

Article

SFSE-SLAM: Semantic-Geometric Filtering and Texture-Aware Compensation for Robust Visual SLAM in Dynamic Indoor Scenes

Yao Wang ^{1,2}, Changzhong Pan ^{1,2,3,*} , Chaoyang Chen ^{1,2,3}, Simon X. Yang ⁴  and Zhijing Li ²

¹ Sanya Institute of Hunan University of Science and Technology, Sanya 572024, China

² School of Information and Electrical Engineering, Hunan University of Science and Technology, Xiangtan 411201, China

³ Hunan Provincial Key Laboratory of Intelligent Control and Maintenance for Complex Systems, Hunan University of Science and Technology, Xiangtan 411201, China

⁴ Advanced Robotics and Intelligent Systems Laboratory, College of Engineering, University of Guelph, Guelph, ON N1G2W1, Canada

* Correspondence: pancz@hnust.edu.cn

Abstract

Visual simultaneous localization and mapping in indoor dynamic environments remains challenging because moving objects introduce unreliable correspondences, whilst removing dynamic feature points often leaves insufficient static features in low-texture regions. This paper proposes SFSE-SLAM (Semantic Feature Selection and Enhancement SLAM), a robust visual SLAM framework that combines semantic-geometric feature filtering with texture-aware feature compensation to improve pose estimation under dynamic interference. The framework first identifies potentially dynamic regions through pixel-level semantic segmentation and removes features associated with highly dynamic objects. To reduce over-filtering and address semi-static objects, depth variation and multi-view geometric consistency are further used to distinguish static and moving feature points across consecutive frames. After dynamic filtering, a learned local feature extractor is introduced to improve descriptor discriminability and feature density in reliable static regions. Two additional modules, semantic confidence weighting and static region feature compensation, adaptively adjust feature extraction thresholds so that low-texture but geometrically useful areas can contribute more stable correspondences. The proposed system is evaluated on public dynamic RGB-D benchmarks, including the TUM RGB-D dataset (featuring high-dynamic human motions) and the Bonn dataset (containing complex non-human dynamic objects such as thrown balloons and moved boxes). Experimental results indicate that the method improves localization robustness in high-dynamic scenarios and reduces trajectory error compared with conventional ORB-based SLAM and several dynamic SLAM baselines. Specifically, on the highly dynamic TUM fr3-walking-xyz sequence, our method achieves an ATE RMSE of 0.0148 m, representing a 97.9% reduction compared to ORB-SLAM3. The study demonstrates the potential of combining semantic priors, geometric verification and adaptive feature enhancement for dynamic indoor localization.

Keywords: visual SLAM; indoor localization; semantic segmentation; feature enhancement; object detection



Academic Editors: Songyan Niu, Yuchen Wei and Zhenghao Li

Received: 16 June 2026

Revised: 13 July 2026

Accepted: 21 July 2026

Published: 23 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article

distributed under the terms and

conditions of the [Creative Commons](#)

[Attribution \(CC BY\)](#) license.

1. Introduction

Simultaneous localization and mapping (SLAM) remain a core capability for mobile robots, autonomous navigation and augmented/virtual reality applications [1]. Visual SLAM in particular is attractive for indoor deployment due to the low cost and wide availability of RGB-D sensors. However, traditional visual SLAM systems commonly rely on the assumption of a static environment, which is often challenged in real-world indoor settings due to dynamic disturbances such as moving people and furniture. These dynamics produce mismatches, corrupt map consistency, and degrade pose estimation accuracy. In addition, many indoor scenes contain large low-texture areas (e.g., blank walls or homogeneous floors) where classical binary descriptors such as ORB suffer from deficient feature coverage [2], further limiting localization and mapping performance. Addressing these dynamic disturbances is critical for practical applications. In autonomous mobile robots operating in human-populated environments, inaccurate localization due to moving people or objects can lead to unsafe navigation decisions and collisions. Similarly, in augmented and virtual reality (AR/VR) applications, tracking failures or jitter induced by dynamic elements severely degrade the user experience and can cause motion sickness. Therefore, developing a visual SLAM system that is robust to both dynamic objects and low-texture regions is essential for the safe and reliable deployment of these technologies in real-world indoor settings.

Traditional visual SLAM systems, such as VINS [3], MonoSLAM [4], PTAM [5], and ORB-SLAM [6] series, mainly rely on local feature point matching and traditional optimization algorithms to achieve pose estimation and map construction. Among them, ORB-SLAM3 [7] integrates information from visual and inertial sensors and adopts a multi-map management strategy to achieve higher robustness and positioning accuracy, but still faces significant challenges in handling dynamic scenes. In dynamic environments, feature points on moving objects are often mistaken for being consistent with the scene background, leading to incorrect matching and interfering with global pose estimation [8].

To enhance trajectory accuracy in dynamic environments, numerous SLAM algorithms [9–16] have been proposed, including the use of geometry based robust estimation, semantic masking and instance level segmentation, tightly coupled multi-target tracking and joint optimization, dense background reconstruction, and long-term consistency strategies. All these methods aim to reduce the impact of dynamic objects and improve localization accuracy in dynamic environments. However, these SLAM systems have overlooked the handling of semi-static objects such as moving chairs.

Meanwhile, feature extraction is a crucial step in visual SLAM. Traditional manually designed features such as SIFT [17], SURF [18] and ORB, have long played a fundamental role in SLAM systems. However, their repeatability and discriminative ability have gradually become bottlenecks in low-texture scenes, environments with significant changes in lighting or angles, and dynamic or weak texture settings. With the advancement of deep learning and attention mechanisms, local feature methods based on deep learning, such as SuperPoint (self-supervised interest point detection and description) [19] and GCNv2 (Graph Convolutional Network version 2) [20], have been adopted to enhance the repeatability of keypoints and the discriminative power of descriptors. Nonetheless, in scenes with high or low texture caused by dynamic objects, there can be a significant decrease in the number of retained static feature points, which may lead to degradation in feature matching and pose instability.

To address the aforementioned issues, this paper proposes a robust SLAM framework, named SFSE-SLAM, which integrates semantic feature selection with SuperPoint feature enhancement. The objective of this work is to reduce trajectory estimation error in dynamic indoor environments by jointly tackling two intertwined problems: (1) the corruption of

feature correspondences caused by moving objects, and (2) the scarcity of static features in low-texture regions resulting from dynamic feature removal. YOLOv11-SEG [21,22] integrates advanced convolutional techniques and innovative attention mechanisms to generate pixel-level dynamic object masks to remove dynamic feature points, while retaining static background features through motion consistency verification. To overcome the lack of feature points in static regions, semantic perception is combined with semantic information to enhance and redistribute feature points. Specifically, the feature extraction is enhanced by strategically increasing sampling density in reliable static semantic areas, thereby compensating for regions affected by dynamic object removal. The core contributions of this work are summarized as follows.

(1) A semantic-geometric collaborative filtering framework that combines pixel-level semantic segmentation with geometric consistency verification. Unlike hard-mask methods such as DS-SLAM and DynaSLAM, which remove all features in detected dynamic regions, the proposed two-stage pipeline retains static points on non-moving instances through geometric verification. Unlike geometric-only methods such as SG-SLAM and DN-SLAM, which lack semantic guidance in low-texture regions, the semantic prior focuses geometric verification on candidate dynamic regions.

(2) A texture-aware feature compensation strategy that integrates semantic confidence weighting with an adaptive threshold function. Unlike learned-feature methods such as SuperPoint-SLAM and GCNv2-SLAM, which enhance feature quality but suffer feature depletion under semantic masking, the compensation mechanism adaptively adjusts extraction thresholds to recover discriminative static features.

(3) Extensive evaluations on public dynamic benchmarks (TUM RGB-D and Bonn) demonstrate that the proposed semantic-geometric filtering and feature compensation framework consistently improves localization accuracy and robustness compared with representative dynamic SLAM approaches across a wide range of indoor dynamic scenarios.

The remainder of the paper is organized as follows. Related work is discussed in Section 2. Section 3 describes the detailed methodology of the proposed SLAM framework. Section 4 presents the experimental results of the dynamic SLAM system. Section 5 concludes our work and suggests future research directions for improving computational efficiency and system robustness of the framework in dynamic environments.

2. Related Work

In recent years, researchers have proposed various strategies to mitigate interference from dynamic objects. Among these, one common approach leverages geometric and motion consistency, either in its traditional form or through improved variants, to eliminate dynamic features or local motion anomalies using techniques such as robust estimation, optical flow, or point correlation. This method effectively restores stable pose estimation in the presence of moving objects. For example, DS-SLAM [23] and DynaSLAM [24] utilize geometric and motion consistency to restore stable pose estimation. However, their effectiveness is severely limited in low-texture regions where feature correspondences are sparse.

Another approach integrates deep learning with SLAM for dynamic environments, leveraging semantic segmentation to identify and filter dynamic objects, or employing deep learning-enhanced feature extraction and matching techniques to improve the overall robustness and localization accuracy of SLAM systems in complex scenes. Xiao et al. [25] proposed a dynamic SLAM system that uses SSD detection algorithm to eliminate dynamic feature points, and proposed a missed detection compensation algorithm to improve the stability of the system. However, SSD detection only provides bounding boxes and cannot accurately segment the edges of dynamic objects, which may mistakenly remove static

feature points within the boxes. More complex frameworks like DN-SLAM [26] achieve high accuracy through multi-stage processing but struggle with near-real-time deployment due to heavy computational overhead and tracking instability when dynamic objects dominate the field of view. Although SG-SLAM [27] improves efficiency using lightweight inference, it fails to address the impact of semi-static objects such as moved furniture, which compromises long-term map consistency. While the studies mentioned above have improved the positioning accuracy of SLAM systems in dynamic environments, the impact of semi-static objects remains unaddressed.

As a classic binary descriptor, ORB has been widely used in various SLAM systems due to its fast computation speed and high matching efficiency. However, in dynamic scenes, ORB features encounter challenges such as repeated elements, occlusion, and motion blur within the region, which affect matching accuracy. In contrast, as a self-supervised learning method for keypoint detection and description, SuperPoint generates high-dimensional, dense features and exhibits good consistency in multi-view geometry problems. Although GCNv2 has significantly improved inference time, even surpassing SuperPoint in some scenarios, SuperPoint still retains advantages in expressive power and matching robustness.

Traditional SLAM methods based on YOLO or instance segmentation typically employ a "hard-mask rejection" strategy. While this effectively filters dynamic interference, it leads to a significant depletion of feature points in low-texture static backgrounds and fails to accurately identify the motion states of "semi-static objects," such as repositioned chairs. Although this work focuses on feature-based visual SLAM, recent developments in 3D Gaussian Splatting (3DGS) have shown promising results in dense reconstruction and novel-view synthesis [28,29]. 3DGS-based SLAM methods represent the map as a set of 3D Gaussians and optimize camera poses via differentiable rendering [30,31]. However, these methods typically assume a static environment and do not explicitly address dynamic objects. The semantic-geometric filtering proposed in this paper could potentially serve as a pre-processing step to remove dynamic Gaussians, which is a direction for future investigation. To address these challenges, we propose a SLAM framework that integrates semantic dynamic filtering with deep feature enhancement, thereby improving localization accuracy in dynamic environments. The objective is to refine pose estimation precision and bolster the robustness of map construction.

3. Proposed Methodology

The proposed SFSE-SLAM system is constructed based on the ORB-SLAM3 framework, and the overall workflow is illustrated in Figure 1. It primarily comprises four parallel threads:

1. In the semantic segmentation thread, a pre-trained YOLOv11-SEG model performs pixel-level segmentation of dynamic objects in the input image, yielding semantic information for all objects present in the current frame.
2. In the tracking thread, feature points from a priori known dynamic objects are first removed using semantic information from the segmentation thread. For semi-static objects, potential dynamic feature points are further eliminated by combining depth information with multi-view geometric consistency checks. Subsequently, the SuperPoint network is employed to extract feature points from the remaining static background regions, with feature point thresholds dynamically adjusted based on semantic information. Finally, the retained features are utilized for pose optimization and map construction.
3. In the local mapping thread, keyframes containing filtered and compensated features are processed. Unlike the original ORB-SLAM3 pipeline, which uses the raw ORB features for map point generation, this thread constructs map points from static

observations that have undergone dynamic filtering and texture-aware compensation. Redundant keyframes are removed to maintain a compact map representation.

4. In the loop closure detection thread, accumulated drift errors are corrected using the compensated feature descriptors. Since the descriptors are extracted from reliably static regions, the risk of false loop closures caused by dynamic feature matches is reduced. The thread completes loop detection and map merging, thereby maintaining global map consistency.

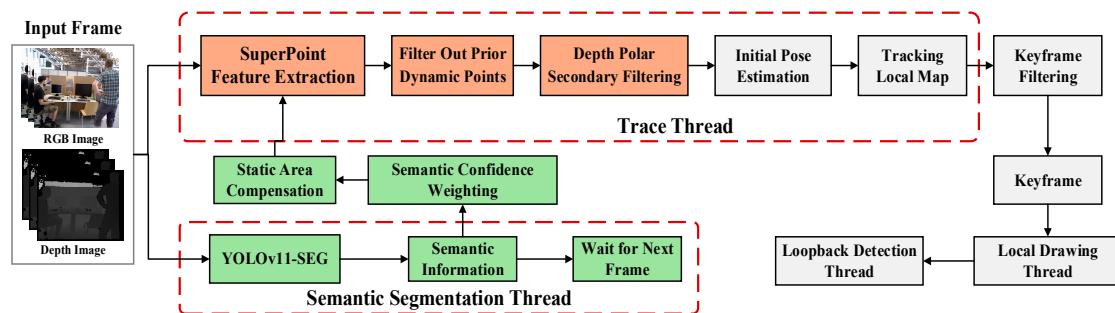


Figure 1. System framework.

3.1. Dynamic Feature Removal via Semantic and Geometric Constraints

To meet the stringent near-real-time processing requirements for handling continuous image streams at the front-end of the SLAM system, our framework prioritizes inference speed (FPS) when selecting the semantic segmentation module. In the benchmark test on the COCO [32] dataset, as shown in Table 1, although Mask R-CNN and Mask2Former achieve the highest accuracy (mean Average Precision, mAP), their excessive parameter counts (Params) and computational costs (FLOPs) result in inference latency far beyond the acceptable real-time threshold. While YOLOv11-SEG and RTMDet-Ins exhibit comparable accuracy, YOLOv11-SEG delivers significantly faster inference. Consequently, YOLOv11-SEG is selected as the final solution, offering an optimal balance between speed and efficiency at a marginal cost of precision.

Table 1. Performance comparison of different models on the COCO dataset.

Method	mAP	Params (M)	FLOPs (G)	FPS
Mask R-CNN [33]	43.7	65	423	8.6
Mask2Former [34]	44.2	63	293	7.8
RTMDet-Ins [35]	38.7	10.2	21.5	60
YOLOv11-SEG	37.8	10.1	35.5	65

Based on this selection, the system utilizes YOLOv11-SEG to identify dynamic objects. To eliminate the interference from dynamic objects in indoor scenes, the dynamic class set D includes {person, bicycle, car, dog, cat, bird}, which are the common movable object classes in the COCO dataset. For objects not in D , such as the thrown balloons and moved boxes in the Bonn dataset, the geometric branch can still detect their motion through depth variation and epipolar consistency verification across consecutive frames. This two-layer design ensures that objects outside the semantic mask are handled by geometric reasoning. The network processes each incoming RGB frame to generate pixel-level dynamic object masks. Let the semantic label assigned to each pixel by the segmentation network be denoted as $l(x, y)$, where (x, y) represents the pixel coordinates. The dynamic object mask $M(x, y)$ can be defined as:

$$M(x, y) = \begin{cases} 1, & l(x, y) \in \text{Dynamic objects} \\ 0, & \text{Others} \end{cases} \quad (1)$$

After removing the prior dynamic feature points (assigned mask value 1) based on the semantic segmentation output, semi-static objects such as relocated books or chairs persist in the scene. However, semantic segmentation networks cannot determine whether feature points within a semi-static object are dynamic feature points, so it is necessary to combine depth epipolar geometric constraints for secondary filtering of semi-static objects. Specifically, by comparing the epipolar consistency and depth variation in feature points in consecutive frames, dynamic feature points and static points can be further distinguished, thereby jointly determining the dynamic and static regions.

Since semi-static objects may exhibit both stationary and dynamic behaviors, it is essential to determine their motion states by employing epipolar geometric constraints. As illustrated in Figure 2, consider two consecutive image frames I_1 and I_2 . Let P denote a point in the three-dimensional space that undergoes displacement to P' in a dynamic scenario. The optical centers of the two cameras are represented by O_1 and O_2 . The projections of the spatial point P in the previous and current frames are denoted as P_1 and P_2 , respectively, while the projections of the displaced point P' in the corresponding frames are denoted as P'_1 and P'_2 . The rays O_1P_1 and O_2P_2 intersect at the spatial point P . The points O_1 , O_2 and P collectively define a plane known as the epipolar plane. The baseline O_1O_2 intersects the image planes I_1 and I_2 at the epipoles e_1 and e_2 , respectively. Additionally, the intersection of the epipolar plane with the two image planes results in the epipolar lines L_1 and L_2 .

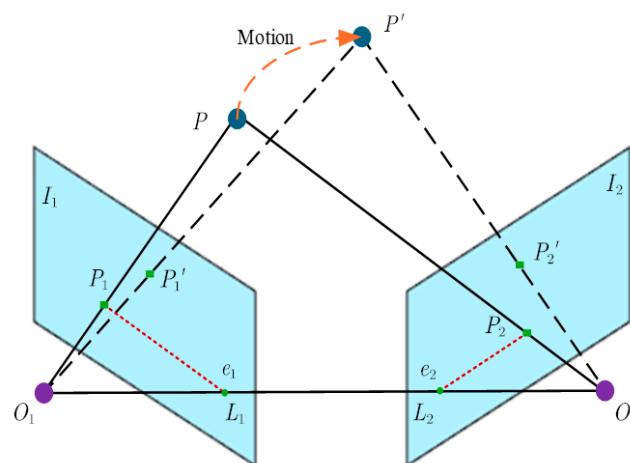


Figure 2. Schematic diagram of the epipolar geometric constraint principle.

The epipolar line L_2 in the current frame corresponding to a feature point from the previous frame is calculated using the fundamental matrix F as follows [36]:

$$L_2 = [X, Y, Z]^T = FP_1 \quad (2)$$

where $[X, Y, Z]^T$ denotes the homogeneous coordinates of the epipolar line. Since the feature point P_2 lies on the epipolar line L_2 , the epipolar constraint condition is expressed as:

$$P_2^T L_2 = P_2^T F P_1 = 0 \quad (3)$$

The perpendicular offset distance E_i from a feature point $P_i (i = 1, 2, 3, \dots)$ to its corresponding epipolar line is defined as:

$$E_i = \frac{|P_i^T F P_1|}{\sqrt{X^2 + Y^2}} \quad (4)$$

In an ideal scenario, the feature point of a static spatial point P should perfectly lie on its corresponding polar line with zero offset distance. However, in practice, various noise factors such as inaccuracies in camera calibration and errors in feature detection often introduce noise, leading to a non-zero offset distance.

Let the depth values of the matched feature points in the previous frame and the current frame be denoted as d_i and d_{i+1} , respectively. The depth variation D_i is computed as the absolute difference between these depth measurements:

$$D_i = |d_{i+1} - d_i| \quad (5)$$

As depth values are inherently scale-dependent, both the depth variation D_i and the polar line constraint offset distance E_i must be normalized to achieve scale invariance for consistent optimization. The normalized quantities are defined as:

$$D_i' = \left| \frac{d_{i+1} - d_i}{\max(d_i, d_{i+1})} \right|, E_i' = \frac{E_i}{\max(E_i)} \quad (6)$$

The weights parameters ω_D (for depth variation) and ω_E (for offset distance) are constrained by $\omega_D + \omega_E = 1$, with values adaptable to task-specific requirements. In our implementation, these weights are determined empirically through a grid search on a validation subset of the TUM dataset. For semi-static objects, where depth variations are often negligible, we set $\omega_D = 0.3$ and $\omega_E = 0.7$ giving higher weight to the epipolar constraint which is more sensitive to geometric inconsistencies. For general dynamic scenes, the dynamic score is computed using these fixed weights.

The dynamic score $S(i)$ for a feature point P_i is then computed as:

$$S(i) = \omega_D D_i' + \omega_E E_i' \quad (7)$$

A feature point is classified as dynamic if $S(i) > T$, where T is a predefined threshold; otherwise, it is deemed static. For semi-static objects exhibiting slow motion, depth variations may be negligible [24]. In such cases, since line-of-sight constraints are more sensitive to geometric inconsistencies and better capture the motion characteristics of semi-static objects, the weights should satisfy $\omega_D < \omega_E$.

It is important to note that while depth variation is included in the dynamic score computation, absolute depth changes can occur for static points due to camera motion and perspective projection. To mitigate this, the depth variation is normalized by the depth value itself, making it scale-invariant and relative rather than absolute. Furthermore, the epipolar constraint is given higher weight in the dynamic score, as it is more robust to viewpoint changes and provides a stronger geometric cue for motion detection. The depth variation term primarily serves as a supplementary indicator for detecting motion along the optical axis, where epipolar constraints may be less sensitive.

The dynamic feature point filtering process is outlined in Algorithm 1. For clarity, the variables used in Algorithm 1 are defined as follows: F denotes the fundamental matrix; $L(p)$ represents the semantic label of feature point p (dynamic or static); and T is the dynamic classification threshold. The algorithm proceeds as follows. Initially, feature points are extracted from two consecutive frames and subsequently matched. A fundamental matrix is then estimated based on these correspondences. Finally, dynamic feature points are identified and filtered out by integrating semantic, depth, and geometric cues. Through extensive experimental tuning, the threshold $T = 0.25$ was determined to be optimal, ensuring robust filtering while preserving sufficient static constraints for localization.

Algorithm 1: Dynamic Feature Point Filtering Algorithm

Input: Previous frame I_1 , current frame I_2 , semantic mask $L(p)$ dynamic score threshold T

Output: The set of static feature points in the current frame feature points S

```

1:  $P_1, P_2 = \text{SuperPoint-Keypoint}(I_1, I_2)$ 
2:  $F = \text{Estimate-Fundamental-Matrix}(P_1, P_2)$ 
3: for  $i$  in range (len ( $P_1$ ))
4:   if  $L(P_2[i]) \in \text{dynamic class}$ 
5:     discard  $P_2[i]$ 
6:   else
7:      $D_i = \text{Calculate-Depth-Variation}(P_1[i], P_2[i])$ 
8:      $E_i = \text{Calculate-EpiLine-Distance}(P_1[i], P_2[i], F)$ 
9:     if  $(\omega_D D_i + \omega_E E_i) > T$  then
10:      discard  $P_2[i]$ 
11:    else
12:      append  $P_2[i]$  to  $S$ 
13:    end if
14:  end if
15: end for
16: return  $S$ 

```

3.2. Texture-Aware Static Feature Compensation

In the dynamic indoor scene, in order to avoid the false filtering of static area features in the case of semantic segmentation error or occlusion, we introduce the semantic confidence weighting mechanism, which processes the feature points on the semi-static object by smoothing the weight, allowing them to participate in the matching when the confidence is high, so as to avoid the loss of feature points caused by hard filtering. At the same time, the feature points of the dynamic region are retained to provide more matching candidates in the low-texture scene and enhance the robustness. Therefore, the semantic confidence weighting of threshold T_f is used to control the retention threshold of SuperPoint feature points.

In YOLOv11-SEG training data, large static background elements like walls and floors typically exhibit high-frequency and regular textures, which are readily learned by the segmentation network, resulting in confidence outputs often approaching 1.0. In contrast, semi-static objects show considerable variation in form due to diverse poses and occlusions, leading to segmentation confidence levels generally clustering between 0.5 and 0.8. Dynamic targets such as pedestrians and pets demonstrate significant shape and pose variations, causing the segmentation network to exhibit lower confidence levels. The semantic segmentation network YOLOv11-SEG provides confidence scores that indicate the likelihood of each pixel belonging to a static structural category. Let the confidence degree of the pixel where the i -th SuperPoint feature point is located be $c_i \in [0, 1]$ and its semantic label be l_i .

Combined with the dynamic point score (7), the semantic confidence weight of the point is defined as:

$$w_i = \begin{cases} 1 & \text{if } l_i \in \text{Staticbackground} \\ S(i) & \text{if } l_i \in \text{Semi-static objects} \\ 0 & \text{if } l_i \in \text{Dynamic objects} \end{cases} \quad (8)$$

Then, the average semantic confidence $\lambda(c_i)$ of all feature points in the whole graph is:

$$\bar{w} = \frac{1}{N} \sum_{i=1}^N w_i c_i \quad (9)$$

where N denotes the set of all candidate feature points generated by SuperPoint. The function $\lambda(c_i)$ quantifies the semantic reliability of the static structure within the current frame. When pixels corresponding to static structures exhibit high semantic confidence, the response threshold is lowered to retain more feature points from these regions. Conversely, the threshold may be appropriately raised for dynamic targets. Previous studies have demonstrated that incorporating semantic information into feature point detection helps filter unstable points and improves matching performance [37], thereby enhancing the overall robustness of the system toward static structures. To dynamically regulate the feature extraction threshold, an offset proportional to $\bar{w}$ is incorporated into the threshold calculation.

In dynamic environment, the removal of dynamic feature points helps to improve the overall robustness, but it will lead to a sharp reduction in the number of globally effective features, resulting in insufficient feature coverage, especially in static and low-texture regions. This sparse feature will weaken the accuracy of relocation and loop detection. Therefore, we propose a static region feature compensation mechanism, which uses the statistical information of the number of features as the feedback signal, and dynamically adjusts the static region detection threshold through the global feature density. Let the set of all feature points extracted from the original image be denoted as $A = \{p_1, \dots, p_a\}$, where a represents the total number of feature points in the current frame. After applying dynamic point filtering, the remaining static features are denoted as $B = \{p'_1, p'_2, \dots, p'_b\} \subset A$, with b indicating the number of valid feature points in the static region of the current frame. The global valid feature retention ratio is defined as:

$$r = \frac{b}{a}, 0 < r \leq 1 \quad (10)$$

where r serves as an indicator for estimating the feature health state of the current frame. When $r \rightarrow 1$, it indicates that the scene contains abundant features with a high proportion of static characteristics, thus requiring no additional compensation. Conversely, when $r \rightarrow 0$, it suggests that most features have been removed due to dynamic interference, resulting in sparse features in static regions. Under such conditions, the detection threshold should be automatically lowered to enhance point cloud coverage and regional information entropy [38].

According to principles of visual information statistics, a more uniform distribution of feature points can increase matching information entropy and improve tracking robustness. Therefore, a feature compensation term is introduced, allowing the threshold to adaptively decrease as r decreases [39]. The compensation activation function $\phi(r)$ is defined to map the feature ratio to the corresponding compensation intensity as follows:

$$\phi(r) = \frac{1}{1 + \exp(-r)} \quad (11)$$

Building upon the original threshold formulation proposed in [32], this study presents an enhanced adaptive threshold function that incorporates two novel components: (1) a semantic confidence weighting mechanism, and (2) a static region feature compensation term. The proposed semantically aware threshold function is formulated as:

$$T_f^{sem} = \mu + \sqrt{\frac{\sigma^2}{2}} + \frac{\mu_1}{1 + \exp(-\mu_2 m)} - \alpha \frac{1}{N} \sum_{i=1}^N c_i w_i - \beta \cdot \phi(r) \quad (12)$$

where μ and σ are the mean and standard deviation of the confidence of the feature points of the current frame, respectively; m is the number of feature points matched with the previous frame; μ_1 and μ_2 are parameters inherited from the original adaptive threshold mechanism; α and β are weighting coefficients for the semantic confidence and feature compensation terms, respectively.

4. Experiments

To validate the performance of the proposed SFSE-SLAM algorithm, comprehensive experiments were conducted on two publicly available datasets: TUM and Bonn. The experimental results were systematically compared with those obtained from ORB-SLAM3 and several state-of-the-art dynamic SLAM algorithms known for their superior performance in dynamic environments. All experiments were carried out on a computer with Intel i5-12600k CPU, NVIDIA RTX 3050 GPU, 8 GB RAM and Ubuntu20.04 operating system.

4.1. SuperPoint Ablation Experiment

To verify the contribution of each module to the overall system performance, the following ablation combinations are designed. ORB: semantic filtering and feature enhancement; +Semantic filtering: add YOLOv11-SEG semantic dynamic point elimination; +Superpoint: replace ORB feature with superpoint; +Adaptive threshold: add semantic confidence weighting and feature compensation mechanism to the original threshold. Table 2 presents the root mean square error (RMSE) and standard deviation (S.D.) of the absolute trajectory error on the fr3-walking-xyz sequence, along with a comparison of the average number of extracted features, matched features, and the inlier ratio.

Table 2. Ablation Experiment.

Methods	Semantic Filtering	Superpoint	Adaptive Threshold	RMSE/m	S.D. /m	Features	Matches	Inlier Ratio
ORB	✗	✗	✗	0.6999	0.4176	-	-	-
+Semantic filtering	✓	✗	✗	0.0207	0.0103	990	228	0.234
+SuperPoint	✓	✓	✗	0.0168	0.0095	715	201	0.302
+Adaptive threshold	✓	✓	✓	0.0148	0.0091	724	218	0.328

It can be seen from the table that when only the YOLOv11-SEG semantic filtering module is introduced, RMSE and S.D. are 97.9% and 91.8% lower than the baseline of ORB-SLAM3. It is verified that semantic dynamic point elimination has a significant effect on SLAM positioning in high dynamic scenes. After superposing SuperPoint feature replacement, the errors are further reduced by 23.7% and 26.0%. Although the average extracted feature points is lower than that of ORB, the inlier rate of feature points is higher, indicating better feature quality and stronger robustness. At the same time, the problem of sparse features after filtering was solved. After using the semantic adaptive threshold, the positioning accuracy and feature point quality are further improved, and the three work together to make the dynamic scene SLAM reach centimeter accuracy.

4.2. Experiment on the TUM Dataset

The TUM RGB-D dataset [40] is widely used as a benchmark for evaluating SLAM systems. It provides synchronized RGB and depth images along with accurate ground-truth trajectories. In this work, we select four dynamic sequences from the fr3/walking subset of the TUM dataset to evaluate the performance of our system under different levels of scene

dynamics: Fr3-walking-static represents a low-dynamic scenario, Fr3-walking-xyz and Fr3-walking-rpy represent moderate-dynamic scenarios, and Fr3-walking-half represents a high-dynamic scenario. These sequences were used to evaluate the system's overall robustness in highly dynamic indoor environments.

Through several experiments, the threshold value of depth epipolar geometric constraint $T = 0.25$ and weight $\omega_D = 0.3$, $\omega_E = 0.7$ were set to balance the contribution of depth change and epipolar constraint. As shown in Figure 3, the suffix “-O” denotes the results of the original ORB-SLAM3 system across four TUM datasets, while “-I” refers to the results of the proposed system. In Figure 3a,c,e,g, ORB-SLAM3 extracts a large number of feature points, many of which are concentrated on the moving human subject. By contrast, Figure 3b,d,f,h present the results after applying the proposed dynamic feature filtering algorithm, where most feature points on moving objects have been effectively removed. This demonstrates the method's ability to achieve effective filtering performance in dynamic environments. As illustrated in Figure 4, the tracking results of ORB-SLAM3 and SFSE-SLAM at different time periods in the Fr3-walking-rpy sequence are compared. After removing dynamic features, the proposed algorithm effectively extracts and tracks feature points even in low-texture areas, whereas ORB-SLAM3 loses track of map points in such areas. These results indicate SFSE-SLAM exhibits strong robustness in handling low-texture static areas with sparse features.

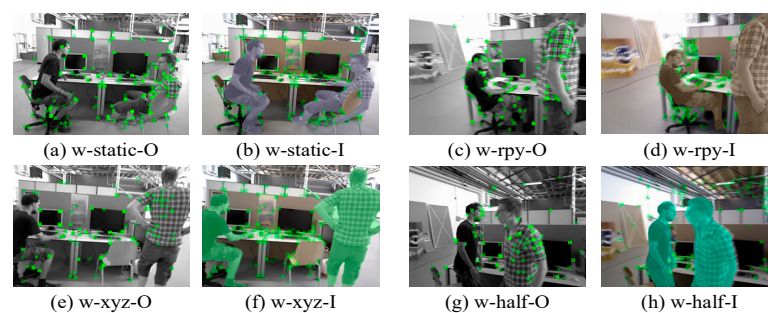


Figure 3. Dynamic feature point filtering results on high-dynamic TUM sequences: (a) w-static-O; (b) w-static-I; (c) w-rpy-O; (d) w-rpy-I; (e) w-xyz-O; (f) w-xyz-I; (g) w-half-O; (h) w-half-I.



Figure 4. Comparative tracking performance on the Fr3-walking-rpy sequence: (a) ORB-SLAM3; (b) SFSE-SLAM.

In order to evaluate the tracking results, experiments were conducted in four dynamic scenes of the tum dataset FR3-walking, using absolute trajectory error (ATE) and relative attitude error (RPE) as error measures. The improved system is compared with the ORB-SLAM3 system. Figures 5 and 6 show the ATE and RPE experimental results of four sequences of ORB-SLAM3 and the algorithm in this paper under the input of RGB-D

camera in the two systems, respectively. In the walking dataset, ORB-SLAM3 algorithm is inaccurate in positioning due to the existence of dynamic objects, and even has wrong trajectory in some areas. The camera motion trajectory obtained by this algorithm has higher coincidence with the real trajectory, which can better deal with high dynamic scenes.

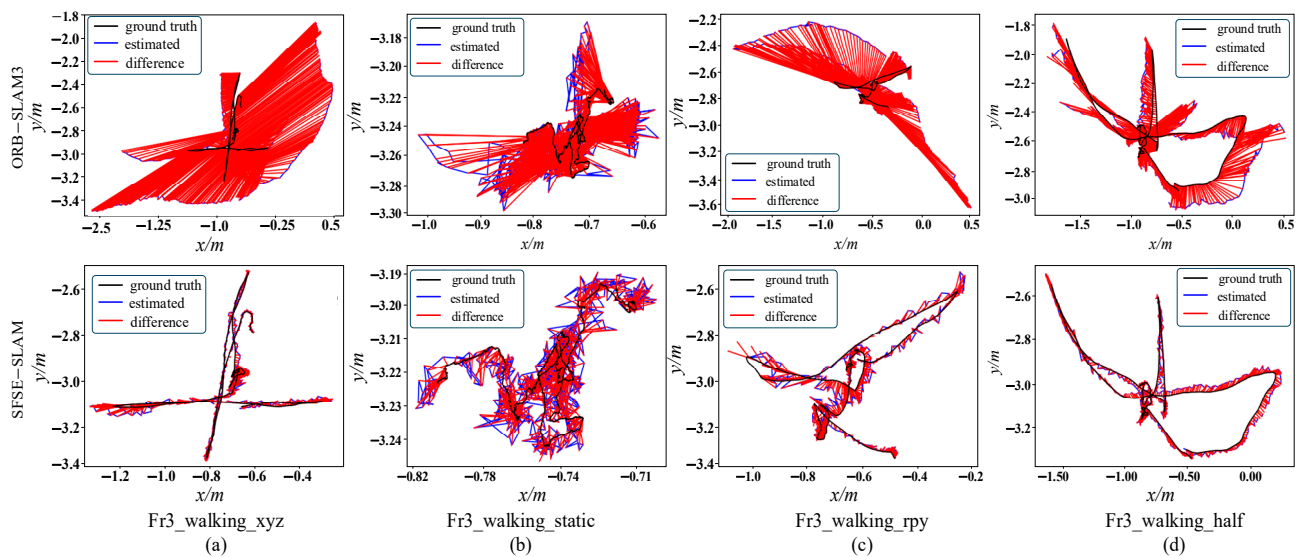


Figure 5. Schematic diagram of the ATE between ORB-SLAM3 (the first row) and the SFSE-SLAM (the second row) in high dynamic scenarios: (a) Fr3_walking_xyz; (b) Fr3_walking_static; (c) Fr3_walking_rpy; (d) Fr3_walking_half.

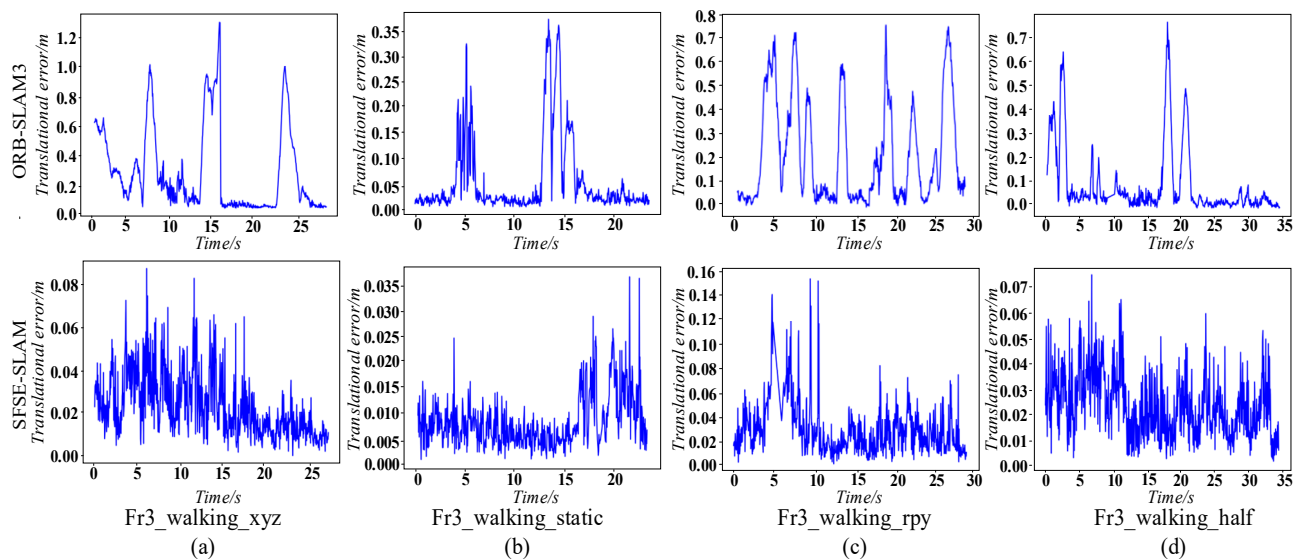


Figure 6. Schematic diagram of the RPE between ORB-SLAM3 (the first row) and the SFSE-SLAM (the second row) in high dynamic scenarios: (a) Fr3_walking_xyz; (b) Fr3_walking_static; (c) Fr3_walking_rpy; (d) Fr3_walking_half.

Tables 3 and 4 show the comparison of absolute trajectory error and relative trajectory error between the proposed system and ORB-SLAM3. Compared with ORB-SLAM3, the proposed algorithm can significantly reduce ATE and RPE of each sequence. In fr3-walking-xyz, the RMSE and S.D. of the absolute trajectory error of high dynamic sequences decreased by 97.9% and 97.8%, respectively. The effectiveness of the geometric verification stage in handling semi-static objects is further examined on the fr3-walking-static sequence. In segments of this sequence where a person remains stationary, the two-stage pipeline retains features on the stationary person that would be removed by hard-mask methods

such as DS-SLAM or DynaSLAM. This is because the geometric verification stage evaluates per-point motion consistency rather than relying solely on the semantic mask. In the sitting-static low dynamic sequence, the RMSE value and standard deviation of absolute trajectory error decreased by 46.1% and 43.1%, and the RMSE value and standard deviation of relative pose error decreased by 9.4% and 11.5%. Therefore, the proposed algorithm has higher positioning accuracy in high dynamic environment.

Table 3. Comparison of absolute trajectory error [m] between ORB-SLAM3 and improved algorithm.

Sequences	ORB-SLAM3				OURS			
	RMSE	Mean	Median	S.D.	RMSE	Mean	Median	S.D.
Fr3-walking-xyz	0.6999	0.5617	0.4111	0.4176	0.0148	0.0117	0.0087	0.0091
Fr3-walking-static	0.3594	0.3188	0.2941	0.1660	0.0052	0.0046	0.0040	0.0025
Fr3-walking-rpy	0.5862	0.5349	0.5184	0.2398	0.0285	0.0227	0.0172	0.0172
Fr3-walking-half	0.3316	0.3077	0.2945	0.1235	0.0188	0.0168	0.0153	0.0084

Table 4. Comparison of relative trajectory error [m] between ORB-SLAM3 and improved algorithm.

Sequences	ORB-SLAM3				OURS			
	RMSE	Mean	Median	S.D.	RMSE	Mean	Median	S.D.
Fr3-walking-xyz	0.0261	0.0213	0.0172	0.0150	0.0130	0.0105	0.0083	0.0077
Fr3-walking-static	0.0174	0.0123	0.0081	0.0123	0.0059	0.0050	0.0043	0.0031
Fr3-walking-rpy	0.0280	0.0216	0.0176	0.0177	0.0183	0.0137	0.0106	0.0121
Fr3-walking-half	0.0228	0.0162	0.0120	0.0159	0.0127	0.0107	0.0091	0.0068

In order to better reflect the performance of this algorithm, we compare it with the advanced visual slam methods SG-SLAM, DN-SLAM and FD-SLAM [41], and use the experimental results in the original paper. For the TUM dataset comparisons with SG-SLAM, DN-SLAM, and FD-SLAM in Tables 5 and 6, the results for these baselines are taken from their respective original publications. Re-implementation of each method on identical hardware was not feasible within the current revision cycle. For the Bonn dataset comparisons in Table 7, SG-SLAM and DN-SLAM were evaluated on the same hardware platform (Intel i5-12600K, NVIDIA RTX 3050) as the proposed method. Evaluation of additional recent baselines on identical hardware is a direction for future work. Tables 5 and 6, respectively, show the comparison of RMSE and S.D. values of the four algorithms in absolute and relative trajectory errors. The results show that for high dynamic sequences, these methods show better adaptability and improvement effect than ORB-SLAM3 algorithm. The proposed algorithm achieves optimal performance on both the walking-half sequence and the walking-static sequence, exhibiting superior reliability in dynamic environments.

Table 5. Comparison of RMSE [m] values of different algorithms.

Sequences	ATE				RPE			
	SG-SLAM	DN-SLAM	FD-SLAM	OURS	SG-SLAM	DN-SLAM	FD-SLAM	OURS
Fr3-w-xyz	0.0152	0.015	0.0166	0.0148	0.0194	0.024	0.0221	0.0130
Fr3-w-static	0.0073	0.008	0.0060	0.0052	0.0100	0.011	0.0116	0.0059
Fr3-w-rpy	0.0324	0.032	0.0412	0.0285	0.0450	0.065	0.0132	0.0183
Fr3-w-half	0.0268	0.026	0.0323	0.0188	0.0279	0.035	0.0235	0.0127

Table 6. Comparison of S.D. [m] values of different algorithms.

Sequences	ATE				RPE			
	SG-SLAM	DN-SLAM	FD-SLAM	OURS	SG-SLAM	DN-SLAM	FD-SLAM	OURS
Fr3-w-xyz	0.0075	0.007	0.0087	0.0091	0.0100	0.008	0.0073	0.0077
Fr3-w-static	0.0034	0.003	0.0033	0.0025	0.0051	0.003	0.0046	0.0027
Fr3-w-rpy	0.0187	0.019	0.0301	0.0172	0.0262	0.032	0.0053	0.0121
Fr3-w-half	0.0134	0.013	0.0158	0.0084	0.0146	0.013	0.0097	0.0068

Table 7. Comparison of absolute trace error [m] of different algorithms on Bonn dataset.

Sequences	ORB-SLAM3		SG-SLAM		DN-SLAM		Ours	
	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.
Crowd	0.980	0.601	0.0234	0.0143	0.025	0.016	0.0181	0.0121
Crowd2	0.628	0.357	0.0584	0.0420	0.028	0.017	0.0631	0.0315
Crowd3	0.367	0.216	0.0319	0.0219	0.026	0.014	0.0182	0.0074
Bonn-balloon	0.158	0.063	-	-	0.030	0.012	0.0231	0.0118
Bonn-balloon2	0.126	0.054	-	-	0.025	0.012	0.0234	0.0090
Person-tracking	0.698	0.327	0.0400	0.0139	0.038	0.015	0.0214	0.0105
Person-tracking2	0.786	0.454	0.0376	0.0154	0.042	0.017	0.0279	0.035

4.3. Experiment on the Bonn Dataset

The Bonn dynamic dataset [42] contains 24 sequences featuring various dynamic activities such as crowd walking, balloon throwing, and box moving. In this study, seven representative sequences were selected for evaluation. The “balloon” sequence records the instant when a balloon is thrown, while the “crowd” sequence involves three individuals walking randomly in an indoor environment. The pedestrian tracking sequence depicts the camera following a person moving at a slow pace. These seven sequences encompass highly dynamic environments that present significantly greater challenges than those in the TUM dataset, characterized by intricate motion patterns that substantially degrade the accuracy and robustness of SLAM systems.

Table 7 shows the experimental results on the Bonn dynamic dataset. Two representative dynamic SLAM algorithms, SG-SLAM and DN-SLAM, were evaluated on the same hardware platform, and their performance was compared using ATE and S.D. The results show that even in complex dynamic environments, the proposed method effectively suppresses dynamic interference and maintains robust localization accuracy. In highly dynamic scenes, such as frames from the “crowd” sequence where moving objects occupy a large portion of the view, the algorithm can still adapt and achieve stable pose estimation. Furthermore, the ‘person-tracking’ sequences in the Bonn dataset serve as a valuable testbed for evaluating the system’s performance on semi-static objects. In these sequences, the moving person occasionally stops and interacts with static objects, such as moving a chair. Our proposed method, through its geometric consistency verification (Algorithm 1), successfully classifies the temporarily moved chair as a dynamic entity and filters out its feature points, while preserving the static background features. As shown in Table 7, our method achieves significantly higher accuracy (RMSE of 0.0214 m and 0.0279 m) on these sequences compared to SG-SLAM (0.0400 m and 0.0376 m), demonstrating the effectiveness of our semantic-geometric filtering strategy in handling such semi-static disturbances.

The proposed method achieves competitive or superior performance on most sequences, though limitations are observed on densely occupied scenes such as the Bonn Crowd2 sequence. On the Crowd2 sequence of the Bonn dataset, the proposed method achieves an ATE RMSE of 0.0631 m, which is higher than DN-SLAM (0.028 m) and SG-SLAM (0.0584 m). In this sequence, three individuals move randomly and occupy a large proportion of the visual field. Under such dense dynamic occupancy, the semantic-geometric filtering removes a substantial number of feature points, and the compensation mechanism is constrained by the remaining static region area. By contrast, DN-SLAM applies a per-point depth-consistency check that can retain individual static points within dynamic-class regions. This result illustrates a known limitation of region-level filtering approaches in densely occupied scenes, which is acknowledged in Section 5.

5. Conclusions

This paper presents SFSE-SLAM, a novel semantic SLAM framework designed for dynamic indoor environments, which combines semantic dynamic feature filtering with deep

feature enhancement to achieve high-precision localization and robust feature matching. The proposed method addresses two critical challenges in dynamic SLAM: (1) unreliable feature points caused by moving objects, and (2) insufficient feature density in low-texture regions. By jointly filtering dynamic feature points using YOLOv11-SEG generated dynamic masks and deep geometric consistency constraints, and incorporating SuperPoint feature extraction augmented with a semantic confidence weighting mechanism, the method effectively enhances both the quality and density of features in low-texture regions. Experimental results on multiple public dynamic datasets validate the effectiveness of the proposed system, demonstrating superior performance in high-motion scenarios.

Notably, the current implementation relies heavily on deep network inference, which incurs substantial computational overhead and remains susceptible to performance degradation caused by semantic segmentation errors, such as missed detection and inaccurate object boundaries. In addition, the following limitations are acknowledged. First, the system requires two deep network inferences per frame (YOLOv11-SEG and SuperPoint), which results in near-real-time rather than real-time performance at standard video frame rates. Second, on sequences where dynamic objects dominate the visual field, such as the Bonn Crowd2 sequence, the region-level filtering may over-reduce the feature pool, and the compensation mechanism is bounded by the available static region area. Third, a dedicated controlled ablation study specifically evaluating semi-static object handling has not been conducted. In addition, although the proposed method demonstrates promising results on public benchmark datasets, its transferability to real-world robotic platforms operating under diverse environmental conditions and hardware constraints requires further investigation. Future research directions include the development of lightweight semantic segmentation architectures, the exploration of end-to-end multimodal feature fusion techniques, extensive validation on real robotic systems, dedicated ablation studies on semi-static objects, and evaluation of additional recent baselines on identical hardware, to enhance computational efficiency, generalization capability, and overall system robustness.

Author Contributions: Conceptualization, Y.W.; methodology, Y.W.; software, Y.W.; validation, Y.W., C.P. and Z.L.; formal analysis, Y.W. and C.P.; investigation, Y.W.; resources, C.P. and S.X.Y.; writing—original draft preparation, Y.W.; writing—review and editing, C.P., C.C. and S.X.Y.; visualization, Y.W.; supervision, C.P., C.C. and S.X.Y.; project administration, C.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China (Grant No. 62173138 and Grant No. 62533010), the Hunan Provincial Natural Science Foundation of China (Grant No. 2026JJ50236), and the Hainan Provincial Natural Science Foundation of China (Grant No. 526MS0239).

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Short, D.; Lei, T.; Luo, C.; Carruth, D.W.; Bi, Z. A bio-inspired algorithm in image-based path planning and localization using visual features and maps. *Intell. Robot.* **2023**, *3*, 222–241. [\[CrossRef\]](#)
2. Rublee, E.; Rabaud, V.; Konolige, K. ORB: An efficient alternative to SIFT or SURF. In *Proceedings of the International Conference on Computer Vision*; IEEE: New York, NY, USA, 2011; pp. 2564–2571. [\[CrossRef\]](#)
3. Qin, T.; Li, P.; Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Robot.* **2018**, *34*, 1004–1020. [\[CrossRef\]](#)
4. Davison, A.J.; Reid, I.D.; Molton, N.D. MonoSLAM: Real-time single camera SLAM. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 1052–1067. [\[CrossRef\]](#) [\[PubMed\]](#)

5. Klein, G.; Murray, D. Parallel tracking and mapping on a camera phone. In Proceedings of the IEEE International Symposium on Mixed and Augmented Reality, Orlando, FL, USA, 19–22 October 2009; pp. 83–86. [\[CrossRef\]](#)
6. Mur-Artal, R.; Montiel, J.M.M.; Tardos, J.D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans. Robot.* **2015**, *31*, 1147–1163. [\[CrossRef\]](#)
7. Campos, C.; Elvira, R.; Rodríguez, J.J.G. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam. *IEEE Trans. Robot.* **2021**, *37*, 1874–1890. [\[CrossRef\]](#)
8. Dang, T.; Nguyen, K.; Huber, M. V3D-SLAM: Robust RGB-D SLAM in Dynamic Environments with 3D Semantic Geometry Voting. In Proceedings of the IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Detroit, MI, USA, 1–5 October 2024; pp. 7847–7853. [\[CrossRef\]](#)
9. Runz, M.; Buffier, M.; Agapito, L. Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects. In Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR), Munich, Germany, 16–20 October 2018; pp. 10–20. [\[CrossRef\]](#)
10. Zhu, H.; Bai, S.; Shi, J.; Lu, J.; Zuo, X.; Huang, S.; Yao, X. Ellipsoid-SLAM: Enhancing dynamic scene understanding through ellipsoidal object representation and trajectory tracking. *Vis. Comput.* **2025**, *41*, 7493–7508. [\[CrossRef\]](#)
11. Dai, W.; Zhang, Y.; Li, P.; Fang, Z.; Scherer, S. RGB-D SLAM in dynamic environments using point correlations. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 373–389. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Jiao, J.; Wang, C.; Li, N.; Deng, Z.; Xu, W. An adaptive visual dynamic-SLAM method based on fusing the semantic information. *IEEE Sens. J.* **2022**, *22*, 17414–17420. [\[CrossRef\]](#)
13. Zhong, Y.; Hu, S.; Huang, G.; Bai, L.; Li, Q. WF-SLAM: A robust VSLAM for dynamic scenarios via weighted features. *IEEE Sens. J.* **2022**, *22*, 10818–10827. [\[CrossRef\]](#)
14. Zhang, X.; Zhang, R.; Wang, X. Visual SLAM mapping based on YOLOv5 in dynamic scenes. *Appl. Sci.* **2022**, *12*, 11548. [\[CrossRef\]](#)
15. Guan, H.; Qian, C.; Wu, T.; Hu, X.; Duan, F.; Ye, X. A dynamic scene vision SLAM method incorporating object detection and object characterization. *Sustainability* **2023**, *15*, 3048. [\[CrossRef\]](#)
16. Zheng, Z.; Lin, S.; Yang, C. RLD-SLAM: A robust lightweight VI-SLAM for dynamic environments leveraging semantics and motion information. *IEEE Trans. Ind. Electron.* **2024**, *71*, 14328–14338. [\[CrossRef\]](#)
17. He, Y.; Deng, G.; Wang, Y. Optimization of SIFT algorithm for fast-image feature extraction in line-scanning ophthalmoscope. *Optik* **2018**, *152*, 21–28. [\[CrossRef\]](#)
18. Bay, H.; Tuytelaars, T.; Van Gool, L. Surf: Speeded up robust features. In *Proceedings of the European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 404–417. [\[CrossRef\]](#)
19. DeTone, D.; Malisiewicz, T.; Rabinovich, A. Superpoint: Self-supervised interest point detection and description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018; pp. 224–236. [\[CrossRef\]](#)
20. Tang, J.; Ericson, L.; Folkesson, J. GCNv2: Efficient correspondence prediction for real-time SLAM. *IEEE Robot. Autom. Lett.* **2019**, *4*, 3505–3512. [\[CrossRef\]](#)
21. Khanam, R.; Hussain, M. Yolov11: An overview of the key architectural enhancements. *arXiv* **2024**, arXiv:2410.17725. [\[CrossRef\]](#)
22. Zhu, H.; Bai, S.; Shi, J.; Wang, C.; Sun, Y.; Lu, J.; Shu, X.; Huang, S. IOFusion: Instance segmentation and optical-flow guided 3D reconstruction in dynamic scenes. *Vis. Comput.* **2025**, *41*, 799–813. [\[CrossRef\]](#)
23. Yu, C.; Liu, Z.; Liu, X.J. DS-SLAM: A semantic visual SLAM towards dynamic environments. In Proceedings of the IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 1–5 October 2018; pp. 1168–1174. [\[CrossRef\]](#)
24. Bescos, B.; Fàcil, J.M.; Civera, J. DynaSLAM: Tracking, mapping, and inpainting in dynamic scenes. *IEEE Robot. Autom. Lett.* **2018**, *3*, 4076–4083. [\[CrossRef\]](#)
25. Xiao, L.; Wang, J.; Qiu, X. Dynamic-SLAM: Semantic monocular visual localization and mapping based on deep learning in dynamic environment. *Robot. Auton. Syst.* **2019**, *117*, 103237. [\[CrossRef\]](#)
26. Ruan, C.; Zang, Q.; Zhang, K. Dn-slam: A visual slam with orb features and nerf mapping in dynamic environments. *IEEE Sens. J.* **2023**, *24*, 5279–5287. [\[CrossRef\]](#)
27. Cheng, S.H.; Sun, C.G.; Zhang, S.J. SG-SLAM: A Real-Time RGB-D Visual SLAM Toward Dynamic Scenes with Semantic and Geometric Information. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 7501012. [\[CrossRef\]](#)
28. He, S.; Ji, P.; Yang, Y.; Wang, C.; Ji, J.; Wang, Y.; Ding, H. A Survey on 3D Gaussian Splatting Applications: Segmentation, Editing, and Generation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (Early Access)*; IEEE: New York, NY, USA, 2026. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Wang, C.; He, S.; Fang, X.; Hu, Z.; Huang, J.-H.; Shen, Y.; Tiwari, P. Reasoning Beyond Points: A Visual Introspective Approach for Few-Shot 3D Segmentation. In Proceedings of the Advances in Neural Information Processing Systems 38 (NeurIPS 2025), San Diego, CA, USA, 2–7 December 2025; pp. 117394–117414.
30. Yang, Y.; Liu, X.; Wang, Y.; Wang, J.; Dou, H.; Wang, C. DiffStyle3D: Consistent 3D Gaussian Stylization via Attention Optimization. *arXiv* **2026**, arXiv:2601.19717. [\[CrossRef\]](#)

31. Wang, C.; He, S.; Fang, X.; Han, J.; Liu, Z.; Ning, X.; Li, W.; Tiwari, P. Point Clouds Meets Physics: Dynamic Acoustic Field Fitting Network for Point Cloud Understanding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 10–17 June 2025; p. 22182. [\[CrossRef\]](#)
32. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*; Springer International Publishing: Cham, Switzerland, 2014; pp. 740–755. [\[CrossRef\]](#)
33. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988. [\[CrossRef\]](#)
34. Cheng, B.; Misra, I.; Schwing, A.G.; Kirillov, A.; Girdhar, R. Masked-attention mask transformer for universal image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1290–1299. [\[CrossRef\]](#)
35. Lyu, C.; Zhang, W.; Huang, H.; Zhou, Y.; Wang, Y.; Liu, Y.; Zhang, S.; Chen, K. RtmDET: An empirical study of designing real-time object detectors. *arXiv* **2022**, arXiv:2212.07784. [\[CrossRef\]](#)
36. Ranftl, R.; Koltun, V. Deep fundamental matrix estimation. In *Proceedings of the Computer Vision, ECCV 2018: 15th European Conference, Munich, Germany, 8–14 September 2018; Part I*; Springer: Cham, Switzerland, 2018; pp. 292–309. [\[CrossRef\]](#)
37. Wu, J.; Xu, R.; Wood-Doughty, Z. Segment anything model is a good teacher for local feature learning. In *IEEE Transactions on Image Processing*; IEEE: New York, NY, USA, 2025. [\[CrossRef\]](#) [\[PubMed\]](#)
38. He, Y.; Yu, J.; He, X. Dynamic SLAM algorithm adopt with eliminating mismatched point chains in grid motion statistics. *Sci. Rep.* **2025**, *15*, 26597. [\[CrossRef\]](#) [\[PubMed\]](#)
39. Chu, G.; Peng, Y.; Luo, X. ALGD-ORB: An improved image feature extraction algorithm with adaptive threshold and local gray difference. *PLoS ONE* **2023**, *18*, e0293111. [\[CrossRef\]](#) [\[PubMed\]](#)
40. Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; pp. 573–580. [\[CrossRef\]](#)
41. Luo, Y.; Rao, Z.; Wu, R. FD-SLAM: A semantic SLAM based on enhanced fast-SCNN dynamic region detection and DeepFillv2-Driven background inpainting. *IEEE Access* **2023**, *11*, 110615–110626. [\[CrossRef\]](#)
42. Palazzolo, E.; Behley, J.; Lottes, P.; Giguère, P.; Stachniss, C. ReFusion: 3D reconstruction in dynamic environments for RGB-D cameras exploiting residuals. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 7855–7862. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.