toolbox. To assess mapping quality, we evaluate photometric reconstruction using Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure [47] (SSIM), and Learned Perceptual Image Patch Similarity [48] (LPIPS). Semantic segmentation performance is quantified using mean Intersection over Union (mIoU). Frequency Weight IoU (FWIoU) and Accuracy (Acc) are employed for ablation studies.

Baselines. We evaluate tracking and novel view synthesis performance by comparing our method with representative monocular visual SLAM [8]–[10] and RGB-D semantic SLAM [13], [14], [41] baselines. For the evaluation of open-set segmentation, we evaluate our performance with offline

feature-based segmentation methods [49], [50]. All baselines are implemented using their official code repositories. To ensure fairness, we fix the number of mapping iterations to 30k for all methods. If a baseline is originally trained with fewer iterations, we fine-tune it until it reaches 30k iterations. Furthermore, to maintain consistent input resolution, we resize all input images to a width of 512 pixels for both our method and the baselines.

B. Results

Novel View Synthesis. We evaluate novel view rendering using all frames except the keyframes used for training. As shown in Tab. I and Fig. 2, our method consistently outperforms both visual and semantic SLAM baselines in most scenes, particularly in SSIM, indicating superior preservation of structural information. This improvement stems from jointly learning semantics and appearance; incorporating semantic cues helps maintain object-level consistency and

preserve structural boundaries, leading to more accurate and coherent reconstructions than baselines. For fair evaluation in semantic SLAM comparisons, our model is trained using ground-truth semantic masks instead of SAM-generated masks. Note that while all semantic SLAM baselines utilize ground-truth depth maps as additional input, our method relies solely on RGB yet still achieves superior performance. In addition, Tab. III reports mapping performance on the TUM-D dataset, where our method achieves the highest PSNR and SSIM and the second-best LPIPS among all methods, demonstrating that the learned 3D Gaussian representation generalizes well beyond Replica and produces sharper, more perceptually faithful renderings than monocular baselines.

Camera Tracking. As shown in Tab. II, our method significantly outperforms existing monocular visual SLAM approaches in terms of absolute trajectory accuracy. Notably, it achieves strong robustness with low variance across a wide range of indoor scenes, consistently maintaining low ATE RMSE values. This stability can be attributed to the use of visual foundation models trained on large-scale, diverse datasets, which provide rich semantic and geometric priors. These priors enable our model to adopt effectively across the dataset and support more accurate and reliable camera pose estimation under challenging conditions. A similar trend is observed on the TUM-D sequences (Tab. III), where our method maintains competitive tracking accuracy on real-world indoor scenes under a monocular input setting.

Methods	Avg.	R0	R1	R2	OF0	OF1	OF2	OF3	OF4
MonoGS	30.48	12.98	48.22	12.11	26.14	19.20	47.30	8.49	69.43
Photo-SLAM	10.63	1.60	22.56	3.36	2.11	30.02	6.95	2.04	16.38
SEGS-SLAM	9.25	1.07	35.09	0.22	1.70	0.74	0.62	1.11	33.40
Ours (SAM 1.0)	1.60	1.96	1.18	1.08	1.16	0.94	1.28	3.32	1.45

TABLE II: Camera tracking results on the Replica dataset. ATE RMSE in cm is reported. Our method demonstrates robust and consistent tracking performance across all scenes, achieving the lowest average error.

Category	Methods	ATE↓	PSNR↑	SSIM↑	LPIPS↓
Visual SLAM	MonoGS [8]	4.40	20.88	0.694	0.358
Visual SLAM	Photo-SLAM [9]	1.80	19.28	0.673	0.334
Visual SLAM	SEGS-SLAM [10]	1.14	23.20	0.769	0.271
Semantic SLAM	Ours (SAM1.0)	1.44	23.33	0.813	0.284

TABLE III: Quantitative comparisons on the TUM-D dataset.

Open-set Segmentation Results. To evaluate open-set segmentation performance using text prompts, we employ Grounded-SAM to generate segmentation masks conditioned on text prompts. Since both OpenMonoGS-SLAM and Feature 3DGS support prompt-based segmentation, we leverage the same evaluation text descriptions as prompts to obtain binary masks for each target concept. These binary masks are then applied to the rendered feature maps, enabling prompt-based segmentation within the visible regions of the scene. This setup allows us to fairly assess each model’s ability to localize and segment open-set concepts using only language-driven guidance, without relying on predefined semantic classes.

In contrast, Gaussian Grouping relies on a separate classifier for semantic prediction and does not support prompt-

based segmentation directly on the rendered feature maps. As a result, we follow their official evaluation protocol to ensure a consistent comparison. Specifically, we compute the Intersection over Union (IoU) between the text-prompted binary masks generated by Grounded-SAM and the predicted segmentation masks produced by Gaussian Grouping. To account for open-set settings and reduce noise from irrelevant regions, we report the mean IoU only for cases where the overlap exceeds a threshold of 0.2, which aligns with the default threshold specified in their implementation.

As shown in Tab. IV, our method demonstrates superior open-set segmentation performance across all scenes. This improvement is largely attributed to our effective integration of CLIP-derived language features into the Gaussian semantic feature space using a memory-based mechanism. This facilitates more accurate alignment between visual features and a wide range of textual prompts, allowing the model to effectively recognize and segment diverse concepts (Fig. 3).



Fig. 3: Qualitative comparisons of open-set segmentation on the Replica dataset. Our method produces cleaner and more complete segmentation masks, particularly for fine-grained structures.

Methods	Avg.	R0	R1	R2	OF0	OF1	OF2	OF3	OF4
Feature 3DGS	0.571	0.506	0.569	0.527	0.633	0.598	0.613	0.529	0.593
Gaussian Grouping	0.690	0.613	0.631	0.732	0.744	0.683	0.763	0.664	0.686
Ours (SAM1.0)	0.845	0.832	0.761	0.873	0.837	0.882	0.851	0.847	0.873

TABLE IV: Quantitative comparisons of open-set segmentation on the Replica dataset. IoU from text prompts is reported.

Closed-set Segmentation Results. While our primary focus is open-set semantic SLAM, we additionally evaluate closed-set segmentation to demonstrate the effectiveness and general applicability of our framework using ground-truth semantic annotations.

Since all baselines directly utilize ground-truth semantic IDs during SLAM optimization, their predicted labels can be directly compared to ground-truth annotations using mean Intersection over Union (mIoU). In contrast, our method is inherently designed with a class-agnostic framework that operates without explicit semantic supervision. To facilitate a meaningful comparison, we introduce an additional loss term during SLAM optimization that incorporates ground-truth semantic IDs, allowing our model to align with the closed-set evaluation protocol. Specifically, we augment our existing self-supervised objectives with a cross-entropy loss to guide semantic label learning. After SLAM optimization, we follow the same evaluation protocol as the baselines by computing mIoU between predicted and ground-truth masks for each semantic ID.

As shown in Tab. V, our method achieves the best performance across all scenes except R1 and Of1, where it remains highly competitive. This demonstrates that, despite being tailored for open-set scenarios, our approach generalizes well to closed-set segmentation tasks.

Methods	Avg.	R0	R1	R2	O0	O1	O2	O3	O4
DNS-SLAM	0.742	0.818	0.831	0.748	0.671	0.798	0.759	0.721	0.589
SGS-SLAM	0.866	0.862	0.887	0.865	0.886	0.927	0.854	0.782	0.866
Hier-SLAM	0.804	0.869	0.767	0.844	0.817	0.807	0.830	0.740	0.793
Ours(Prior)	0.896	0.886	0.881	0.900	0.904	0.915	0.909	0.875	0.901

TABLE V: Quantitative comparisons of close-set segmentation on the Replica dataset. mIoU based on ground-truth semantic labels is reported.

Ablations and Analysis. To validate the effectiveness of each component in our framework, we perform comprehensive ablation studies, including both qualitative and quantitative evaluations, as presented in Fig. 4 and Tab. VI. All methods are trained for 30K iterations using the full set of scenes from the Replica dataset for a fair comparison.

We begin by analyzing the contribution of each loss term through individual ablations. Removing the multi-view contrastive loss (“w/o contrastive loss”) results in a substantial drop in mIoU by approximately 26%, along with notable declines in FWIoU and pixel accuracy. This highlights the importance of enforcing semantic consistency across views through multi-view contrastive learning. Similarly, excluding the regression loss (“w/o regression loss”) also results in degraded performance, highlighting its essential role in guiding semantic alignment. The combination of both losses yields a synergistic effect that is critical for achieving robust and reliable open-set segmentation. As shown in the visual results in Fig. 4, removing either loss leads to inaccurate and semantically inconsistent masks. In particular, the absence of the contrastive loss produces spatially fragmented and less coherent segmentation outputs.

We also ablate the memory attention mechanism by removing the memory bank used to accumulate masked CLIP features. Without the memory bank (“w/o memory”), mIoU drops by 19%, and FWIoU and accuracy decline by around 10%, confirming the pivotal role of memory-based attention in semantic learning. This indicates that leveraging past CLIP embeddings across frames helps enforce multi-view consistency, in contrast to relying solely on the current frame’s CLIP feature. These results collectively verify the design choices of our framework and demonstrate that each component contributes significantly to the final performance.



Fig. 4: Qualitative comparison of ablated components on the Replica dataset. The top row shows the open-set segmentation results, while the bottom row presents the corresponding rendered features (with the ground-truth RGB image in the first column).

Setting	mIoU↑	FWIoU↑	Acc↑	PSNR↑
w/o contrastive loss	0.477	0.628	0.697	34.02
w/o regression loss	0.616	0.660	0.705	34.30
w/o memory	0.520	0.615	0.656	34.38
Ours (SAM1.0)	0.645	0.685	0.730	34.47

TABLE VI: Ablations. Our method achieves better semantic segmentation and reconstruction performance, demonstrating the importance of each component in our framework.

V. CONCLUSION

We introduced *OpenMonoGS-SLAM*, a novel semantic SLAM framework capable of open-set segmentation under a monocular setup. By integrating visual foundation models (VFs) such as MAST3R, SAM, and CLIP, our system achieves robust generalization across diverse tasks, enabling accurate monocular camera tracking and mapping, while providing rich semantic understanding in open-world environments. Furthermore, our method operates without any depth input or 3D semantic ground truth, relying entirely on self-supervised learning objectives to guide both geometric reconstruction and semantic understanding. Experimental results demonstrate that OpenMonoGS-SLAM achieves superior performance compared to existing baselines, highlighting the promise of integrating VFs into SLAM. However, a current limitation lies in the lack of robustness to dynamic scenes, inherited from the MAST3R backbone. This opens future directions for leveraging models like [51], which are trained on dynamic environments, to enable even more general and wild-scene applicability.

REFERENCES

- [1] Y. Mao, J. Zhong, C. Fang, J. Zheng, R. Tang, H. Zhu, P. Tan, and Z. Zhou, “Spatiallm: Training large language models for structured indoor modeling,” *arXiv preprint*, 2025.
- [2] A. J. Davison, “Futuremapping: The computational structure of spatial ai systems,” *arXiv preprint arXiv:1803.11288*, 2018.
- [3] 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. [Online]. Available: <https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/>
- [4] N. Keetha, J. Karhade, K. M. Jatavallabhula, G. Yang, S. Scherer, D. Ramanan, and J. Luiten, “Splatmap: Splat track & map 3d gaussians for dense rgb-d slam,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 357–21 366.
- [5] S. Ha, J. Yeon, and H. Yu, “Rgbd gs-icp slam,” in *European Conference on Computer Vision*. Springer, 2024, pp. 180–197.
- [6] C. Yan, D. Qu, D. Xu, B. Zhao, Z. Wang, D. Wang, and X. Li, “Gs-slam: Dense visual slam with 3d gaussian splatting,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 595–19 604.
- [7] 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 2024 Conference Papers*, 2024, pp. 1–11.
- [8] 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*, 2024, pp. 18 039–18 048.
- [9] H. Huang, L. Li, C. Hui, and S.-K. Yeung, “Photo-slam: Real-time simultaneous localization and photorealistic mapping for monocular, stereo, and rgbd cameras,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.
- [10] T. Wen, Z. Liu, B. Lu, and Y. Fang, “Scaffold-slam: Structured 3d gaussians for simultaneous localization and photorealistic mapping,” *arXiv preprint arXiv:2501.05242*, 2025.
- [11] Y. Haghghi, S. Kumar, J.-P. Thiran, and L. Van Gool, “Neural implicit dense semantic slam,” *arXiv preprint arXiv:2304.14560*, 2023.

- [12] S. Zhu, G. Wang, H. Blum, J. Liu, L. Song, M. Pollefeys, and H. Wang, “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.
- [13] M. Li, S. Liu, H. Zhou, G. Zhu, N. Cheng, T. Deng, and H. Wang, “Sgs-slam: Semantic gaussian splatting for neural dense slam,” in *European Conference on Computer Vision*. Springer, 2024, pp. 163–179.
- [14] 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,” *arXiv preprint arXiv:2409.12518*, 2024.
- [15] B. Li, V. C. Hao, P. J. Stuckey, I. Reid, and H. Rezatofighi, “Hier-slam++: Neuro-symbolic semantic slam with a hierarchically categorical gaussian splatting,” *arXiv preprint arXiv:2502.14931*, 2025.
- [16] D. Yang, Y. Gao, X. Wang, Y. Yue, Y. Yang, and M. Fu, “Openg-slam: Open-set dense semantic slam with 3d gaussian splatting for object-level scene understanding,” *arXiv preprint arXiv:2503.01646*, 2025.
- [17] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al., “Segment anything,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.
- [18] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, et al., “Grounded sam: Assembling open-world models for diverse visual tasks,” *arXiv preprint arXiv:2401.14159*, 2024.
- [19] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., “Learning transferable visual models from natural language supervision,” in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.
- [20] S. Wang, V. Leroy, Y. Cabon, B. Chidlovskii, and J. Revaud, “Dust3r: Geometric 3d vision made easy,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 697–20 709.
- [21] V. Leroy, Y. Cabon, and J. Revaud, “Grounding image matching in 3d with mast3r,” in *European Conference on Computer Vision*. Springer, 2024, pp. 71–91.
- [22] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, “Emerging properties in self-supervised vision transformers,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 9650–9660.
- [23] M. Oquab, T. Dariseti, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al., “Dinov2: Learning robust visual features without supervision,” *arXiv preprint arXiv:2304.07193*, 2023.
- [24] P. Weinzaepfel, V. Leroy, T. Lucas, R. Brégier, Y. Cabon, V. Arora, L. Antsfeld, B. Chidlovskii, G. Csurka, and J. Revaud, “Croco: Self-supervised pre-training for 3d vision tasks by cross-view completion,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 3502–3516, 2022.
- [25] P. Weinzaepfel, T. Lucas, V. Leroy, Y. Cabon, V. Arora, R. Brégier, G. Csurka, L. Antsfeld, B. Chidlovskii, and J. Revaud, “Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 17 969–17 980.
- [26] Y. Wang, X. Shen, S. X. Hu, Y. Yuan, J. L. Crowley, and D. Vaufreydaz, “Self-supervised transformers for unsupervised object discovery using normalized cut,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 543–14 553.
- [27] H. Zhang, F. Li, S. Liu, L. Zhang, H. Su, J. Zhu, L. M. Ni, and H.-Y. Shum, “Dino: Detr with improved denoising anchor boxes for end-to-end object detection,” *arXiv preprint arXiv:2203.03605*, 2022.
- [28] H. Jiang, A. Karpur, B. Cao, Q. Huang, and A. Araujo, “Omniglue: Generalizable feature matching with foundation model guidance,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 865–19 875.
- [29] J. Edstedt, Q. Sun, G. Bökman, M. Wadenbäck, and M. Felsberg, “Roma: Robust dense feature matching,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 19 790–19 800.
- [30] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, “Vggt: Visual geometry grounded transformer,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 5294–5306.
- [31] J. L. Schönberger and J.-M. Frahm, “Structure-from-motion revisited,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [32] L. Pan, D. Baráth, M. Pollefeys, and J. L. Schönberger, “Global structure-from-motion revisited,” in *European Conference on Computer Vision*. Springer, 2024, pp. 58–77.
- [33] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.
- [34] J. L. Schönberger, E. Zheng, M. Pollefeys, and J.-M. Frahm, “Pixel-wise View Selection for Unstructured Multi-View Stereo,” in *European Conference on Computer Vision (ECCV)*, 2016.
- [35] 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*, vol. 65, no. 1, pp. 99–106, 2021.
- [36] R. Murai, E. Dexheimer, and A. J. Davison, “MASI3R-SLAM: Real-time dense SLAM with 3D reconstruction priors,” in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.
- [37] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, “Lerf: Language embedded radiance fields,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 19 729–19 739.
- [38] C. M. Kim, M. Wu, J. Kerr, K. Goldberg, M. Tancik, and A. Kanazawa, “Garfield: Group anything with radiance fields,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 530–21 539.
- [39] J. Cen, J. Fang, C. Yang, L. Xie, X. Zhang, W. Shen, and Q. Tian, “Segment any 3d gaussians,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, 2025, pp. 1971–1979.
- [40] X. Zou, Y. Song, R.-Z. Qiu, X. Peng, J. Ye, S. Liu, and X. Wang, “3d-spatial multimodal memory,” in *The Thirteenth International Conference on Learning Representations*, 2025.
- [41] K. Li, M. Niemeyer, N. Navab, and F. Tombari, “Dns-slam: Dense neural semantic-informed slam,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 7839–7846.
- [42] V. Ye, R. Li, J. Kerr, M. Turkulainen, B. Yi, Z. Pan, O. Seiskari, J. Ye, J. Hu, M. Tancik, et al., “gsplat: An open-source library for gaussian splatting,” *Journal of Machine Learning Research*, vol. 26, no. 34, pp. 1–17, 2025.
- [43] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma, et al., “The replica dataset: A digital replica of indoor spaces,” *arXiv preprint arXiv:1906.05797*, 2019.
- [44] S. Zhi, T. Laidlow, S. Leutenegger, and A. J. Davison, “In-place scene labelling and understanding with implicit scene representation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15 838–15 847.
- [45] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, “A benchmark for the evaluation of rgbd slam systems,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 573–580.
- [46] M. Grupp, “evo: Python package for the evaluation of odometry and slam.” <https://github.com/MichaelGrupp/evo>, 2017.
- [47] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.
- [48] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.
- [49] S. Zhou, H. Chang, S. Jiang, Z. Fan, Z. Zhu, D. Xu, P. Chari, S. You, Z. Wang, and A. Kadambi, “Feature 3dg: Supercharging 3d gaussian splatting to enable distilled feature fields,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 676–21 685.
- [50] M. Ye, M. Danelian, F. Yu, and L. Ke, “Gaussian grouping: Segment and edit anything in 3d scenes,” in *European conference on computer vision*. Springer, 2024, pp. 162–179.
- [51] J. Zhang, C. Herrmann, J. Hur, V. Jampani, T. Darrell, F. Cole, D. Sun, and M.-H. Yang, “Monst3r: A simple approach for estimating geometry in the presence of motion,” *arXiv preprint arxiv:2410.03825*, 2024.