

Stereo 3D Gaussian Splatting SLAM for Outdoor Urban Scenes

Xiaohan Li^{1*}, Ziren Gong^{2*}, Fabio Tosi², Matteo Poggi², Stefano Mattoccia², Dong Liu¹, Jun Wu³

¹University of Science and Technology of China,

²University of Bologna,

³Fudan University,

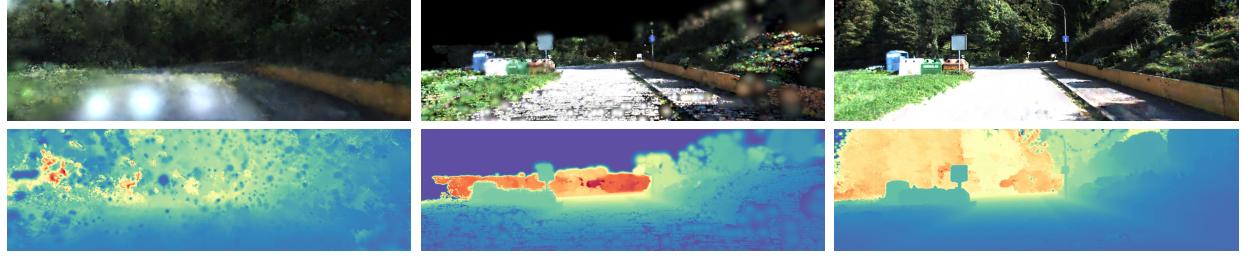


Figure 1: **Comparison of Rendering and Depth Estimation.** The top row shows RGB renderings generated by SplaTAM (Keetha et al. 2024), our BGS-SLAM method trained with LiDAR depth points, and our approach using only stereo RGB pairs with depth maps from deep stereo networks for supervision. The bottom row presents the corresponding depth renderings.

Abstract

3D Gaussian Splatting (3DGS) has recently gained popularity in SLAM applications due to its fast rendering and high-fidelity representation. However, existing 3DGS-SLAM systems have predominantly focused on indoor environments and relied on active depth sensors, leaving a gap for large-scale outdoor applications. We present BGS-SLAM, the first binocular 3D Gaussian Splatting SLAM system designed for outdoor scenarios¹. Our approach uses only RGB stereo pairs without requiring LiDAR or active sensors. BGS-SLAM leverages depth estimates from pre-trained deep stereo networks to guide 3D Gaussian optimization with a multi-loss strategy enhancing both geometric consistency and visual quality. Experiments on multiple datasets demonstrate that BGS-SLAM achieves superior tracking accuracy and mapping performance compared to other 3DGS-based solutions in complex outdoor environments.

Introduction

Simultaneous Localization and Mapping (SLAM), a core research area in computer vision, has been widely applied in autonomous driving, metaverse, and robotics. It primarily utilizes sensor data to estimate the state of a robot while simultaneously constructing an accurate scene representation. Traditional methods (Campos et al. 2021; Wang, Schworer, and Cremers 2017; Li, Liu, and Wu 2024) typically formulate this as a maximum a posteriori (MAP) estimation problem, where both robot ego-motion and scene modeling are described as factors in a graph for joint optimization.

In recent years, neural rendering-based methods have made significant advancements. The emergence of Neural Radiance Fields (NeRF) (Mildenhall et al. 2021) has profoundly influenced the community by revolutionizing novel view synthesis and scene representation, shifting the focus towards data-driven and differentiable rendering methods. Lately, 3D Gaussian Splatting (3DGS) (Kerbl et al. 2023) has emerged as a promising alternative. By representing scenes as a collection of 3D Gaussians and leveraging an efficient rasterization strategy, 3DGS achieves fast rendering while providing high-quality scene representation. This naturally aligns with SLAM’s requirements for real-time processing and accurate scene reconstruction, making 3DGS-SLAM (Keetha et al. 2024; Matsuki et al. 2024a) a rapidly growing research focus in recent years. However, existing 3DGS-SLAM methods mainly rely on dense and accurate depth maps from RGB-D sensors as training supervision for geometric reconstruction, while also being limited to small-scale indoor scenes. These methods achieve high-fidelity representations in controlled indoor environments but encounter severe challenges in large, complex outdoor settings.

First, active depth sensors like LiDAR and RGB-D cameras have inherent limitations outdoors. LiDAR systems are expensive, bulky, and power-intensive, making them impractical for many applications. Consumer RGB-D sensors like Microsoft Kinect or Intel RealSense, while more affordable and compact, face even greater limitations outdoors. These devices have significantly shorter effective ranges (typically under 5 meters), and their infrared-based depth sensing becomes unreliable in direct sunlight due to interference with their projected patterns. Second, outdoor scenes typically

¹The code will be released in case of acceptance.

span much larger scales. For example, in the KITTI dataset (Geiger, Lenz, and Urtasun 2012a), trajectories often extend over kilometers, causing substantial memory consumption during scene reconstruction and making real-time, efficient large-scale mapping particularly challenging. Finally, outdoor environments frequently involve drastic viewpoint changes and limited frame overlap, resulting in insufficient optimization constraints. This leads to convergence difficulties and visual artifacts that destabilize training.

To address these challenges, we propose a novel 3DGS-based architecture specifically designed for large-scale outdoor environments such as autonomous driving scenarios. Our approach leverages passive RGB stereo cameras only, which are affordable and lightweight compared to expensive and cumbersome active sensors, and leverages the use of pre-trained deep stereo networks to generate dense depth maps that guide the training of 3D Gaussians, effectively overcoming the lack of reliable depth information in outdoor environments. Additionally, we employ an external tracker based on ORB-SLAM2 (Mur-Artal, Montiel, and Tardos 2015), which significantly optimizes the entire pipeline and improves the overall system performance.

To the best of our knowledge, we are the first to integrate deep binocular stereo networks with 3DGS-SLAM specifically tailored for outdoor scenarios. Compared to the sparse LiDAR point clouds, stereo also provides more complete scene coverage, while demonstrating strong generalization and robustness under challenging lighting (Tosi, Bartolomei, and Poggi 2025). Furthermore, conversely to ill-posed, single-view depth estimation approaches (Yang et al. 2024b,a; Ke et al. 2024), stereo still provides proper metric estimates, grounded in epipolar geometry. Our experiments confirm that even approximate depth estimations from these networks significantly enhance the optimization process by guiding the positioning of 3D Gaussians and preventing artifacts that typically occur when splats become trapped in incorrect geometric configurations.

In summary, our contributions are the following:

- We propose BGS-SLAM, the first 3D Gaussian Splatting SLAM system for outdoor environments using passive RGB stereo pairs only.
- We integrate pre-trained deep stereo networks for dense depth supervision in 3D Gaussian optimization, showing that passive stereo can effectively replace expensive active sensors for outdoor scene reconstruction.
- We introduce a combination of normal-based and smoothness losses alongside depth-from-stereo supervision to enhance geometric consistency, reduce artifacts, and improve overall mapping quality.
- We present experiments on multiple large-scale outdoor datasets, including KITTI and KITTI-360, demonstrating that our approach significantly surpasses existing 3DGS-SLAM methods in outdoor scenarios, achieving superior tracking, mapping accuracy, and visual quality.

Related Work

Our work builds upon neural radiance field-based SLAM (Tosi et al. 2024), particularly focusing on RGB-only meth-

ods and 3DGS for outdoor environments.

Neural Implicit Representations for SLAM. Neural implicit representations have revolutionized SLAM research. iMAP (Sucar et al. 2021) pioneered this integration by employing an MLP to map 3D coordinates to color and density. NICE-SLAM (Zhu et al. 2022) addressed scalability through hierarchical representation using multiple pre-trained MLPs. Vox-Fusion (Yang et al. 2022) combined traditional volumetric techniques with neural implicit representations, while Co-SLAM (Wang, Wang, and Agapito 2023) developed hybrid encodings for robust camera tracking.

For large-scale environments, GO-SLAM (Zhang et al. 2023b) implemented global optimization techniques including loop closure and bundle adjustment, whereas Point-SLAM (Sandström et al. 2023) introduced a dynamic neural point cloud representation that adapts point density based on scene complexity. Most of these methods, however, rely on RGB-D sensors, limiting outdoor applications.

RGB-only SLAM with External Supervision. RGB-only SLAM methods overcome depth ambiguity through various external supervision signals. DIM-SLAM (Li et al. 2023) employed neural implicit map representation with multi-resolution volume encoding and photometric warping loss. NICER-SLAM (Zhu et al. 2024b) incorporated monocular depth and normal supervision alongside RGB rendering losses. iMODE (Matsuki et al. 2023) utilized ORB-SLAM2 for camera pose estimation while enhancing reconstruction through depth-rendered geometry supervision. NeRF-VO (Naumann et al. 2024) combined DPVO tracking with DPT for depth estimation. Hi-SLAM (Zhang et al. 2023a) leveraged DROID-SLAM-based dense correspondence and monocular depth priors to address low-texture environments. Recent approaches include MoD-SLAM (Zhou et al. 2024), which enhanced depth estimation through DPT and ZoeDepth, and MGS-SLAM (Zhu et al. 2024a), which unified sparse visual odometry with 3DGS through MVS-derived depth supervision.

3D Gaussian Splattting for SLAM. 3D Gaussian Splattting (Kerbl et al. 2023) offers faster rendering capabilities and improved representation of complex scenes compared to NeRF-based approaches. MonoGS (Matsuki et al. 2024b) pioneered this paradigm shift by leveraging 3D Gaussians with splatting rendering techniques for a single moving camera. Concurrently, Photo-SLAM (Huang et al. 2023) integrated explicit geometric features with implicit texture representations within a hyper primitives map. SplatTAM (Keetha et al. 2024) represented scenes as collections of simplified 3D Gaussians for high-quality color and depth image rendering, while GS-SLAM (Yan et al. 2024) introduced an adaptive expansion strategy and coarse-to-fine tracking technique. Recent advances include HF-GS SLAM (Sun et al. 2024b), which proposed rendering-guided densification strategies, and CG-SLAM (Hu et al. 2024), which implemented an uncertainty-aware 3D Gaussian field. MM3DGS-SLAM (Sun et al. 2024a) expanded to multi-modal inputs including inertial measurements, while RTG-SLAM (Peng et al. 2024) addressed large-scale environments by enforcing binary opacity classifications. For monocular setups, MonoGS++ (Li et al. 2024) exploited

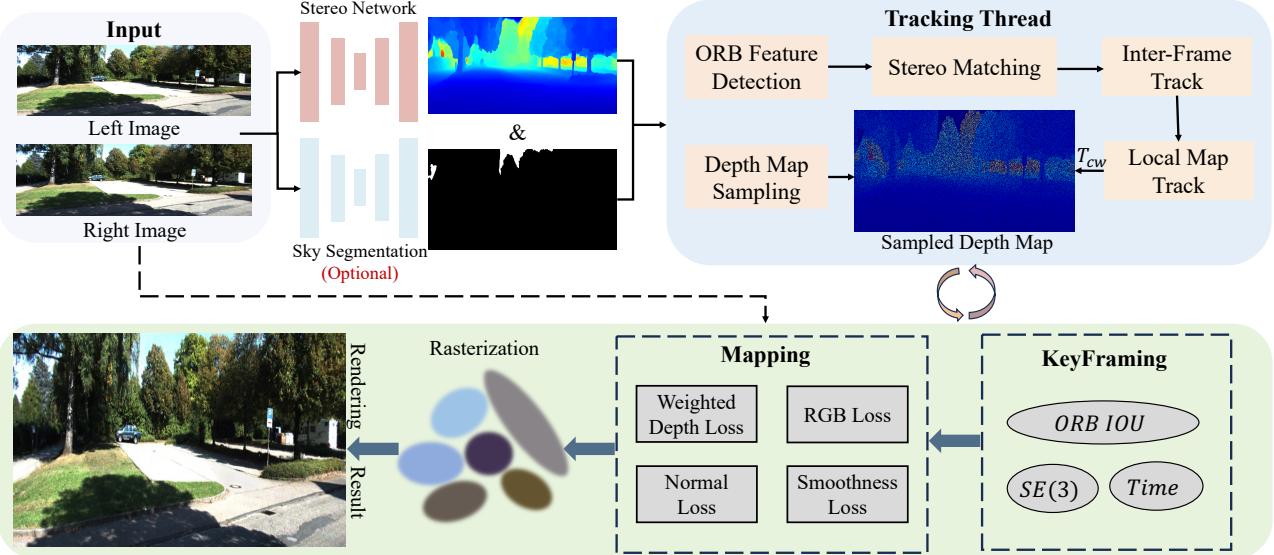


Figure 2: Framework Overview. BGS-SLAM uses stereo images to reconstruct outdoor environments using 3D Gaussians. A pre-trained stereo network extracts dense depth maps from the stereo pairs, with optional sky masking to improve reconstruction. The tracking thread estimates camera poses through feature matching and local bundle adjustment, while the keyframing thread maintains a buffer of key observations. In the mapping thread, a combination of depth, normal, and smoothness losses supervise the 3D Gaussian optimization, enhancing geometric consistency and visual quality of the reconstructed scenes.

DPVO (Teed, Lipson, and Deng 2022) as an external tracker to estimate initial camera poses. However, in outdoor environments, LIV-GaussMap (Hong and et al. 2024) and MM-Gaussian (Wu et al. 2024) still relied on LiDAR sensors for accurate depth measurements. Our work bridges this gap by utilizing stereo RGB images and recent deep stereo matching networks (Tosi, Bartolomei, and Poggi 2025) to predict depth maps that supervise 3D Gaussian Splatting optimization, enabling high-quality SLAM in outdoor environments without relying on active depth sensors.

Methods

We detail our binocular BGS-SLAM approach for outdoor environments in this section, with an overview in Fig. 2.

3D Gaussian Splatting (3DGS)

BGS-SLAM models the scene as a set of 3D Gaussians, denoted as $\mathcal{G} = \{g_1, g_2, \dots, g_N\}$, where N is the number of Gaussians. Each 3D Gaussian g_i is parameterized by both appearance attributes (color \mathbf{c}_i represented by spherical harmonics and opacity $o_i \in [0, 1]$), geometric properties (center position $\boldsymbol{\mu}_i \in \mathbb{R}^3$ and covariance matrix $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3 \times 3}$) parameters. The spatial influence of each Gaussian is defined as:

$$g_i(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1} (\mathbf{x}-\boldsymbol{\mu}_i)} \quad (1)$$

where the covariance matrix $\boldsymbol{\Sigma}_i = \mathbf{R}\mathbf{S}\mathbf{S}^\top\mathbf{R}^\top$ with $\mathbf{S} \in \mathbb{R}^3$ representing the spatial scale and $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ the rotation, parameterized by a quaternion.

In the rendering process, 3D Gaussians are first projected onto the 2D camera plane as:

$$\boldsymbol{\mu}' = \mathbf{J}\mathbf{W}\boldsymbol{\mu}, \quad \boldsymbol{\Sigma}' = \mathbf{J}\mathbf{W}\boldsymbol{\Sigma}\mathbf{W}^\top\mathbf{J}^\top \quad (2)$$

where \mathbf{W} is the rotational component of the viewing transformation \mathbf{T}_{cw} and \mathbf{J} is the Jacobian matrix which performs linear approximation of the projective transformation.

With alpha blending, the color and depth at each pixel are generated through:

$$C_p = \sum_{i=1}^n \mathbf{c}_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \quad D_p = \sum_{i=1}^n d_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j) \quad (3)$$

where the opacity α_i is:

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

During optimization, the parameters of all observed 3D Gaussians are iteratively refined through backpropagation using our mapping losses. For more details, please refer to (Kerbl et al. 2023; Chen and Wang 2024; Tosi et al. 2024).

Deep Stereo Depth Estimation

Given a pair of synchronized stereo images, I_{left} and I_{right} , we employ a pre-trained deep stereo network, f_θ , to estimate a dense disparity map d for outdoor environments:

$$d = f_\theta(I_{\text{left}}, I_{\text{right}}). \quad (5)$$

Modern stereo matching networks have evolved into advanced architectures that can be broadly classified into three main categories: Convolutional Neural Network (CNN)-based cost volume aggregation (Mayer et al. 2016; Kendall et al. 2017), transformer-based models (Li et al. 2021), and iterative optimization approaches (Lipson, Teed, and Deng

2021). These networks typically construct a cost volume $C(d)$ by establishing pixel-wise correspondences between the stereo image pair. Depending on the architecture, this correspondence computation can be performed using correlation layers, absolute or relative feature differences, or direct feature concatenation across the disparity range:

$$C(d) = \Psi(\Phi(I_{\text{left}}), T_d(\Phi(I_{\text{right}}))), \quad d \in [d_{\min}, d_{\max}], \quad (6)$$

where $\Phi(\cdot)$ denotes a feature extraction function, $T_d(\cdot)$ represents a disparity-dependent shift operation, and $\Psi(\cdot)$ defines the cost computation mechanism, which varies based on the network architecture (e.g., feature concatenation, subtraction, or correlation).

In our framework, we use pre-trained stereo networks that exhibit strong generalization across diverse environments, e.g., recent foundation models trained on extensive datasets (Wen et al. 2025; Cheng et al. 2025; Bartolomei et al. 2025; Jiang et al. 2025), allowing us to obtain reliable depth estimates without requiring additional domain-specific training. Disparity maps are converted to metric depth \tilde{D} via the standard stereo triangulation formula:

$$\tilde{D} = \frac{f \cdot b}{d}, \quad (7)$$

where f is the focal length and b the stereo baseline. These dense depth maps provide rich geometric supervision for our 3DGS optimization process, offering significant advantages over LiDAR-based depth estimation methods due to their higher spatial density and more complete scene coverage.

Sky Segmentation

3DGS often generates ambiguous floaters when rendering sky, leading to a significant number of unnecessary Gaussians and violating multi-view consistency. However, the sky typically occupies only a small portion of outdoor scenes, and the depth map in these regions tends to be inaccurate. To address this, we integrate a sky segmentation network (Xie et al. 2021) into our pipeline, which unifies transformers with a lightweight MLP. Notably, the sky segmentator is optional and designed to enhance system stability.

Tracking

While 3D Gaussian Splatting offers remarkable rendering capabilities, it faces challenges in large-scale outdoor environments due to computational demands and convergence issues. To cope with these issues, we adopt an external tracker based on ORB-SLAM2 (Mur-Artal and Tardós 2017) for several key reasons: (1) feature-based methods like ORB-SLAM2 provide robust real-time tracking even in challenging outdoor conditions with varying illumination and viewpoints, (2) the sparse feature matching approach is computationally efficient compared to the dense optimization required for direct 3D Gaussian optimization, and (3) decoupling the tracking from the mapping thread allows us to maintain stable pose estimation while the 3D Gaussian representation is still being optimized.

Specifically, at time i , the left image I_{left} and corresponding right image I_{right} are processed into our tracking thread. Following ORB-SLAM2 (Mur-Artal and Tardós 2017), the ORB features are extracted from both the left and right images which ensures the real-time performance compared to other feature points. Given the camera-inherent \mathbf{K} , the ORB features from the left image are searched for matches against those from the right image, yielding a set of stereo correspondences. For the following frames, the incremental camera pose is first initialized under a constant velocity motion model. Then, the matched stereo features are incorporated into a frame-level bundle adjustment (BA). The loss for the frame-level BA can be expressed as:

$$\min_{\mathbf{T}_i} \sum_{i \in \mathcal{O}} \rho \left([\mathbf{z}_i - \pi_s(\mathbf{T}_i, \mathbf{X}_i)]^T \Omega_i [\mathbf{z}_i - \pi_s(\mathbf{T}_i, \mathbf{X}_i)] \right), \quad (8)$$

where \mathbf{T}_i is the camera pose of current frame i , \mathbf{z}_i is the observed stereo measurement, \mathbf{X}_i is the corresponding 3D mappoint, $\pi_s(\cdot)$ is the stereo projection function, Ω_i is the inverse covariance matrix, and $\rho(\cdot)$ is a robust kernel cost function. Once the incremental ego-motion is estimated, a sliding window of selected keyframes is activated for local bundle adjustment optimization. By constructing a local BA cost function over these keyframes, the tracking thread further reduce the reprojection error and improve the accuracy of ego-motion estimation. Formally, the local BA is:

$$\min_{\{\mathbf{T}_k\}, \{\mathbf{X}_j\}} \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{O}_k} \rho \left(\mathbf{e}_{kj}^T \Omega_{kj} \mathbf{e}_{kj} \right), \quad (9)$$

$$\mathbf{e}_{kj} = \mathbf{z}_{kj} - \pi(\mathbf{T}_k, \mathbf{X}_j). \quad (10)$$

where $\{\mathbf{T}_k\}$ are the keyframe poses within the sliding window, $\{\mathbf{X}_j\}$ are the 3D mappoints visible across keyframes, and \mathbf{z}_{kj} is the observation of point j in keyframe k . This local BA jointly optimizes camera poses and 3D structure, providing accurate ego-motion estimates crucial for our mapping thread. Precise pose estimation ensures proper alignment between 3D Gaussians and stereo depth maps, preventing the uncontrolled expansion of Gaussians that would otherwise lead to artifacts and excessive memory consumption in large outdoor scenes.

KeyFraming

In BGS-SLAM, we integrate a keyframing module to ensure the robustness of our system. Most of the recent 3DGS-SLAM systems determine the keyframes according to time intervals. This approach is able to uniformly add keyframes, but the ego-motion in outdoor scenes rarely exhibits linear changes over time and contains intense camera movements. Thus, relying solely on time intervals often leads to inadequate scene overlaps and catastrophic forgetting in outdoor scenes. The keyframing module first assesses the covisibility from the intersection over union (IoU) of the observed ORB keypoints extracted with inter-frames. If the IoU falls below the threshold, the current frame I_i is registered as a new keyframe. To address system instability caused by intense camera movements, the inter-frame transformation is evaluated to determine whether a significant movement occurs. If a large motion change appears, a new keyframe is

inserted into the keyframe buffer \mathcal{T} . In our paper, the intense camera movement is adaptively defined as 1.5 times the previous motion change:

$$\mathbf{T}_{\mathcal{W}C}^t (\mathbf{T}_{\mathcal{W}C}^{t-1})^{-1} > 1.5 \times \mathbf{T}_{\mathcal{W}C}^{t-1} (\mathbf{T}_{\mathcal{W}C}^{t-2})^{-1} \rightarrow \mathcal{T}. \quad (11)$$

To maintain computational efficiency, the system retains a limited number of keyframes within the sliding window to optimize both the ego-motion and 3D Gaussian representations. Moreover, a keyframe will be marginalized from the sliding window if the keypoint IoU with the most recent keyframe drops below a specified threshold.

Mapping

The mapping thread receives camera poses from the tracking thread and stereo-estimated depth maps. It represents the scene as a collection of 3D Gaussians optimized through several complementary loss functions:

RGB Loss. We supervise the color reconstruction using a combination of L1 and structural similarity (SSIM) losses:

$$L_{color} = \lambda_{rgb} \|I(\mathcal{G}, T_{CW}) - I\|_1 + \lambda_{ssim} L_{ssim}(I(\mathcal{G}, T_{CW}), I), \quad (12)$$

where $I(\mathcal{G}, T_{CW})$ and I are the rendered and real RGB images, respectively.

Weighted Geometric Loss. While RGB loss is essential, it provides insufficient supervision for outdoor scenes with large textureless regions. Therefore, we introduce a weighted depth loss that incorporates RGB gradient information:

$$\mathcal{L}_{geo} = g_{rgb} \frac{1}{n} \sum \log \left(1 + M \|D_{s_i} - \hat{D}_{s_i}\|_1 \right) \quad (13)$$

where $g_{rgb} = \exp(-\nabla I)$, ∇I is the RGB image gradient, n is the number of pixels, M is a mask for valid depth values, D_{s_i} is the stereo network-estimated depth map, and \hat{D}_{s_i} is the rendered depth map.

Based on empirical observations in our experimental validation, we discovered that uniform sampling of stereo depth supervision significantly improves reconstruction quality compared to using the full depth map. This sampled depth loss is formally defined as:

$$\mathcal{L}_{geo}^{\text{sampled}} = g_{rgb} + \frac{1}{|S|} \sum_{(i,j) \in S} \log \left(1 + M |D_{s_{i,j}} - \hat{D}_{s_{i,j}}|_1 \right) \quad (14)$$

where $S \subset \{1, \dots, H\} \times \{1, \dots, W\}$ represents a uniform subset of pixel coordinates. For our implementation, we sample approximately 25% of the total pixels using a regular grid pattern. This approach offers several advantages: it reduces the influence of locally correlated errors in stereo depth estimation and promotes smoother optimization by effectively regularizing the supervision signal.

Normal Consistency Loss. To enhance geometric supervision, we compute normal vectors from both the stereo-estimated and rendered depth maps, and enforce consistency between them:

$$\mathcal{L}_n = \frac{1}{|H|} \sum \|N - \hat{N}\|_1, \quad (15)$$

Methods	PSNR↑	SSIM↑	LPIPS↓	Depth L1↓
w/o ss	24.19	0.92	0.11	141.71
w/o wd	23.86	0.91	0.12	284.36
w/o nl	24.25	0.92	0.11	193.20
w/o sl	23.82	0.91	0.12	186.46
with ds	21.17	0.84	0.22	263.68
Ours	24.82	0.93	0.10	136.10

Table 1: **Ablation Study on the KITTI dataset.** We analyze the effectiveness of sky segmentator (ss), weighted depth loss (wd), normal loss (nl), smoothness loss (sl) and dense depth map supervision (ds) in our proposed SLAM system.

Backbones	PSNR ↑	SSIM ↑	LPIPS ↓	Depth L1 ↓
IGEV	23.34	0.90	0.14	278.20
IGEV++	23.19	0.89	0.15	293.57
TCSM	23.16	0.89	0.15	300.29
MonSter-K	23.33	0.90	0.15	186.32
Mocha	24.17	0.91	0.12	462.00
FoundationStereo	24.57	0.92	0.11	132.65
MonSter-M	24.82	0.93	0.10	136.10

Table 2: **Ablation Study on Stereo Network Selection.** Evaluation of our method's performance using different stereo networks. Depth L1 is in [cm], backbones in the upper part are fine-tuned on KITTI datasets, while backbones in the bottom part are trained on a mix of datasets.

where \hat{N} is the rendered normal map, N is the normal map derived from stereo depth estimation, and H denotes the number of patches with valid normal values.

Smoothness Loss. To ensure geometric consistency, we introduce smoothness loss that penalized abrupt changes in the normal map:

$$\mathcal{L}_s = \frac{1}{|n|} \sum_r \sum_{i,j} \left(\|\hat{N}_{i+r,j} - \hat{N}_{i,j}\| + \|\hat{N}_{i,j+r} - \hat{N}_{i,j}\| \right). \quad (16)$$

where n is the number of valid depth values in the rendered depth map.

Final Loss. The final mapping loss combines these components with appropriate weights:

$$L_{mapping} = \lambda_{rgb} L_{rgb} + \lambda_{ssim} L_{ssim} + \lambda_{geo} L_{geo}^{\text{sampled}} + \lambda_n L_n + \lambda_s L_s \quad (17)$$

where we set $\lambda_{rgb} = 0.8$, $\lambda_{ssim} = 0.2$, $\lambda_{geo} = 0.1$, $\lambda_n = 0.1$, and $\lambda_s = 0.5$ in our experiments.

Experiments

Experimental Setup

Datasets. We evaluate BGS-SLAM on the KITTI (Geiger, Lenz, and Urtasun 2012b) and the KITTI-360 datasets (Liao, Xie, and Geiger 2022). Both datasets provide rich sensor data from a vehicle platform with stereo cameras, Velodyne LiDAR, GPS, and IMU, covering diverse driving scenarios including urban areas, residential streets, and highways under varying illumination conditions. We focus on

Methods	Metrics (km/frames)	03 (0.56/801)	05 (2.2/2761)	06 (1.2/1101)	07 (0.69/1101)	09 (1.7/1591)	10 (0.92/1201)	Average
Point-SLAM†	ATE ↓	81.51	104.61	170.73	79.00	138.50	102.81	112.86
	PSNR ↑	9.09	12.58	4.33	11.89	11.69	8.26	9.64
	SSIM ↑	0.30	0.48	0.24	0.47	0.37	0.38	0.37
	LPIPS ↓	0.74	0.66	0.89	0.64	0.71	0.69	0.72
	Depth-L1 ↓	227.89	428.29	405.80	211.97	248.12	306.25	304.72
SplaTAM†	ATE ↓	10.20	37.13	53.78	32.82	70.23	33.96	39.69
	PSNR ↑	14.26	14.78	16.40	16.05	15.91	14.18	15.26
	SSIM ↑	0.47	0.48	0.55	0.63	0.54	0.45	0.52
	LPIPS ↓	0.56	0.53	0.46	0.43	0.52	0.56	0.51
	Depth-L1 ↓	277.91	319.99	474.61	355.67	673.33	277.86	396.89
MonoGS†	ATE ↓	57.87	51.77	92.81	51.23	81.23	61.96	66.14
	PSNR ↑	10.40	12.20	11.15	10.94	12.65	12.71	11.67
	SSIM ↑	0.25	0.37	0.28	0.38	0.42	0.38	0.35
	LPIPS ↓	0.71	0.65	0.76	0.67	0.71	0.68	0.70
	Depth-L1 ↓	681.49	403.81	575.07	568.34	666.94	674.65	595.05
BGS-SLAM (Lidar)	ATE ↓	1.77	1.86	0.90	0.60	4.76	3.70	2.26
	PSNR ↑	11.87	6.92	10.47	8.04	8.08	10.40	9.30
	SSIM ↑	0.56	0.37	0.47	0.39	0.31	0.45	0.42
	LPIPS ↓	0.63	0.70	0.66	0.67	0.71	0.64	0.67
	Depth-L1 ↓	234.77	286.10	303.22	273.60	381.24	256.28	289.20
BGS-SLAM (Ours)	ATE ↓	1.77	1.86	0.90	0.60	4.76	3.70	2.26
	PSNR ↑	24.82	19.16	23.57	20.14	18.99	22.56	21.54
	SSIM ↑	0.93	0.72	0.87	0.78	0.73	0.85	0.81
	LPIPS ↓	0.10	0.29	0.15	0.22	0.30	0.18	0.21
	Depth-L1 ↓	136.10	253.03	226.89	192.99	371.00	161.70	223.62

Table 3: **Quantitative Evaluation on the KITTI dataset.** Our BGS-SLAM is evaluated on the whole image recorded on the sequences. Methods indicated with †fail to process the entire recorded image and therefore, their performance is reported on the first 300 frames of all sequences. MonoGS is reported in RGB-D mode. ATE RMSE [m]↓, Depth L1 [cm]↓ and bold numbers indicate the best result.

the KITTI Odometry split, which contains 22 sequences. Among them, we randomly select 6 sequences with ground truth poses, with trajectory lengths ranging from 0.56 km to 2.2 km. We additionally evaluate on KITTI-360, which offers expanded coverage and complexity, selecting multiple sequences with different scene scales and dynamics to validate BGS-SLAM’s robustness and scalability under more complex outdoor conditions.

Evaluation Metrics. We evaluate BGS-SLAM on tracking and mapping. For tracking, we report the RMSE of Absolute Trajectory Error (ATE). For rendering quality, we follow radiance-field-based SLAM methods and report PSNR, SSIM (Wang et al. 2004), and LPIPS (Zhang et al. 2018). Geometric accuracy is measured via Depth L1 error between rendered depth maps and LiDAR ground truth.

Implementation Details. All experiments, including BGS-SLAM and baselines, are run on a desktop with Intel(R) Xeon(R) Gold 6326 CPU @ 2.90GHz and a NVIDIA A40 GPU with 48Gb memory. We adopt ORB-SLAM2 (Mur-Artal and Tardós 2017) as the external tracker for robust pose estimation. For stereo depth, we use the publicly available MonSter (Cheng et al. 2025) network with pre-trained weights. For learning rate in 3DGS mapping, color is set to 2.5×10^{-3} , rotation and scale to 10^{-3} , opacity to 5×10^{-2} , and the opacity removal threshold to 5×10^{-3} . All results are averaged over three runs.

Ablation Study

Component Analysis. In Table 1, we present ablation experiments on a KITTI sequence to validate each component of our approach. Our full system achieves the best overall performance across all metrics. Removing the sky segmentation module (“w/o ss”) leads to decreased visual quality metrics (PSNR -0.63dB) by introducing inaccurate supervision from sky regions. Without the weighted depth loss (“w/o wd”), depth accuracy deteriorates substantially (Depth L1 increases by 148.26 cm), while maintaining reasonable visual quality, highlighting its importance for geometric reconstruction. The absence of normal loss (“w/o nl”) or smoothness loss (“w/o sl”) results in increased depth errors and slightly reduced rendering quality, confirming their role in enhancing structural details. Using dense depth maps without our selective supervision strategy (“with ds”) performs worse, demonstrating that balancing supervision signals is crucial.

Stereo Network Analysis. Table 2 reports the performance of BGS-SLAM integrated with various state-of-the-art stereo matching networks. The upper section of the table includes models fine-tuned on the KITTI dataset (IGEV (Xu et al. 2023), IGEV++(Xu et al. 2024), TCSM(Zeng et al. 2024), and MonSter-K (Cheng et al. 2025), while the lower section includes models trained on a broader mix of datasets (Mocha (Chen et al. 2024), FoundationStereo (Wen et al. 2025), and MonSter-M (Cheng et al. 2025)). Notably, compared with KITTI-only models, networks trained on multiple datasets exhibit overall superior performance across both

Methods	Metrics (km/frames)	0002 (11.5/14k)	0004 (9.97/11.6k)	0005 (4.69/6.7k)	0007 (4.89/3.4k)	0008 (7.13/8.8k)	0009 (10.58/14k)	Average
Point-SLAM†	ATE ↓	99.56	161.56	56.07	247.38	99.53	159.80	137.32
	PSNR ↑	12.65	7.90	12.90	12.21	7.16	5.83	9.77
	SSIM ↑	0.47	0.38	0.43	0.41	0.18	0.26	0.35
	LPIPS ↓	0.69	0.70	0.67	0.73	0.92	0.89	0.77
	Depth-L1 ↓	371.20	485.35	720.09	746.07	494.01	676.43	582.19
SplaTAM†	ATE ↓	56.19	67.55	23.96	138.98	58.12	57.13	66.99
	PSNR ↑	12.53	12.39	12.24	13.02	13.33	11.94	12.57
	SSIM ↑	0.29	0.34	0.31	0.34	0.36	0.40	0.34
	LPIPS ↓	0.60	0.57	0.61	0.58	0.58	0.58	0.59
	Depth-L1 ↓	492.51	586.81	684.68	727.28	501.30	724.98	619.59
MonoGS†	ATE ↓	43.91	79.70	31.11	177.12	52.08	103.45	81.23
	PSNR ↑	11.23	12.10	11.00	11.43	10.63	11.12	11.25
	SSIM ↑	0.32	0.38	0.33	0.32	0.30	0.47	0.35
	LPIPS ↓	0.73	0.69	0.70	0.68	0.69	0.67	0.69
	Depth-L1 ↓	616.64	681.48	818.60	797.51	629.82	880.14	737.36
BGS-SLAM (Lidar)	ATE ↓	3.43	3.25	2.81	2.77	5.55	6.28	4.01
	PSNR ↑	10.54	10.27	10.99	10.21	10.07	9.04	10.19
	SSIM ↑	0.40	0.40	0.38	0.39	0.36	0.32	0.38
	LPIPS ↓	0.68	0.64	0.68	0.66	0.69	0.71	0.68
	Depth-L1 ↓	291.01	312.82	407.84	387.22	385.43	445.84	371.69
BGS-SLAM (Ours)	ATE ↓	3.43	3.25	2.81	2.77	5.55	6.28	4.01
	PSNR ↑	23.24	24.68	24.93	24.29	24.36	20.47	23.66
	SSIM ↑	0.87	0.90	0.91	0.88	0.89	0.80	0.87
	LPIPS ↓	0.18	0.14	0.14	0.17	0.15	0.26	0.17
	Depth-L1 ↓	314.80	215.93	285.88	457.50	306.47	428.89	334.91

Table 4: **Quantitative Evaluation on the KITTI-360 dataset.** Our BGS-SLAM is evaluated on the whole image recorded on the sequences. Methods indicated with †fail to process the entire image and is reported on the first 300 frames of all sequences. MonoGS is reported in RGB-D mode. Note that in the ”(km/frames)” row, ”k” is used as a shorthand for 1,000 frames.

Methods	03	05	06	07	09	10
SplaTAM	X	X	X	X	X	X
BGS-SLAM (Ours)	8.08	45.12	16.77	14.81	20.36	20.62

Table 5: **Memory Consumption Analysis (GB).** X indicates that the method fails to process full sequences, running out of memory after few hundreds frames.

rendering quality metrics and geometric accuracy. In particular, MonSter-M achieves the best PSNR (24.82), while maintaining a low depth error (136.10 cm), significantly outperforming models such as IGEV (278.20 cm) and TCSM (300.29 cm). Furthermore, these multi-dataset models exhibit stronger zero-shot generalization capability, which is critical for long-term SLAM deployment in unseen environments. We therefore adopt MonSter-M as our default stereo network for optimal accuracy-generalization trade-off.

Comparison with State-of-The-Art SLAM

Tracking and Mapping Performance. We compare BGS-SLAM against state-of-the-art radiance field-based SLAM methods on the KITTI and KITTI-360 datasets in terms of tracking accuracy and mapping quality. Quantitative results are reported in Table 3 and Table 4. Due to memory constraints, methods like SplaTAM (Keetha et al. 2024), MonoGS (Matsuki et al. 2024a), and Point-SLAM (Sandström et al. 2023) were evaluated only on the first 300 frames per sequence. However, their tracking threads showed large pose estimation errors in outdoor environments, limiting their applicability in real-world large-scale scenes. In con-

trast, our tracking, grounded in classical SLAM pose estimation, provides robust and accurate performance even in complex, large-scale scenarios.

For mapping and view synthesis, BGS-SLAM substantially outperforms all baselines across all visual metrics, achieving an average PSNR improvement of over 6 dB in KITTI dataset. In KITTI-360 dataset, BGS-SLAM achieves more than a 10 dB improvement in PSNR and the best depth reconstruction accuracy with the lowest Depth L1 error.

Fig. 3 illustrates the rendering performance of BGS-SLAM vs. baselines. A LiDAR-supervised variant is also shown, highlighting its advantages over active sensors in outdoor settings. Compared to SplaTAM, MonoGS, and Point-SLAM, our method achieves the highest fidelity and continuity in large-scale outdoor scenes.

Memory Efficiency. Table. 5 shows the memory consumption analysis for all methods. SplaTAM fails to process complete KITTI sequences even on an NVIDIA A40 GPU with 48GB of memory, whereas BGS-SLAM succeeds at it.

Conclusion

In this paper, we present BGS-SLAM, the first 3DGSSLAM system for outdoor scenarios using only stereo RGB input. Our novel contributions include leveraging pre-trained deep stereo networks for depth supervision and introducing a multi-loss optimization strategy that combines RGB, depth, normal, and smoothness losses to enhance geometric consistency and novel view synthesis quality. Experiments on KITTI and KITTI-360 demonstrate that BGS-SLAM achieves superior tracking and mapping quality com-



Figure 3: Visualization of rendering quality on KITTI.

pared to existing radiance-field SLAM approaches without requiring expensive LiDAR sensors.

Limitations. BGS-SLAM does not yet operate in real-time, with average tracking and mapping times of 0.24 s and 1.37 s per frame, respectively—posing a limitation for practical SLAM applications. The computational overhead is further increased by the inference time of the deep stereo network, in addition to the iterative optimization of 3D Gaussians for each frame.

References

- Bartolomei, L.; Tosi, F.; Poggi, M.; and Mattoccia, S. 2025. Stereo anywhere: Robust zero-shot deep stereo matching even where either stereo or mono fail. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 1013–1027.
- Campos, C.; Elvira, R.; Rodríguez, J. J. G.; Montiel, J. M.; and Tardós, J. D. 2021. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE Transactions on Robotics*, 37(6): 1874–1890.
- Chen, G.; and Wang, W. 2024. A Survey on 3D Gaussian Splatting. *arXiv preprint arXiv:2401.03890*.
- Chen, Z.; Long, W.; Yao, H.; Zhang, Y.; Wang, B.; Qin, Y.; and Wu, J. 2024. Mocha-stereo: Motif channel attention network for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 27768–27777.
- Cheng, J.; Liu, L.; Xu, G.; Wang, X.; Zhang, Z.; Deng, Y.; Zang, J.; Chen, Y.; Cai, Z.; and Yang, X. 2025. Monster: Marry monodepth to stereo unleashes power. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 6273–6282.
- Geiger, A.; Lenz, P.; and Urtasun, R. 2012a. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE conference on computer vision and pattern recognition*, 3354–3361. IEEE.
- Geiger, A.; Lenz, P.; and Urtasun, R. 2012b. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Hong, S.; and et al. 2024. LIV-GaussMap: LiDAR-Inertial-Visual Fusion for Real-time 3D Radiance Field Map Rendering. *IEEE Robotics and Automation Letters*.
- Hu, J.; Chen, X.; Feng, B.; Li, G.; Yang, L.; Bao, H.; Zhang, G.; and Cui, Z. 2024. CG-SLAM: Efficient Dense RGB-D SLAM in a Consistent Uncertainty-aware 3D Gaussian Field. In *European Conference on Computer Vision (ECCV)*.
- Huang, H.; Li, L.; Cheng, H.; and Yeung, S.-K. 2023. Photo-SLAM: Real-time Simultaneous Localization and Photorealistic Mapping for Monocular, Stereo, and RGB-D Cameras. *arXiv preprint arXiv:2311.16728*.
- Jiang, H.; Lou, Z.; Ding, L.; Xu, R.; Tan, M.; Jiang, W.; and Huang, R. 2025. Defom-stereo: Depth foundation model based stereo matching. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 21857–21867.
- Ke, B.; Obukhov, A.; Huang, S.; Metzger, N.; Daudt, R. C.; and Schindler, K. 2024. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9492–9502.
- Keetha, N.; Karhade, J.; Jatavallabhula, K. M.; Yang, G.; Scherer, S.; Ramanan, D.; and Luiten, J. 2024. Splatam: Splat track & map 3d gaussians for dense rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 21357–21366.
- Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, P.; Kennedy, R.; Bachrach, A.; and Bry, A. 2017. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE international conference on computer vision*, 66–75.
- Kerbl, B.; Kopanas, G.; Leimkühler, T.; and Drettakis, G. 2023. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Transactions on Graphics*, 42(4).
- Li, H.; Gu, X.; Yuan, W.; Yang, L.; Dong, Z.; and Tan, P. 2023. Dense RGB SLAM With Neural Implicit Maps. In *Proceedings of the International Conference on Learning Representations*.
- Li, R.-W.; Ke, W.; Li, D.; Tian, L.; and Barsoum, E. 2024. MonoGS++: Fast and Accurate Monocular RGB Gaussian SLAM. In *British Conference on Machine Vision (BMVC)*.
- Li, X.; Liu, D.; and Wu, J. 2024. CTO-SLAM: contour tracking for object-level robust 4D SLAM. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 10323–10331.
- Li, Z.; Liu, X.; Drenkow, N.; Ding, A.; Creighton, F. X.; Taylor, R. H.; and Unberath, M. 2021. Revisiting stereo depth estimation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, 6197–6206.

- Liao, Y.; Xie, J.; and Geiger, A. 2022. Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3292–3310.
- Lipson, L.; Teed, Z.; and Deng, J. 2021. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *2021 International Conference on 3D Vision (3DV)*, 218–227. IEEE.
- Matsuki, H.; Murai, R.; Kelly, P. H.; and Davison, A. J. 2024a. Gaussian splatting slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 18039–18048.
- Matsuki, H.; Murai, R.; Kelly, P. H. J.; and Davison, A. J. 2024b. Gaussian Splatting SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.
- Matsuki, H.; Sucar, E.; Laidow, T.; Wada, K.; Scona, R.; and Davison, A. J. 2023. iMODE: Real-Time Incremental Monocular Dense Mapping Using Neural Field. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 4171–4177. IEEE.
- Mayer, N.; Ilg, E.; Hausser, P.; Fischer, P.; Cremers, D.; Dosovitskiy, A.; and Brox, T. 2016. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4040–4048.
- Mildenhall, B.; Srinivasan, P. P.; Tancik, M.; Barron, J. T.; Ramamoorthi, R.; and Ng, R. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1): 99–106.
- Mur-Artal, R.; Montiel, J. M. M.; and Tardos, J. D. 2015. ORB-SLAM: a versatile and accurate monocular SLAM system. *IEEE transactions on robotics*, 31(5): 1147–1163.
- Mur-Artal, R.; and Tardós, J. D. 2017. Orb-slam2: An open-source slam system for monocular, stereo, and rgbd cameras. *IEEE transactions on robotics*, 33(5): 1255–1262.
- Naumann, J.; Xu, B.; Leutenegger, S.; and Zuo, X. 2024. NeRF-VO: Real-Time Sparse Visual Odometry With Neural Radiance Fields. *IEEE Robotics and Automation Letters*.
- Peng, Z.; Shao, T.; Liu, Y.; Zhou, J.; Yang, Y.; Wang, J.; and Zhou, K. 2024. Rtg-slam: Real-time 3d reconstruction at scale using gaussian splatting. In *ACM SIGGRAPH 2024 Conference Papers*, 1–11.
- Sandström, E.; Li, Y.; Van Gool, L.; and R. Oswald, M. 2023. Point-SLAM: Dense Neural Point Cloud-based SLAM. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.
- Sucar, E.; Liu, S.; Ortiz, J.; and Davison, A. J. 2021. iMAP: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 6229–6238.
- Sun, L. C.; Bhatt, N. P.; Liu, J. C.; Fan, Z.; Wang, Z.; Humphreys, T. E.; and Topcu, U. 2024a. MM3DGS SLAM: Multi-modal 3D Gaussian Splatting for SLAM Using Vision, Depth, and Inertial Measurements. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- Sun, S.; Mielle, M.; Lilenthal, A. J.; and Magnusson, M. 2024b. High-Fidelity SLAM Using Gaussian Splatting with Rendering-Guided Densification and Regularized Optimization. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.
- Teed, Z.; Lipson, L.; and Deng, J. 2022. Deep patch visual odometry. *arXiv preprint arXiv:2208.04726*.
- Tosi, F.; Bartolomei, L.; and Poggi, M. 2025. A Survey on Deep Stereo Matching in the Twenties. *International Journal of Computer Vision*.
- Tosi, F.; Zhang, Y.; Gong, Z.; Sandström, E.; Mattoccia, S.; Oswald, M. R.; and Poggi, M. 2024. How nerfs and 3d gaussian splatting are reshaping slam: a survey. *arXiv preprint arXiv:2402.13255*, 4: 1.
- Wang, H.; Wang, J.; and Agapito, L. 2023. Co-SLAM: Joint Coordinate and Sparse Parametric Encodings for Neural Real-Time SLAM. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 13293–13302.
- Wang, R.; Schworer, M.; and Cremers, D. 2017. Stereo DSO: Large-scale direct sparse visual odometry with stereo cameras. In *Proceedings of the IEEE international conference on computer vision*, 3903–3911.
- Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4): 600–612.
- Wen, B.; Trepte, M.; Aribido, J.; Kautz, J.; Gallo, O.; and Birchfield, S. 2025. Foundationstereo: Zero-shot stereo matching. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 5249–5260.
- Wu, C.; Duan, Y.; Zhang, X.; Sheng, Y.; Ji, J.; and Zhang, Y. 2024. MM-Gaussian: 3D Gaussian-based Multi-modal Fusion for Localization and Reconstruction in Unbounded Scenes. In *RSJ International Conference on Intelligent Robots and Systems (IROS)*.
- Xie, E.; Wang, W.; Yu, Z.; Anandkumar, A.; Alvarez, J. M.; and Luo, P. 2021. SegFormer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34: 12077–12090.
- Xu, G.; Wang, X.; Ding, X.; and Yang, X. 2023. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 21919–21928.
- Xu, G.; Wang, X.; Zhang, Z.; Cheng, J.; Liao, C.; and Yang, X. 2024. IGEV++: iterative multi-range geometry encoding volumes for stereo matching. *arXiv preprint arXiv:2409.00638*.
- Yan, C.; Qu, D.; Xu, D.; Zhao, B.; Wang, Z.; Wang, D.; and Li, X. 2024. GS-SLAM: Dense Visual SLAM with 3D Gaussian Splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Yang, L.; Kang, B.; Huang, Z.; Xu, X.; Feng, J.; and Zhao, H. 2024a. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, 10371–10381.

Yang, L.; Kang, B.; Huang, Z.; Zhao, Z.; Xu, X.; Feng, J.; and Zhao, H. 2024b. Depth anything v2. *Advances in Neural Information Processing Systems*, 37: 21875–21911.

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

Zeng, J.; Yao, C.; Wu, Y.; and Jia, Y. 2024. Temporally consistent stereo matching. In *European Conference on Computer Vision*, 341–359. Springer.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 586–595.

Zhang, W.; Sun, T.; Wang, S.; Cheng, Q.; and Haala, N. 2023a. Hi-slam: Monocular real-time dense mapping with hybrid implicit fields. *IEEE Robotics and Automation Letters*.

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

Zhou, H.; Guo, Z.; Ren, Y.; Liu, S.; Zhang, L.; Zhang, K.; and Li, M. 2024. MoD-SLAM: Monocular Dense Mapping for Unbounded 3D Scene Reconstruction.

Zhu, P.; Zhuang, Y.; Chen, B.; Li, L.; Wu, C.; and Liu, Z. 2024a. MGS-SLAM: Monocular Sparse Tracking and Gaussian Mapping with Depth Smooth Regularization. *IEEE Robotics and Automation Letters*.

Zhu, Z.; Peng, S.; Larsson, V.; Cui, Z.; Oswald, M. R.; Geiger, A.; and Pollefeys, M. 2024b. NICER-SLAM: Neural Implicit Scene Encoding for RGB SLAM. In *International Conference on 3D Vision (3DV)*.

Zhu, Z.; Peng, S.; Larsson, V.; Xu, W.; Bao, H.; Cui, Z.; Oswald, M. R.; and Pollefeys, M. 2022. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12786–12796.