Advances in Computer, Signals and Systems (2026)
Clausius Scientific Press, Canada

DOI: 10.23977/acss.2026.100112
ISSN 2371-8838 Vol. 10 Num. 1

A Review of Common Datasets and Algorithms for 3D Gaussian Splatting SLAM in Dynamic Scenes

Enbo Zhang

Yunnan Normal University, Kunming, Yunnan, China
2517339360@qq.com

Keywords: 3D Gaussian Splatting, SLAM, Dynamic Scenes

Abstract: Simultaneous Localization and Mapping (SLAM) is a fundamental technology for autonomous navigation, augmented reality, and virtual reality. However, dynamic factors widely present in real-world environments, such as pedestrians, vehicles, and movable objects, severely violate the static-world assumption adopted by traditional SLAM methods, posing significant challenges to localization accuracy, map consistency, and long-term stability. In recent years, 3D Gaussian Splatting (3DGS) has attracted increasing attention due to its compact and efficient scene representation, favorable differentiability, and excellent real-time rendering performance. It provides a novel technical paradigm for SLAM in dynamic environments and has gradually become a research hotspot. This paper presents a systematic review of 3D Gaussian SLAM algorithms in dynamic scenes. First, commonly used benchmark datasets are analyzed, including the TUM and BONN datasets for indoor dynamic environments and the KITTI dataset for outdoor scenarios. Comparisons are conducted in terms of scene scale, dynamic object types, and evaluation protocols. Second, commonly used evaluation metrics for pose accuracy, rendering quality, and system efficiency are summarized. Third, representative dynamic-scene 3D Gaussian SLAM algorithms are reviewed, and their core ideas and technical characteristics for handling dynamic interference are systematically analyzed. Finally, existing challenges are discussed, and future research directions are outlined.

1. Introduction

With the rapid development of robotics, autonomous driving, and augmented reality (AR), intelligent agents are required to achieve higher levels of accuracy, real-time performance, and robustness in environmental perception. Simultaneous Localization and Mapping (SLAM), as a core technology for spatial understanding, enables agents to estimate their poses and construct maps in unknown environments, serving as a foundation for autonomous navigation and intelligent decision-making. Under static-environment assumptions, traditional feature-based and direct SLAM methods have achieved mature performance and wide application.

However, real-world environments usually contain numerous dynamic factors, such as pedestrians, vehicles, and movable objects. These dynamic elements introduce erroneous feature correspondences, depth inconsistencies, and geometric variations, leading to pose drift, map corruption, and even tracking failure. This severely limits the applicability of SLAM in complex environments. To address these challenges, researchers have proposed various strategies, including semantic-based or motion-consistency-based dynamic object detection and removal, as well as explicit modeling of dynamic objects. Nevertheless, most existing approaches rely on discrete geometric representations such as point clouds, voxels, or meshes, which suffer from limitations in modeling efficiency, dynamic updating cost, and rendering quality.

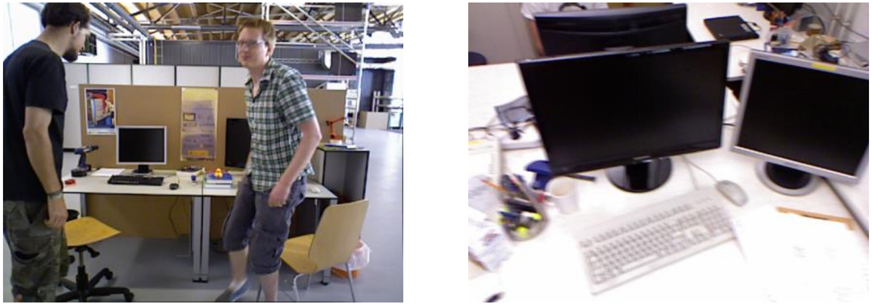
The emergence of 3D Gaussian Splatting[1] has brought new opportunities to dynamic SLAM research. By representing scenes using anisotropic Gaussian primitives, 3DGS provides continuous, compact, and differentiable representations with high-quality real-time rendering capabilities. These properties make it particularly suitable for modeling object deformations, scene variations, and multi-view consistency. By integrating 3D Gaussian modeling with SLAM frameworks, 3D Gaussian SLAM systems offer a promising solution to overcome the limitations of traditional SLAM in dynamic environments and have become an important research direction.

Despite the rapid progress in this field, a comprehensive review is still lacking. Therefore, this paper systematically reviews research on 3D Gaussian SLAM in dynamic scenes, including datasets, algorithmic frameworks, technical components, advantages, limitations, and future directions.

2. Commonly Used Datasets

2.1 TUM Dynamic Dataset

The TUM RGB-D Dataset[2], as shown in Figure 1. released by the Technical University of Munich, is one of the most influential datasets in visual SLAM and 3D reconstruction. It provides synchronized RGB and depth images captured by RGB-D sensors such as Microsoft Kinect. Depth maps are stored in 16-bit PNG format in meters. Accurate ground-truth camera poses are obtained using a motion capture system and provided as timestamps, translations, and quaternions. The dataset consists of three subsets: freiburg1, freiburg2, and freiburg3, covering scenarios from simple static scenes to complex dynamic environments with fast motion, illumination changes, and occlusions. Sequences with strong dynamic interference are widely used to evaluate the robustness of dynamic 3D Gaussian SLAM systems.



(a) (b)

Figure 1. Pictures of TUM dataset[2]

2.2 BONN Dataset

The BONN dataset[3], as shown in Figure 2. released by the University of Bonn, is designed for RGB-D SLAM in highly dynamic environments. It provides synchronized RGB images, depth images, and accurate ground-truth poses. Compared with TUM, BONN emphasizes highly dynamic scenes containing large-scale non-rigid human motions and movable objects. It covers diverse motion patterns and dynamic intensities, making it suitable for evaluating dynamic object detection, motion segmentation, and robust mapping. Due to its realistic modeling of complex dynamics, the BONN dataset serves as an important benchmark for dynamic 3D Gaussian SLAM.

(a) (b)

Figure 2. Pictures of BONN dataset[3]

2.3 KITTI Dataset

The KITTI dataset[4], released by KIT and Toyota Technological Institute at Chicago, is a large-scale real-world dataset for autonomous driving and robotic perception. It includes monocular and stereo cameras, LiDAR, and GPS/IMU sensors.Unlike TUM and BONN, KITTI focuses on large-scale outdoor dynamic environments characterized by long viewing distances, rapid motion, and numerous traffic participants. High-quality pose ground truth and multi-task annotations make it a key benchmark for evaluating robustness and scalability in outdoor scenarios.

3. Evaluation Metrics

3D Gaussian SLAM systems are typically responsible for two core tasks: camera pose estimation and scene reconstruction. Therefore, their performance evaluation not only focuses on localization accuracy but also involves rendering quality and system efficiency. In existing studies, commonly used evaluation metrics can be categorized into three main groups: pose estimation accuracy metrics, rendering quality metrics, and system efficiency metrics.

3.1 Pose Accuracy

Pose estimation accuracy is a fundamental indicator for evaluating the performance of 3D Gaussian SLAM systems, as it directly reflects tracking stability and global consistency in dynamic environments.

Absolute Trajectory Error (ATE) is used to measure the overall deviation between the estimated camera trajectory and the ground-truth trajectory in the global coordinate system. It is usually reported in the form of Root Mean Square Error (RMSE). In 3D Gaussian SLAM, ATE reflects the accumulated error during long-term operation and is sensitive to loop closure detection and global optimization strategies. Therefore, it effectively evaluates map consistency and global localization accuracy. This metric is widely used in trajectory evaluation on datasets such as TUM and EuRoC. Relative Pose Error (RPE) evaluates the relative pose estimation error between consecutive frames or within fixed time intervals, mainly reflecting the accuracy of local motion estimation. Compared with ATE, RPE focuses more on the stability of front-end tracking and is more sensitive to rapid motion, dynamic disturbances, and short-term mismatches. In dynamic environments, RPE can intuitively indicate the system's robustness to instantaneous dynamic interference.

3.2 Rendering Quality

Rendering quality is one of the key advantages that distinguishes 3D Gaussian-based methods from traditional geometric SLAM approaches. It is also an important criterion for evaluating the quality of Gaussian maps and novel view synthesis performance. Quantitative evaluation is usually conducted by comparing rendered images with ground-truth images.

PSNR (Peak Signal-to-Noise Ratio) is a classical image quality assessment metric based on pixel-wise Mean Squared Error (MSE). It measures the overall error between rendered images and real observations, where higher values indicate closer similarity. In 3D Gaussian SLAM, PSNR is commonly used to evaluate novel-view rendering quality and is one of the most basic and widely adopted quantitative metrics. However, it mainly focuses on pixel-level errors and has limited consistency with human visual perception. SSIM (Structural Similarity Index) measures the similarity between two images in terms of luminance, contrast, and structural information, making it more consistent with human visual perception. In 3D Gaussian rendering, SSIM effectively reflects the preservation of object edges and structural details and is sensitive to blur and structural distortions. It is often used in conjunction with PSNR for comprehensive quality evaluation. LPIPS (Learned Perceptual Image Patch Similarity) is a perceptual similarity metric based on deep neural network features. It measures the difference between two images at high-level semantic and perceptual levels. In 3D Gaussian SLAM, LPIPS is commonly used to evaluate visual realism and fine-detail preservation, and it is more sensitive to texture variations and perceptual quality. Lower LPIPS values indicate better rendering performance.

3.3 System Efficiency

In addition to accuracy and quality, system efficiency is an important dimension for evaluating the practicality of 3D Gaussian SLAM, especially in real-time applications.

Training and Optimization Time measures the total time required from system initialization to model convergence, reflecting the overall optimization efficiency. When comparing 3D Gaussian methods with implicit representation approaches such as NeRF, training time is often regarded as a major advantage. Rendering Speed is usually measured in frames per second (FPS) or rendering time per frame (milliseconds) and is used to evaluate novel-view rendering performance. Higher rendering frame rates indicate better suitability for real-time SLAM and online applications. Model Size is commonly measured by the number of Gaussian primitives or GPU memory consumption, reflecting the compactness of scene representation and resource usage. Proper control of the number of Gaussians is a critical prerequisite for achieving real-time mapping in large-scale dynamic environments.

4. 3D Gaussian SLAM Algorithms

4.1 Overview of 3D Gaussian Splatting

3D Gaussian Splatting represents a three-dimensional scene as a set of anisotropic Gaussian distributions, enabling a continuous and compact representation of the environment while supporting high-quality real-time rendering. Its core idea is to convert point clouds or surface representations into parameterized Gaussian primitives, generate images through forward projection, and iteratively update Gaussian parameters via optimization algorithms to improve rendering consistency and geometric accuracy. Each Gaussian primitive can be modeled as a probability distribution in three-dimensional space:

$$G_i(\mathbf{x}) = c_i \exp\left(-\frac{1}{2}(\mathbf{x}- \mu_i)^T \Sigma_i^{-1}(\mathbf{x}-\mu_i)\right) \tag{1}$$

where $\mathbf{x}$ denotes the spatial location in 3D space, μ_i represents the center of the Gaussian, Σ_i is the covariance matrix that determines the shape and orientation of the Gaussian primitive, and c_i denotes the intensity or weight controlling its contribution to the overall rendering. By superimposing all Gaussian primitives, a continuous representation of the entire scene can be obtained:

$$S(\mathbf{x}) = \sum_{i=1}^{N} G_i(\mathbf{x}) \tag{2}$$

where N denotes the total number of Gaussian primitives. The core of the rendering process is to project three-dimensional Gaussian primitives onto the two-dimensional image plane and compute pixel colors. For each pixel, the rendered value is obtained through Gaussian projection and volumetric integration:

$$C(\mathbf{p}) = \sum_{i=1}^{N} w_i(\mathbf{p}) c_i \tag{3}$$

where $\mathbf{p}$ represents the pixel coordinates, $w_i(\mathbf{p})$ denotes the projection weight of the i^{th} Gaussian at the pixel location, which typically incorporates depth attenuation and occlusion handling, and c_i represents the color of the Gaussian primitive.

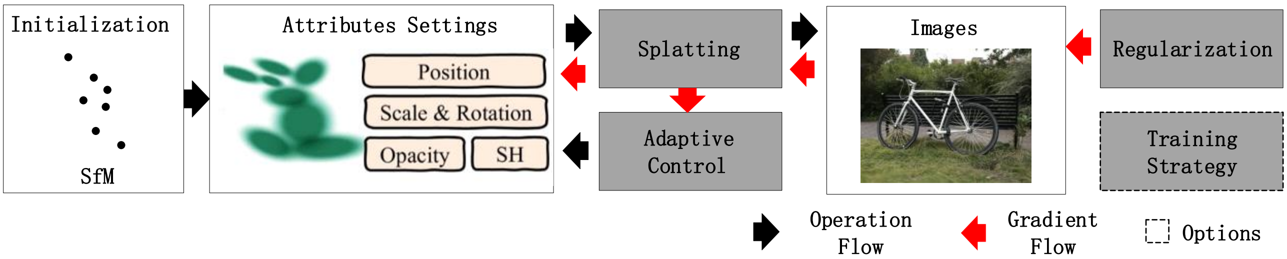


Figure 3. 3D Gaussian Overall Framework Diagram

The overall rendering process is shown in Figure 3, and the Gaussian projection process is fully differentiable. By minimizing the loss between rendered images and ground-truth observations, the Gaussian centers, covariance matrices, and color parameters are iteratively optimized. To balance rendering quality and computational efficiency, large-scale or low-density Gaussians are split to enhance fine details, while occluded Gaussians or those with negligible contributions are pruned to reduce computational cost. Furthermore, tile-based rendering strategies are employed to accelerate visibility determination and pixel projection of Gaussian primitives, thereby improving rendering frame rates. Depth sorting is applied to Gaussian primitives, and weighted blending strategies are adopted to handle front–back occlusion relationships, ensuring rendering consistency and visual realism.

4.2 Static Scene Reconstruction via Dynamic Removal

In recent years, numerous algorithms have been developed for 3D Gaussian SLAM in dynamic environments, with most of them aiming to achieve reliable static scene reconstruction through dynamic object detection and separation. Since identifying moving objects is a critical step, many approaches rely on semantic networks to detect pedestrians, vehicles, and other dynamic elements, generating motion masks to distinguish dynamic regions from static backgrounds. Nevertheless, these methods often fail to cover all dynamic objects in complex real-world scenarios, such as moved furniture or carried items. To address this limitation, several studies integrate geometric constraints or optical flow information with semantic cues for more robust dynamic modeling.

In 2024, Xu et al. introduced DG-SLAM[5], the first dynamic visual SLAM system based on 3D Gaussian Splatting. Semantic priors provided by OneFormer are combined with spatiotemporally consistent depth masks to generate reliable motion masks. Meanwhile, a hybrid coarse-to-fine pose optimization framework, integrating DROID-VO[6] and Gaussian-based photometric calibration, enhances pose-map consistency. Furthermore, adaptive Gaussian densification and pruning strategies are adopted to maintain high-quality static maps. Building upon instance segmentation and photometric analysis, Kong et al. developed DGS-SLAM[7], in which Track Anything and histogram-based cues are used to generate integrated dynamic masks. After removing dynamic features, only static regions are used for Gaussian map updates. In addition, loop-aware keyframe management and joint optimization with isotropic regularization are employed to suppress reconstruction artifacts. To improve dynamic-static discrimination, Huang et al. presented MPDG-SLAM[8], which performs instance segmentation using YOLOv7 and introduces a motion probability labeling mechanism that combines semantic and geometric information. Misclassified Gaussians are corrected through epipolar verification and map densification, while an MP-weighted rendering loss further reduces visual artifacts. A different strategy is adopted in DyPho-SLAM[9] proposed by Liu et al. By fusing prior image information, semantic segmentation, and optical-flow masks, optimized motion masks are generated. An iterative background modeling scheme reduces false detections, and adaptive feature extraction ensures sufficient constraints after dynamic region removal. Hu et al. proposed Hu_DyGS-SLAM[10], where YOLOX is used to detect potential moving objects. A Dynamic Feature Detection module integrates reprojection errors, depth residuals, and appearance similarity to identify unknown dynamic targets. Moreover, reconstruction quality in sparse areas is improved through texture-aware adaptive densification.

To overcome the limitations of category-dependent segmentation, some studies leverage text-prompt-based detection models in combination with SAM for fine-grained segmentation. In SDD-SLAM[11], Liu et al. utilize Grounding DINO to generate instance proposals from textual descriptions, covering both active and passive dynamic objects. These proposals are refined by SAM, enabling accurate pixel-level segmentation and stable temporal tracking. Subsequently, semantic–geometric joint optimization and object-level Gaussian removal are applied to achieve precise static reconstruction. Similarly, Zhu et al. proposed DyGS-SLAM[12], which adopts YOLO-World for open-set detection based on textual prompts. EfficientSAM is then employed to perform fast segmentation. To further improve reconstruction completeness, occluded background regions are recovered using ProPainter, and keyframe selection is applied to ensure reliable observations.

Despite their advantages, the real-time performance of such approaches is largely constrained by the inference speed of the segmentation networks.In contrast to semantic-driven approaches, several methods aim to reduce dependence on category information.

For example, Li et al. introduced GARAD-SLAM[13], which segments dynamic Gaussians using conditional random fields combined with reprojection and depth variation cues. Erroneous Gaussians are corrected through sparse optical-flow verification, while dynamic Gaussian penalty terms are incorporated into the optimization process. Rather than relying on predefined semantic categories, Wen et al. proposed Gassidy[14], where instance segmentation is only used to guide Gaussian initialization. Object and background components are modeled separately, and a photometric–geometric loss formulation is designed to amplify loss differences between dynamic and static regions. In ADD-SLAM[15], Wu et al. adopt MobileSAM as a lightweight segmentation tool. By using inconsistent regions as prompts, dynamic objects can be segmented in a category-agnostic manner. Continuous 2D tracking and temporal Gaussian modeling are further combined to achieve incremental dynamic reconstruction.

Although RGB-D inputs are commonly used, several studies explore monocular solutions.Li et al. proposed Dy3DGS-SLAM[16], which integrates optical flow segmentation and monocular depth estimation to generate fused motion masks. Bayesian modeling is employed for multi-object separation, while scale-constrained motion loss and photometric–depth penalties mitigate transient artifacts. Zhou et al.[17] enhanced monocular dynamic SLAM by combining pixel-set completion with YOLOv8-Seg and Zoe-Depth. Scene details are refined using non-uniform Gaussian pyramid optimization, and dynamic-aware residual losses suppress interference. In WildGS-SLAM[18], Zheng et al. introduce an uncertainty prediction module for monocular RGB data. Dynamic regions are distinguished using 3D-aware DINOv2 features and depth constraints, and uncertainty-guided optimization improves pose stability.

Some studies focus on large-scale outdoor reconstruction and improve system stability by integrating multiple sensors. Deng et al. proposed SGF-SLAM[19], in which the STDC-seg semantic segmentation network is adopted as the core semantic module and embedded into the multi-task front-end network SGF-net. This framework is specifically designed to recognize dynamic objects in urban road environments, such as vehicles and pedestrians, and outputs semantic labels and segmentation masks. For urban driving scenarios, SGF-SLAM combines semantic segmentation with coarse-to-fine (C2F) feature extraction. By introducing dynamic penalties and static rewards, feature points are encouraged to concentrate on static regions. In addition, a semantic Gaussian filtering mechanism is developed to smoothly blend colors from dynamic areas with background information, thereby optimizing map updating and reloading processes. Zhu et al. presented LVD-GS[20], which employs Grounded SAM—an open-world semantic segmentation framework integrating Grounded DINO and SAM. Through scene-aware prompt generation, various dynamic objects in outdoor environments are identified, providing semantic priors for dynamic feature localization. Furthermore, LiDAR–vision fusion is utilized to construct multi-scale collaborative representations. By combining explicit and implicit dynamic modeling modules, fine-grained motion masks are generated to assist Gaussian initialization and scan matching, leading to improved pose stability in dynamic scenes.

It is worth noting that most of the above approaches introduce dedicated strategies for keyframe selection. This is mainly due to the imbalance between localization and Gaussian-based mapping in SLAM systems, where pose tracking typically operates faster than map construction. When all tracking keyframes are directly used for mapping, the number of keyframes inevitably increases, resulting in degraded real-time performance. As the system continues to operate, excessive keyframes further cause computational bottlenecks and performance degradation. Consequently, various mechanisms have been developed to balance keyframe usage between tracking and mapping. Although removing dynamic regions is an effective way to adapt SLAM systems to dynamic environments, this straightforward strategy has inherent limitations. Most existing methods rely heavily on semantic priors, which introduces uncertainty when encountering object categories that are not included in the training set. Moreover, in highly complex scenarios where most or even all regions in the image are dynamic, these systems remain vulnerable to tracking failure.

4.3 Dynamic Scene Reconstruction

Recent studies have increasingly focused on explicitly tracking and modeling dynamic objects to improve overall tracking stability. Most of these approaches adopt a 4D mapping paradigm by incorporating temporal information into scene representations. By embedding dynamic elements into reconstructed maps, the expressive power and descriptive capability of scene models are significantly enhanced. In 4DTAM[21], non-rigid surfaces are represented using 2D Gaussian splatting. Non-rigid deformations are captured through explicit surface normal encoding and MLP-driven warp fields. To accelerate optimization, analytical camera pose Jacobians are employed, while quasi-rigid and normal regularization terms are introduced to enforce local consistency. A different strategy is adopted in D^{2}GSLAM[22], where geometric cues are integrated with YOLO-World and MobileSAM to generate accurate motion masks. Static and dynamic regions are jointly represented using a hybrid formulation consisting of static 3D Gaussians and dynamic 4D Gaussians. Transitions between static and dynamic states are realized through KNN-based association, and temporal continuity is maintained via backward-frame optimization and dynamic buffer management. In D4DGS-SLAM[23], 4D Gaussians are incorporated into dynamic scene modeling to capture spatiotemporal characteristics. Temporal means and covariance matrices are used to describe motion patterns, while an InfoModule filters highly dynamic and low-reliability points. Meanwhile, adaptive isotropic regularization is applied to adjust Gaussian distributions to different motion behaviors. The JPG-SLAM[24] framework adopts a joint point–Gaussian representation, in which static regions are modeled with 3D Gaussians and dynamic areas with 4D Gaussians. Interfering observations are removed through a dedicated dynamic region identification module. Moreover, local map management and joint optimization mechanisms are designed to achieve a balance between pose accuracy and reconstruction fidelity.

Beyond general dynamic scene modeling, several studies extend their focus to human bodies and foreground objects. In ODHSR[25], joint human–scene reconstruction is performed using monocular videos. Human motion is represented through an SMPL-driven hybrid Gaussian model that separates rigid and non-rigid components, whereas the surrounding environment is described using standard 3D Gaussians. Robust camera tracking and dynamic human modeling are achieved by jointly optimizing RGB, optical flow, monocular depth, and silhouette-based loss terms. The PG-SLAM[26] system addresses dynamic foreground modeling and static background reconstruction in a unified framework. Deformations of non-rigid humans and rigid objects are constrained using SMPL and MLP-based networks, while static backgrounds are optimized through multi-view appearance and geometric consistency. In addition, a two-stage camera localization strategy combined with human-scale regularization improves modeling stability and map completeness.

5. Conclusions

This paper reviews SLAM methods based on 3D Gaussian modeling in dynamic environments, focusing on benchmark datasets and representative algorithms. Typical datasets such as TUM, BONN, and KITTI are introduced, followed by a classification of key techniques, including dynamic object detection, scene modeling, and camera pose optimization. Existing approaches are analyzed in terms of efficiency, robustness, and long-term stability. Although 3D Gaussian SLAM provides continuous and differentiable scene representations and balances rendering quality with mapping accuracy, it still faces challenges such as reliance on semantic priors, limited dynamic detection accuracy, weak long-term modeling, and high computational costs. Future research may explore multimodal sensor fusion, more efficient Gaussian representations, and deeper integration of learning-based and optimization methods to improve real-time performance and long-term mapping. Overall, 3D Gaussian-based dynamic SLAM shows strong potential for autonomous navigation and robotic perception.

References

[1] Kerbl B, Kopanas G, Leimkühler T, et al. 3D Gaussian splatting for real-time radiance field rendering[J]. ACM Trans. Graph., 2023, 42(4): 139:1-139:14.
[2] Sturm J, Engelhard N, Endres F, et al. A benchmark for the evaluation of RGB-D SLAM systems[C]//2012 IEEE/RSJ international conference on intelligent robots and systems. IEEE, 2012: 573-580.
[3] Palazzolo E, Behley J, Lottes P, et al. ReFusion: 3D reconstruction in dynamic environments for RGB-D cameras exploiting residuals[C]//2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2019: 7855-7862.
[4] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? the kitti vision benchmark suite[C]//2012 IEEE conference on computer vision and pattern recognition. IEEE, 2012: 3354-3361.
[5] Xu Y, Jiang H, Xiao Z, et al. Dg-slam: Robust dynamic gaussian splatting slam with hybrid pose optimization[J]. Advances in Neural Information Processing Systems, 2024, 37: 51577-51596.
[6] Teed Z, Deng J. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras[J]. Advances in neural information processing systems, 2021, 34: 16558-16569.
[7] Kong M, Lee J, Lee S, et al. DGS-SLAM: Gaussian splatting SLAM in dynamic environment[J]. arXiv preprint arXiv:2411.10722, 2024.
[8] Huang C, Zhang L, Deng T, et al. MPDG-SLAM: Motion Probability-Based 3DGS-SLAM in Dynamic Environment[C]//2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2025: 14045-14052.
[9] Liu Y, Fan K, Lan B, et al. DyPho-SLAM: Real-time Photorealistic SLAM in Dynamic Environments[C]//2025 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2025: 1-6.
[10] Hu X, Zhang C, Zhao M, et al. DyGS-SLAM: Real-Time Accurate Localization and Gaussian Reconstruction for Dynamic Scenes[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2025: 9561-9571.
[11] Liu H, Wang L, Luo H, et al. SDD-SLAM: Semantic-Driven Dynamic SLAM with Gaussian Splatting[J]. IEEE Robotics and Automation Letters, 2025.
[12] Zhu F, Zhao Y, Chen Z, et al. DyGS-SLAM: Realistic Map Reconstruction in Dynamic Scenes Based on Double-Constrained Visual SLAM[J]. Remote Sensing, 2025, 17(4): 625.
[13] Li M, Chen W, Cheng N, et al. GARAD-SLAM: 3D GAussian splatting for Real-time Anti Dynamic SLAM[J]. arXiv preprint arXiv:2502.03228, 2025.
[14] Wen L, Li S, Zhang Y, et al. Gassidy: Gaussian splatting slam in dynamic environments[C]//2025 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2025: 8471-8477.
[15] Wu W, Su C, Zhu S, et al. ADD-SLAM: Adaptive Dynamic Dense SLAM with Gaussian Splatting[J]. arXiv preprint arXiv:2505.19420, 2025.
[16] Li M, Zhou Y, Zhou H, et al. Dy3DGS-SLAM: Monocular 3D Gaussian Splatting SLAM for Dynamic Environments[J]. arXiv preprint arXiv:2506.05965, 2025.
[17] Zhou H, Chen J, Li Z. Dynamic SLAM with 3D Gaussian Splatting Supporting Monocular Sensing[J]. IEEE Sensors Journal, 2025.
[18] Zheng J, Zhu Z, Bieri V, et al. Wildgs-slam: Monocular gaussian splatting slam in dynamic environments[C]//Proceedings of the Computer Vision and Pattern Recognition Conference. 2025: 11461-11471.
[19] Deng Z, Wang R. SGF-SLAM: Semantic Gaussian Filtering SLAM for Urban Road Environments[J]. Sensors, 2025, 25(12): 3602.
[20] Zhu W, Li X, Xu Q, et al. LVD-GS: Gaussian Splatting SLAM for Dynamic Scenes via Hierarchical Explicit-Implicit Representation Collaboration Rendering[J]. arXiv preprint arXiv:2510.22669, 2025.
[21] Matsuki H, Bae G, Davison A J. 4DTAM: Non-Rigid Tracking and Mapping via Dynamic Surface Gaussians[C]//Proceedings of the Computer Vision and Pattern Recognition Conference. 2025: 26921-26932.
[22] Zhu S, Huang Y, Wu W, et al. D^{2}GSLAM: 4D Dynamic Gaussian Splatting SLAM[J]. arXiv preprint arXiv:2512.09411, 2025.
[23] Sun Z, Lo J, Hu J. Embracing Dynamics: Dynamics-aware 4D Gaussian Splatting SLAM[J]. arXiv preprint arXiv:2504.04844, 2025.
[24] Huang K, Yang W, Zhou P, et al. JPG-SLAM: Joint point-Gaussian splatting representation for dense dynamic SLAM[C]//2025 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2025: 5018-5024.
[25] Zhang Z, Kaufmann M, Xue L, et al. ODHSR: Online Dense 3D Reconstruction of Humans and Scenes from Monocular Videos[C]//Proceedings of the Computer Vision and Pattern Recognition Conference. 2025: 21824-21835.
[26] Li H, Meng X, Zuo X, et al. PG-SLAM: Photo-realistic and geometry-aware RGB-D SLAM in dynamic environments[J]. IEEE Transactions on Robotics, 2025.