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

GeoGS-SLAM: Online Monocular Reconstruction Using Gaussian Splatting with Geometric Priors

Ruilan Gao[1], Letian Jin[1], Yu Zhang[1,2,*]

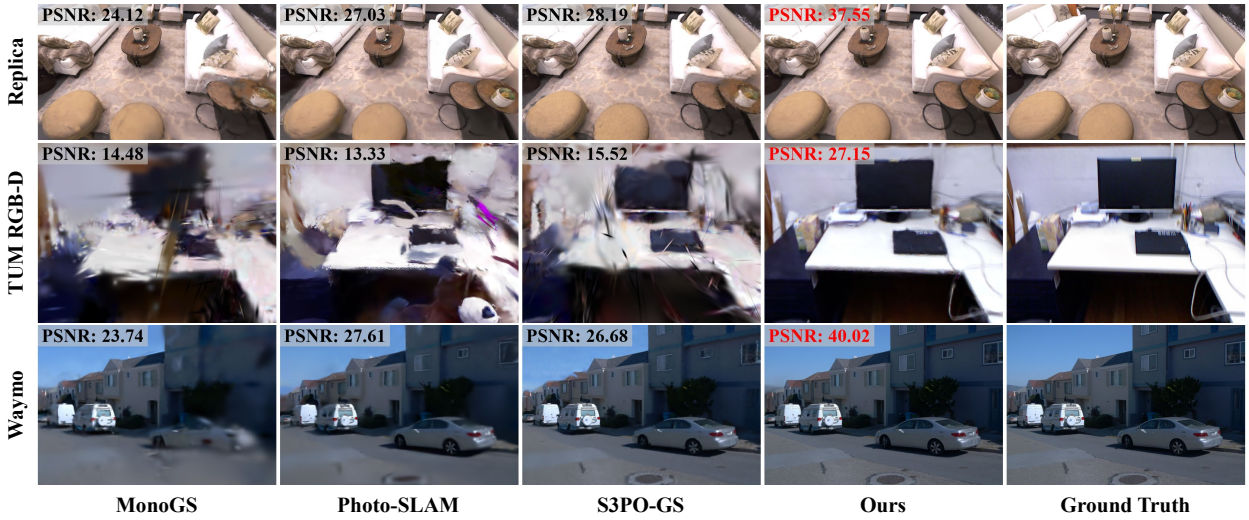

Fig. 1: **Rendering results on three datasets.** Our method produces high-fidelity reconstructions on both indoor and outdoor benchmarks, outperforming state-of-the-art monocular 3DGS-based SLAM methods.

***Abstract*— SLAM methods based on 3D Gaussian Splatting (3DGS) have demonstrated impressive tracking and mapping performance, but typically require additional geometric information from external depth sensors. Meanwhile, recent SLAM systems that leverage geometric priors from pre-trained feed-forward models enable real-time dense reconstruction, yet often discard original RGB information during optimization, thus degrading overall reconstruction quality. We present GeoGS-SLAM, an online monocular dense reconstruction system that combines the 3DGS-based map representation with learned geometric priors. Given uncalibrated RGB input, we first employ a feed-forward visual geometry model to predict camera and scene priors. The Gaussian scene map is then expanded by directly sampling Gaussian primitives from both RGB input and geometric priors. Camera poses and the scene map are jointly optimized through a coarse-to-fine strategy that minimizes both photometric and geometric losses. To ensure global consistency, we further incorporate online loop closure detection and pose graph optimization. Extensive experiments across indoor and outdoor benchmarks demonstrate that GeoGS-SLAM achieves superior rendering quality and tracking accuracy compared to state-of-the-art methods while maintaining online real-time performance. Project page: `https://rlgao.github.io/geogs_slam`.**

This work was supported by the National Natural Science Foundation of China (Grant No. 62576311), in part by NSFC 62088101 Autonomous Intelligent Unmanned Systems, and in part by Zhejiang Provincial Natural Science Foundation of China under Grant No. LD24F030001.

[1]State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou, China, 310027.

[2]Key Laboratory of Collaborative Sensing and Autonomous Unmanned Systems of Zhejiang Province, Hangzhou, China, 310027.

*Corresponding author: Yu Zhang (Email: zhangyu80@zju.edu.cn).

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) is a core problem in computer vision, serving as the foundation for applications ranging from robotics and autonomous driving to augmented reality systems. Recent advances in SLAM have been propelled by two complementary breakthroughs in view synthesis and 3D reconstruction: radiance field rendering [1] and feed-forward scene reconstruction [2].

The advent of neural radiance field (NeRF) [3] and 3D Gaussian Splatting (3DGS) [4] has fundamentally transformed scene representations in SLAM systems. In particular, 3DGS employs differentiable rasterization of 3D Gaussians to achieve efficient, photorealistic rendering, and has been shown to support high-quality tracking and mapping in SLAM [5], [6]. However, existing 3DGS-based SLAM systems predominantly rely on external geometric measurements from depth sensors and exhibit degraded performance when constrained to RGB-only input [7].

More recently, feed-forward models such as DUSt3R [8] and VGGT [9] have revolutionized 3D scene reconstruction through Transformer-based architectures trained at scale. A growing number of SLAM systems now leverage geometric priors from these powerful models to achieve real-time pose estimation and dense scene reconstruction [10], [11]. However, these methods typically treat the learned priors as the primary optimization signal while excluding original RGB observations from the optimization loop. For example, VGGT-SLAM [12] relies on the alignment of point map

priors predicted by VGGT. This limitation of discarding photometric information prevents closed-loop verification against visual evidence, potentially degrading reconstruction fidelity and consistency.

In this paper we present GeoGS-SLAM, an online monocular dense reconstruction system that synergistically combines 3DGS-based map representation with learned geometric priors. The key idea is to construct a closed-loop reconstruction pipeline that leverages feed-forward priors for geometric bootstrapping while preserving original RGB evidence for radiance field rendering-based optimization.

Our approach begins by employing a pre-trained visual geometry model to predict camera intrinsics and extrinsics, depth maps, and point maps from uncalibrated RGB input. Followed by scale alignment, these priors provide robust multi-view geometric guidance. We then directly sample Gaussian primitives from both images and geometric priors to expand the 3D Gaussian scene map. The map and camera poses are jointly refined through a coarse-to-fine, rendering-based optimization that minimizes both photometric and geometric losses. To ensure global consistency, we further integrate online loop closure detection and pose graph optimization.

Comprehensive experiments across indoor and outdoor benchmarks, including Replica [13], TUM RGB-D [14] and Waymo [15], demonstrate that GeoGS-SLAM achieves superior rendering quality and tracking accuracy compared to state-of-the-art (SOTA) monocular SLAM methods while maintaining online real-time performance.

The main contributions of our work are:

- We propose GeoGS-SLAM, a novel RGB-only SLAM system that integrates 3D Gaussian Splatting with feed-forward geometric priors in a unified closed-loop reconstruction pipeline, achieving robust tracking and high-fidelity mapping.
- We leverage efficient modules, including direct primitive sampling, rendering-based joint optimization, and online loop closure to enable accurate pose estimation and photorealistic reconstruction with real-time processing capabilities.
- We provide extensive experimental validation across indoor and outdoor benchmarks, demonstrating the superior performance of our approach compared to SOTA monocular SLAM methods in tracking accuracy and rendering quality.

II. RELATED WORK

A. Classical Visual SLAM

Early visual SLAM methods mainly employ feature-based pipelines. PTAM [16] introduces the first parallelized tracking and mapping framework, using sparse feature correspondences and bundle adjustment (BA) to produce accurate trajectories and sparse 3D maps. ORB-SLAM [17] and its successors [18], [19] extend this paradigm with efficient feature extraction, loop closure detection, and pose graph optimization to reduce drift and maintain long-term consistency.

In contrast to feature-based approaches, direct methods such as LSD-SLAM [20] and DSO [21] operate directly on pixel intensities rather than extracted keypoints, offering enhanced robustness in texture-poor scenes at the cost of increased photometric sensitivity. Dense SLAM methods, often incorporating multi-sensor configurations (stereo or RGB-D), enable richer scene reconstruction, producing dense maps suitable for interaction and navigation [22], [23].

Nonetheless, sparse feature-based methods provide robust tracking but only sparse geometry, while dense methods yield richer maps yet are photometrically sensitive and computationally expensive. These limitations have motivated recent efforts to integrate photorealistic rendering or learned priors into SLAM frameworks.

B. Radiance Field-based SLAM

Scene representations in SLAM systems have undergone a paradigm shift with the emergence of NeRF [3], [24], [25]. iMAP [26] has pioneered the integration of neural implicit representations into SLAM, utilizing multilayer perceptrons (MLPs) to encode both geometry and appearance within a unified framework. NICE-SLAM [27] and Vox-Fusion [28] further extend this paradigm by incorporating hierarchical feature grids and voxel-based neural implicit surface representations to enhance reconstruction quality and computational efficiency.

More recently, 3DGS [4] has emerged as a compelling alternative to NeRF-based representations, offering real-time rendering through differentiable rasterization of 3D Gaussians [29], [30], [31]. Pioneering methods such as MonoGS [5] and SplaTAM [6] demonstrate the successful integration of 3DGS as the sole scene representation for SLAM, achieving robust frame-to-model tracking and mapping through joint optimization of camera poses and Gaussian primitives. Other approaches, including Photo-SLAM [32], incorporate separate tracking modules to improve pose estimation accuracy. Recent works have focused on extending 3DGS-based SLAM to large-scale outdoor environments [7], [33], [34].

However, most existing radiance field-based SLAM systems rely heavily on depth measurements from RGB-D sensors, limiting their applicability in scenarios where only monocular RGB input is available. Therefore, we leverage pre-trained geometric priors to enable robust bootstrapping and optimization of 3DGS-based SLAM systems operating on uncalibrated RGB-only input.

C. Geometric Prior-based SLAM

The recent advancement of feed-forward reconstruction models [8], [9], [35], [36] has introduced a novel paradigm in SLAM through learned geometric priors, enabling dense reconstruction without relying on traditional geometric pipelines. MASt3R-SLAM [10] leverages MASt3R-predicted point maps and matching features [35] to construct a real-time dense monocular SLAM system, achieving globally consistent pose estimation and dense reconstruction. Similarly, SLAM3R [11] is built upon DUSt3R [8], utilizing the Image-to-Points (I2P) and Local-to-World (L2W) modules to

establish an end-to-end dense reconstruction framework. In contrast to these two-view prior-based approaches, VGGT-SLAM [12] leverages the more powerful VGGT architecture [9], employing point map alignment strategies for dense RGB SLAM.

However, these geometric prior-based SLAM methods predominantly use the learned priors as the primary optimization signal, where original RGB observations are often discarded. This limitation prevents closed-loop verification against photometric evidence and can compromise reconstruction quality. Therefore, we integrate geometric priors with photorealistic rendering-based optimization to achieve both accurate tracking and high-fidelity reconstruction.

III. METHOD

GeoGS-SLAM integrates 3D Gaussian Splatting with learned geometric priors in a unified framework for monocular dense reconstruction, as illustrated in Fig. 2.

Our approach consists of four core components: *Geometric Prior Prediction* (Sec. III-B) leverages a pre-trained model to produce priors of camera parameters and scene geometry from RGB keyframes; *Map Expansion* (Sec. III-C) updates the Gaussian map through direct primitive sampling; *Joint Optimization* (Sec. III-D) refines both the map and poses via rendering-based photometric and geometric loss minimization; and *Online Loop Closure* (Sec. III-E) with pose graph optimization further ensures global consistency.

A. 3DGS Scene Representation

We employ 3DGS as the scene representation, where each anisotropic Gaussian primitive $\mathcal{G}_i$ $(i = 1, \ldots, N)$ is parameterized by the following properties: position $\boldsymbol{\mu}_i \in \mathbb{R}^3$, color $\mathbf{c}_i \in \mathbb{R}^3$, scale $\mathbf{S}_i = \mathrm{diag}(\mathbf{s}_i) \in \mathbb{R}^{3\times3}$, rotation $\mathbf{R}_i \in \mathbb{R}^{3\times3}$, and opacity $o_i \in [0, 1]$. The covariance matrix $\boldsymbol{\Sigma}_i$ defining the ellipsoidal shape is computed as

$$\boldsymbol{\Sigma}_i = \mathbf{R}_i \mathbf{S}_i \mathbf{S}_i^\top \mathbf{R}_i^\top. \tag{1}$$

To render color and depth images from a given world-to-camera pose $\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathrm{SE}(3)$, the 3D Gaussians are first projected onto the image plane, with each splatted 2D Gaussian $\mathcal{G}'_i(\boldsymbol{\mu}'_i, \boldsymbol{\Sigma}'_i)$ obtained via

$$\boldsymbol{\mu}'_i = \pi(\mathbf{R}\boldsymbol{\mu}_i + \mathbf{t}),\ \boldsymbol{\Sigma}'_i = \mathbf{J}\mathbf{R}\boldsymbol{\Sigma}_i\mathbf{R}^\top\mathbf{J}^\top, \tag{2}$$

where $\pi(\cdot)$ denotes the projection operation, and $\mathbf{J}$ is the Jacobian of the affine approximation of the projective transformation.

The final color and depth at pixel $\mathbf{x}'$ can be rendered through α-blending of the depth-sorted Gaussians, given by

$$\mathbf{C} = \sum_{i=1}^{N} \mathbf{c}_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j),\ D = \sum_{i=1}^{N} d_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{3}$$

where the opacity weight α_i for each Gaussian is computed via

$$\alpha_i = o_i \exp\left(-\frac{1}{2}(\mathbf{x}' - \boldsymbol{\mu}'_i)^\top \left(\boldsymbol{\Sigma}'_i\right)^{-1} (\mathbf{x}' - \boldsymbol{\mu}'_i)\right). \tag{4}$$

B. Geometric Prior Prediction

We leverage the visual geometry model VGGT [9] for geometric prior prediction, which processes image sets of arbitrary length and predicts comprehensive 3D attributes in a single forward pass. This choice addresses the limitations of earlier two-view models like DUSt3R [8] and MASt3R [35], which are constrained to pairwise inference and thus limit multi-view consistency.

Similar to VGGT-SLAM [12], we employ a sliding window strategy to organize keyframes for multi-view prediction. An uncalibrated RGB frame is selected as a keyframe and added to the active sliding window if its optical flow-based relative displacement from the last keyframe exceeds a predefined threshold τ_{disp}. The initial window $\mathcal{W}^1$ accumulates keyframes $\mathbf{I}^1_1, \ldots, \mathbf{I}^1_w$ until reaching the fixed window size w. For subsequent windows $\mathcal{W}^k$ $(k > 1)$, we use the last keyframe from the previous window as the first keyframe, i.e., $\mathbf{I}^{k-1}_w = \mathbf{I}^k_0$, and add new keyframes $\mathbf{I}^k_1, \ldots, \mathbf{I}^k_w$ until it contains $w + 1$ frames in total. This overlapping design facilitates scale alignment between consecutive windows, as discussed below.

Each window $\mathcal{W}^k$ containing keyframes $\mathbf{I}^k_i \in \mathbb{R}^{W\times H\times3}$ $(i = 0, \ldots, w)$ is processed by the pre-trained model to generate predictions $\mathcal{F}(\mathcal{W}^k)$, including camera extrinsics $\widehat{\mathbf{T}}^k_i \in \mathrm{SE}(3)$, intrinsics $\widehat{\mathbf{K}}^k_i \in \mathbb{R}^{3\times3}$, depth maps $\widehat{\mathbf{D}}^k_i \in \mathbb{R}^{W\times H}$, and corresponding confidence maps $\widehat{\mathbf{Q}}^k_i \in \mathbb{R}^{W\times H}$. To obtain point maps $\widehat{\mathbf{X}}^k_i \in \mathbb{R}^{W\times H\times3}$ expressed in each keyframe's coordinate system, we unproject the depth maps to 3D using the predicted camera parameters rather than employing direct point map regression through the DPT head, as this approach yields better accuracy according to the original findings [9].

We align the scale of current predictions $\mathcal{F}(\mathcal{W}^k)$ with previous ones $\mathcal{F}(\mathcal{W}^{k-1})$ using the mutual keyframe $\mathbf{I}^{k-1}_w = \mathbf{I}^k_0$. For each pixel (u, v) where both predictions exhibit high confidence (i.e., $\widehat{\mathbf{Q}}^{k-1}_w(u, v) > \tau_{\text{conf}}$ and $\widehat{\mathbf{Q}}^k_0(u, v) > \tau_{\text{conf}}$), we compute the scale factor $\rho_{k-1,k}$ by

$$\rho_{k-1,k} = \frac{1}{|\mathcal{V}|} \sum_{(u,v)\in\mathcal{V}} \frac{\left\|\widehat{\mathbf{X}}^{k-1}_w(u, v)\right\|}{\left\|\widehat{\mathbf{X}}^k_0(u, v)\right\|}, \tag{5}$$

where $\mathcal{V}$ denotes the set of valid high-confidence pixels. This scale factor is applied to adjust all metric predictions in $\mathcal{F}(\mathcal{W}^k)$, including translational components $\widehat{\mathbf{t}}^k_i$ of camera poses $\widehat{\mathbf{T}}^k_i$, depth maps $\widehat{\mathbf{D}}^k_i$, and point maps $\widehat{\mathbf{X}}^k_i$ $(i = 0, \ldots, w)$, thereby ensuring scale consistency across all geometric priors.

C. Map Expansion

We leverage a direct sampling strategy to generate new Gaussian primitives from both input images and predicted geometric priors, inspired by recent work [37]. This approach enables map expansion while avoiding redundant primitive placement through a two-stage probability assessment. To determine optimal locations for new Gaussian primitives, we utilize the Difference of Gaussians (DoG) operator [38] to

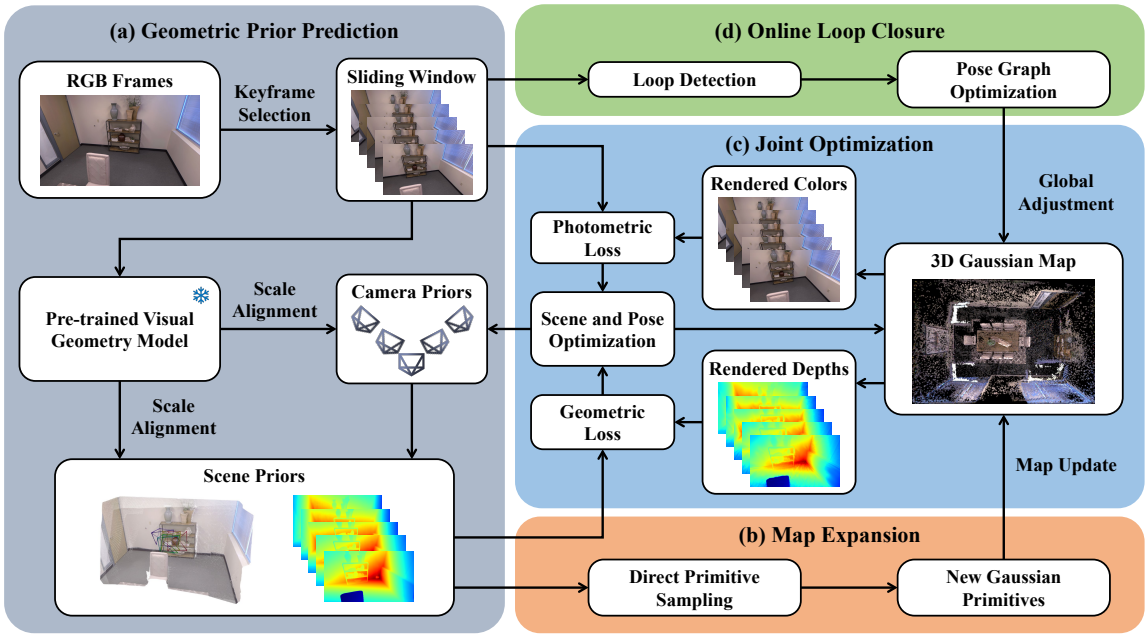

Fig. 2: **Overview of GeoGS-SLAM.** (a) Given uncalibrated RGB input, the system first selects keyframes and performs *Geometric Prior Prediction*, producing camera and scene priors using a pre-trained visual geometry model. (b) *Map Expansion* directly samples Gaussian primitives from both input images and geometric priors to update the map. (c) The map and camera poses are refined through rendering-based *Joint Optimization* that minimizes both photometric and geometric losses. (d) *Online Loop Closure* followed by pose graph optimization is integrated to further enhance global consistency.

identify regions with rich geometric detail. The probability matrix $\mathbf{P}_1 \in \mathbb{R}^{W \times H}$ for primitive placement at each pixel in image $\mathbf{I}$ is computed as

$$\mathbf{P}_1 = \left\| (\mathbf{\Phi}_{\sigma_1} - \mathbf{\Phi}_{\sigma_2}) * \mathbf{I} \right\|, \tag{6}$$

where $\mathbf{\Phi}_\sigma$ denotes a Gaussian kernel with zero mean and standard deviation σ, and we set $\sigma_1 = 0.5, \sigma_2 = 1.5$.

To prevent redundant primitive placement in already well-represented regions, we render a synthetic view $\widetilde{\mathbf{I}}$ using the current 3D Gaussian map from the predicted camera pose, and compute a corresponding occupancy probability $\mathbf{P}_2$ using the same DoG operator as in Eq. (6). A new Gaussian primitive can be spawned at pixel (u, v) only when the difference between placement and occupancy probabilities exceeds threshold τ_{prim}, i.e., $\mathbf{P}_1(u, v) - \mathbf{P}_2(u, v) > \tau_{\text{prim}}$.

For each qualified pixel (u, v), we initialize the new Gaussian primitive by combining information from both the input image and geometric priors. The color $\mathbf{c}$ is directly sampled from the corresponding pixel value $\mathbf{I}(u, v)$. The 3D position $\boldsymbol{\mu}$ is obtained from the point prior $\widehat{\mathbf{X}}(u, v)$ transformed to the world coordinate system. The scale s for each dimension is computed based on the predicted focal length $\hat{f}$ from intrinsics $\widehat{\mathbf{K}}$ and the point position $\widehat{\mathbf{X}}(u, v)$ in the camera coordinate system, as given by

$$s = \frac{\left\| \widehat{\mathbf{X}}(u, v) \right\|}{\hat{f}} + \epsilon, \tag{7}$$

where ϵ is a constant.

D. Joint Optimization

Following Gaussian primitive expansion from the current keyframe, we perform online joint optimization of both the scene map $\mathcal{G}$ and camera poses within the active keyframe window $\mathcal{W}$ by minimizing a combination of photometric and geometric losses. The photometric loss $\mathcal{L}_{\text{pho}}$ enforces visual consistency between rendered and observed images by combining L1 and SSIM losses [39], as computed by

$$\mathcal{L}_{\text{pho}} = \left\| \widetilde{\mathbf{I}} - \mathbf{I} \right\|_1 + \lambda_{\text{SSIM}} \left(1 - \text{SSIM}(\widetilde{\mathbf{I}}, \mathbf{I}) \right), \tag{8}$$

where $\widetilde{\mathbf{I}}$ represents the rendered image and $\mathbf{I}$ denotes the input ground truth.

The geometric loss $\mathcal{L}_{\text{geo}}$ maintains consistency between the rendered depth map $\widetilde{\mathbf{D}}$ and the predicted depth prior $\widehat{\mathbf{D}}$, as obtained by

$$\mathcal{L}_{\text{geo}} = \left\| \widetilde{\mathbf{D}} - \widehat{\mathbf{D}} \right\|_1. \tag{9}$$

The complete optimization objective combines photometric and geometric terms across all keyframes in the active window, which is formulated as

$$\min_{\mathcal{G}, \{\mathbf{T}_i\}} \sum_{\mathbf{I}_i \in \mathcal{W}} \left(\mathcal{L}_{\text{pho},i} + \lambda_{\text{geo}} \mathcal{L}_{\text{geo},i} \right). \tag{10}$$

To accelerate convergence and improve optimization stability, we employ a coarse-to-fine training strategy [32]. We construct n-level image pyramids for both the input image and prior depth map of each keyframe, where level l corresponds to the image or depth map downsampled by a factor of 2^l. Optimization begins at the coarsest level using downsampled images and depth maps as the photometric

and geometric supervision signals, respectively, then progressively decreasing the pyramid level at fixed iteration intervals until reaching the original full resolution. This multi-scale training approach enables rapid initial convergence while preserving fine-grained details in the final optimization stages.

E. Online Loop Closure

To mitigate accumulated drift and ensure global consistency, we incorporate online loop closure detection with pose graph optimization.

We first extract a global descriptor $\mathbf{d} \in \mathbb{R}^{8448}$ for each keyframe using the pre-trained MegaLoc model [40], which provides robust place recognition capability. For a query keyframe $\mathbf{I}^k_{\text{query}}$, loop closure detection proceeds by identifying the most similar keyframe candidate $\mathbf{I}^l_{\text{cand}}$ from all previous windows based on the cosine similarity between the global descriptors, as obtained via

$$\mathbf{I}^l_{\text{cand}} = \underset{\mathbf{I}^j_i \in \mathcal{W}^j,\, \forall j<k}{\arg\max} \cos\left(\mathbf{d}(\mathbf{I}^k_{\text{query}}), \mathbf{d}(\mathbf{I}^j_i)\right). \tag{11}$$

To prevent false positive detections, the candidate is validated as a genuine loop frame $\mathbf{I}^l_{\text{loop}}$ only if the similarity score exceeds a predefined threshold τ_{sim}, as given by

$$\mathbf{I}^l_{\text{loop}} = \begin{cases} \mathbf{I}^l_{\text{cand}}, & \text{if } \cos\left(\mathbf{d}(\mathbf{I}^k_{\text{query}}), \mathbf{d}(\mathbf{I}^l_{\text{cand}})\right) > \tau_{\text{sim}}, \\ \text{null}, & \text{otherwise.} \end{cases} \tag{12}$$

Upon detecting valid loop closures, we extend the active window to include all detected loop frames, as obtained via

$$\overline{\mathcal{W}}^k = \{\mathbf{I}^k_0, \ldots, \mathbf{I}^k_w\} \cup \{\mathbf{I}^{l_1}_{\text{loop},1}, \ldots, \mathbf{I}^{l_m}_{\text{loop},m}\}. \tag{13}$$

This extended window is processed by the feed-forward geometry model to obtain relative pose estimates between query frames in $\overline{\mathcal{W}}^k$ and corresponding loop frames from previous windows $\mathcal{W}^{l_1}, \ldots, \mathcal{W}^{l_m}$. After scale alignment, these pose estimates serve as loop closure constraints in the pose graph, complementing the odometry constraints derived from locally optimized keyframe poses. Then the complete pose graph is optimized using the Levenberg-Marquardt algorithm, yielding pose correction $\Delta\mathbf{T}_i = \begin{bmatrix} \Delta\mathbf{R}_i & \Delta\mathbf{t}_i \\ \mathbf{0} & 1 \end{bmatrix}$ for each keyframe $\mathbf{I}_i$.

Finally, we apply these pose corrections to update both keyframe poses and the associated Gaussian primitives, as given by

$$\mathbf{T}'_i = \Delta\mathbf{T}_i \mathbf{T}_i, \tag{14}$$

$$\boldsymbol{\mu}'_{\mathcal{G}_i} = \Delta\mathbf{R}_i \boldsymbol{\mu}_{\mathcal{G}_i} + \Delta\mathbf{t}_i,\ \mathbf{R}'_{\mathcal{G}_i} = \Delta\mathbf{R}_i \mathbf{R}_{\mathcal{G}_i}, \tag{15}$$

where $\mathcal{G}_i$ is a primitive originally spawned by keyframe $\mathbf{I}_i$.

IV. EXPERIMENTS

A. Experimental Setup

i) **Datasets.** We conduct experiments across indoor and outdoor benchmarks, including Replica [13], TUM RGB-D [14] and Waymo [15]. Specifically, we use the `Office` and `Room` sequences from Replica, the `fr1` sequences from TUM RGB-D, and nine 200-frame sequences from Waymo. Results presented below are averaged across sequences within each respective dataset.

ii) **Metrics.** We use PSNR, SSIM [39] and LPIPS [41] for rendering quality evaluation, and the RMSE of ATE for tracking accuracy assessment.

iii) **Baseline Methods.** For rendering quality evaluation, our method is compared against SOTA radiance field-based SLAM methods operating on monocular RGB input, including neural implicit representation-based approaches (GO-SLAM [42], GlORIE-SLAM [43]) and 3DGS-based methods (MonoGS [5], Photo-SLAM [32], Splat-SLAM [44], DROID-Splat [45], S3PO-GS [34]). For tracking accuracy assessment, we additionally compare against geometric prior-based SLAM methods (SLAM3R [11], MASt3R-SLAM [10], VGGT-SLAM [12]).

iv) **Implementation Details.** Our method is implemented using PyTorch and CUDA. All experiments are conducted on a desktop with an NVIDIA RTX 4090 GPU and an Intel Core i9-13900K CPU.

B. Results

i) **Rendering Quality.** Tab. I presents rendering quality comparisons across all evaluated datasets, showing that our method achieves SOTA performance across all metrics and benchmarks. Compared to current best radiance field-based monocular SLAM methods, our method delivers substantial improvements in PSNR: **+4.94** on Replica, **+3.15** on TUM RGB-D, and **+3.81** on Waymo.

Fig. 3 provides qualitative comparisons across representative scenes from all three datasets. For indoor environments (Replica and TUM RGB-D), our method produces detailed reconstructions that accurately capture fine-grained textures, lighting variations, and geometric structures. For complex outdoor scenarios (Waymo), our method successfully handles challenging conditions including varying illumination, dynamic content, and large-scale environments, showcasing its scalability to real-world autonomous driving applications.

These results demonstrate the substantial benefits of integrating learned geometric priors into 3DGS-based SLAM frameworks.

ii) **Tracking Accuracy.** Tab. II presents tracking results across the datasets, with the comparison of estimated trajectories shown in Fig. 4. Our method achieves SOTA tracking performance compared to methods operating on uncalibrated RGB input, while maintaining competitive accuracy against approaches that leverage calibrated camera parameters.

Specifically, our method outperforms the best-performing geometric prior-based SLAM method, MASt3R-SLAM, reducing tracking error by **46.7%** on Replica, **40.7%** on TUM RGB-D, and **64.6%** on Waymo. While these methods, like MASt3R-SLAM and VGGT-SLAM, achieve reasonable initial pose estimates through learned priors, they fail to refine these estimates against actual image evidence. This limitation leads to accumulated drift and reduced accuracy over extended sequences and large-scale environments with greater scene complexity.

TABLE I: **Rendering quality comparison across three datasets.** Best results are highlighted in **bold**. Our method demonstrates superior rendering performance compared to both neural implicit representation-based approaches (*NI*) and 3DGS-based methods (*GS*) across all indoor and outdoor benchmarks.

	Method	Replica			TUM RGB-D			Waymo		
		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NI	GO-SLAM [42]	24.20	0.935	0.462	12.59	0.601	0.727	9.72	0.109	0.812
	GlORIE-SLAM [43]	24.73	0.911	0.171	17.20	0.725	0.384	23.90	0.892	0.247
GS	MonoGS [5]	26.77	0.848	0.281	15.27	0.534	0.575	16.60	0.649	0.655
	Photo-SLAM [32]	32.62	0.917	0.162	16.46	0.593	0.471	19.86	0.767	0.595
	Splat-SLAM [44]	31.86	0.906	0.133	20.87	0.700	0.370	25.10	0.773	0.349
	DROID-Splat [45]	29.96	0.882	0.227	19.80	0.685	0.397	22.84	0.737	0.350
	S3PO-GS [34]	29.94	0.892	0.199	18.00	0.627	0.442	21.82	0.798	0.471
	GeoGS-SLAM (Ours)	**37.56**	**0.968**	**0.039**	**24.02**	**0.797**	**0.220**	**28.91**	**0.900**	**0.135**

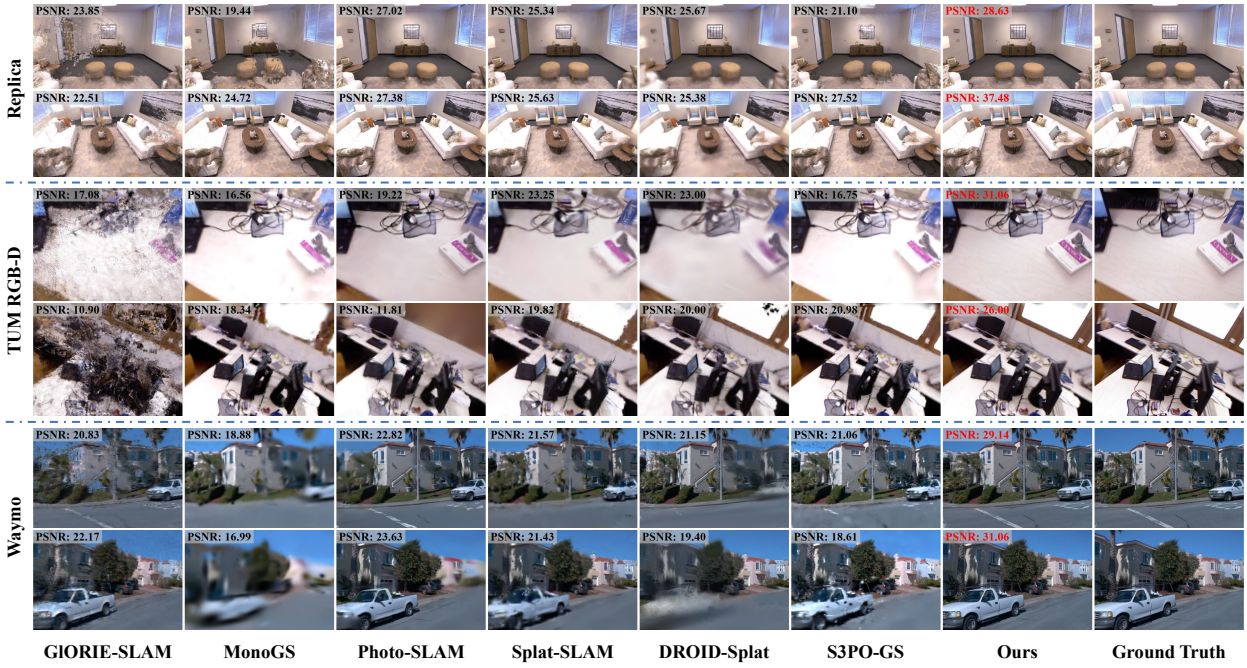

Fig. 3: **Comparison of rendering results.** Our method produces photorealistic reconstructions for both indoor and outdoor scenes. For indoor environments, it captures fine-grained textures and geometric details with minimal artifacts. For outdoor scenarios, it successfully handles complex driving scenes, preserving architectural structures and vehicle details.

These substantial tracking improvements across diverse environments validate the effectiveness of our closed-loop optimization approach that fuses geometric prior guidance with photometric refinement.

iii) **Real-time Performance.** We conduct runtime analysis as presented in Tab. III. The results demonstrate that our system achieves real-time performance by applying joint optimization only at keyframes. Our selective optimization strategy enables efficient online reconstruction while maintaining high-quality tracking and mapping performance.

C. Ablation Study

To validate the contribution of each component in our system, we conduct comprehensive ablation studies as presented in Tab. IV.

i) **Loop Closure.** Removing loop closure detection and pose graph optimization leads to significant degradation in tracking accuracy, demonstrating that our system's pose estimation substantially benefits from global consistency enhancement.

ii) **Primitive Sampling.** Replacing our direct primitive sampling strategy with uniform sampling approaches employed in most previous works [31] results in notable rendering quality deterioration, confirming the effectiveness of our sampling method.

iii) **Scene Priors.** Eliminating feed-forward scene priors and using randomly initialized depth values as adopted by previous monocular methods [5] causes severe decline in both rendering and tracking performance, validating the importance of geometric constraints for monocular 3DGS-based SLAM.

iv) **Camera Priors.** Most critically, removing learned camera priors and instead using raw intrinsic estimates with the last keyframe's pose for new frame initialization leads to catastrophic degradation in both rendering and tracking performance. This result underscores the essential role of

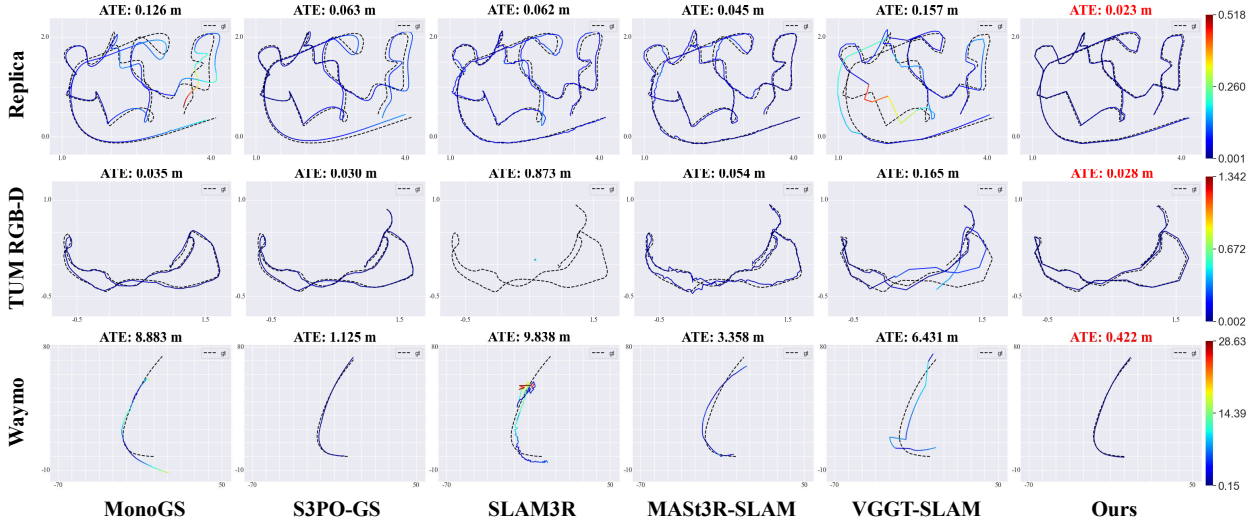

Fig. 4: **Comparison of estimated trajectories.** Each tracking result is projected onto the x-y plane, with ground truth shown as a dashed line. Our method achieves superior accuracy and robustness across indoor and outdoor environments.

TABLE II: **Tracking accuracy comparison across three datasets.** ATE RMSE [m] (↓) is reported. Our method achieves superior tracking performance compared to methods operating on uncalibrated RGB input (*Uncalib.*), while maintaining comparable accuracy against approaches utilizing calibrated camera parameters (*Calib.*).

	Method	Replica	TUM RGB-D	Waymo
Calib.	GlORIE-SLAM [43]	0.039	**0.046**	0.500
	MonoGS [5]	0.348	0.302	7.077
	Photo-SLAM [32]	0.022	0.325	**0.366**
	Splat-SLAM [44]	**0.018**	0.052	0.495
	S3PO-GS [34]	0.075	0.121	0.869
Uncalib.	SLAM3R [11]	0.064	0.652	18.265
	MASt3R-SLAM [10]	0.045	0.059	2.431
	VGGT-SLAM [12]	0.177	0.121	7.517
	GeoGS-SLAM (Ours)	**0.024**	**0.035**	**0.861**

learned camera priors in addressing the inherent scale ambiguity and convergence challenges of monocular SLAM.

V. CONCLUSIONS

In this paper we propose GeoGS-SLAM, a novel monocular SLAM system that integrates 3DGS with learned geometric priors. Our method achieves SOTA performance against existing monocular SLAM systems based on radiance fields and geometric priors across various benchmarks. Our work establishes a new paradigm for visual SLAM by building a closed-loop pipeline that leverages feed-forward priors for geometric bootstrapping while preserving photometric evidence for radiance field rendering-based optimization to achieve high-fidelity online reconstruction.

TABLE III: **Runtime evaluation results.** Average processing time per keyframe across datasets is reported, demonstrating the computational efficiency of our approach.

Step (per keyframe)	Time [ms]
Keyframe decision	5.8
Prior prediction	57.2
Scale alignment	0.5
Primitive sampling & Map expansion	9.2
Joint optimization	480.6
Loop closure detection	6.5
Pose graph optimization	0.6
Global map adjustment	21.7
Sum	**582.1**

TABLE IV: **Ablation study results on Replica.** Each row shows the impact of removing a component from the system.

Method	PSNR↑	SSIM↑	LPIPS↓	ATE↓
w/o loop closure	37.18	0.967	0.041	0.034
w/o primitive sampling	35.62	0.954	0.065	**0.024**
w/o scene priors	22.28	0.799	0.345	0.052
w/o camera priors	15.94	0.673	0.671	0.922
Ours	**37.56**	**0.968**	**0.039**	**0.024**

REFERENCES

[1] F. Tosi, Y. Zhang, Z. Gong, E. Sandström, S. Mattoccia, M. R. Oswald, and M. Poggi, "How nerfs and 3d gaussian splatting are reshaping slam: a survey," *arXiv preprint arXiv:2402.13255*, vol. 4, p. 1, 2024.

[2] J. Zhang, Y. Li, A. Chen, M. Xu, K. Liu, J. Wang, X.-X. Long, H. Liang, Z. Xu, H. Su *et al.*, "Advances in feed-forward 3d reconstruction and view synthesis: A survey," *arXiv preprint arXiv:2507.14501*, 2025.

[3] 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.

[4] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[5] 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.

[6] 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, 2024, pp. 21 357–21 366.

[7] S. Yu, C. Cheng, Y. Zhou, X. Yang, and H. Wang, "Rgb-only gaussian splatting slam for unbounded outdoor scenes," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 11 068–11 074.

[8] 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.

[9] 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.

[10] R. Murai, E. Dexheimer, and A. J. Davison, "Mast3r-slam: Real-time dense slam with 3d reconstruction priors," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 16 695–16 705.

[11] Y. Liu, S. Dong, S. Wang, Y. Yin, Y. Yang, Q. Fan, and B. Chen, "Slam3r: Real-time dense scene reconstruction from monocular rgb videos," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 16 651–16 662.

[12] D. Maggio, H. Lim, and L. Carlone, "Vggt-slam: Dense rgb slam optimized on the sl (4) manifold," *arXiv preprint arXiv:2505.12549*, 2025.

[13] 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.

[14] 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*. IEEE, 2012, pp. 573–580.

[15] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine *et al.*, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2446–2454.

[16] G. Klein and D. Murray, "Parallel tracking and mapping for small ar workspaces," in *2007 6th IEEE and ACM international symposium on mixed and augmented reality*. IEEE, 2007, pp. 225–234.

[17] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, "Orb-slam: A versatile and accurate monocular slam system," *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[18] R. Mur-Artal and J. D. Tardós, "Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1255–1262, 2017.

[19] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam," *IEEE transactions on robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[20] J. Engel, T. Schöps, and D. Cremers, "Lsd-slam: Large-scale direct monocular slam," in *European conference on computer vision*. Springer, 2014, pp. 834–849.

[21] J. Engel, V. Koltun, and D. Cremers, "Direct sparse odometry," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 3, pp. 611–625, 2017.

[22] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE international symposium on mixed and augmented reality*. Ieee, 2011, pp. 127–136.

[23] A. Dai, M. Nießner, M. Zollhöfer, S. Izadi, and C. Theobalt, "Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface reintegration," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, p. 1, 2017.

[24] 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)*, 2023, pp. 9400–9406.

[25] Z. Xin, Y. Yue, L. Zhang, and C. Wu, "Hero-slam: Hybrid enhanced robust optimization of neural slam," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 8610–8616.

[26] E. Sucar, S. 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*, 2021, pp. 6229–6238.

[27] 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*, 2022, pp. 12 786–12 796.

[28] X. Yang, H. Li, H. Zhai, Y. Ming, Y. Liu, and G. Zhang, "Vox-fusion: Dense tracking and mapping with voxel-based neural implicit representation," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2022, pp. 499–507.

[29] 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 *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 9748–9754.

[30] D. Yang, Y. Gao, X. Wang, Y. Yue, Y. Yang, and M. Fu, "Opengs-slam: Open-set dense semantic slam with 3d gaussian splatting for object-level scene understanding," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 8486–8492.

[31] 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.

[32] 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*, 2024, pp. 21 584–21 593.

[33] Z. Xin, C. Wu, P. Huang, Y. Zhang, Y. Mao, and G. Huang, "Large-scale gaussian splatting slam," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025, pp. 8478–8485.

[34] C. Cheng, S. Yu, Z. Wang, Y. Zhou, and H. Wang, "Outdoor monocular slam with global scale-consistent 3d gaussian pointmaps," *arXiv preprint arXiv:2507.03737*, 2025.

[35] 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.

[36] J. Yang, A. Sax, K. J. Liang, M. Henaff, H. Tang, A. Cao, J. Chai, F. Meier, and M. Feiszli, "Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 21 924–21 935.

[37] A. Meuleman, I. Shah, A. Lanvin, B. Kerbl, and G. Drettakis, "On-the-fly reconstruction for large-scale novel view synthesis from unposed images," *ACM Transactions on Graphics (TOG)*, vol. 44, no. 4, pp. 1–14, 2025.

[38] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 207, no. 1167, pp. 187–217, 1980.

[39] 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.

[40] G. Berton and C. Masone, "Megaloc: One retrieval to place them all," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 2861–2867.

[41] 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.

[42] Y. Zhang, F. Tosi, S. Mattoccia, and M. Poggi, "Go-slam: Global optimization for consistent 3d instant reconstruction," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 3727–3737.

[43] G. Zhang, E. Sandström, Y. Zhang, M. Patel, L. Van Gool, and M. R. Oswald, "Glorie-slam: Globally optimized rgb-only implicit encoding point cloud slam," *arXiv preprint arXiv:2403.19549*, 2024.

[44] E. Sandström, G. Zhang, K. Tateno, M. Oechsle, M. Niemeyer, Y. Zhang, M. Patel, L. Van Gool, M. Oswald, and F. Tombari, "Splat-slam: Globally optimized rgb-only slam with 3d gaussians," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 1680–1691.

[45] C. Homeyer, L. Begiristain, and C. Schnörr, "Droid-splat: Combining end-to-end slam with 3d gaussian splatting," *arXiv preprint arXiv:2411.17660*, 2024.