Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

A Deep Learning-Based Visual SLAM Approach for Indoor Dynamic Scenes

jiarui qin
Liaoning University of Technology
yugang wang
paperwang-doc@163.com

Liaoning University of Technology
xueli cong
Liaoning University of Technology
liyao zhou
Liaoning University of Technology

Research Article

Keywords: ORB-SLAM3, line feature extraction, YOLOv8, epipolar geometric constraint, P3P pose estimation

Posted Date: March 30th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8981127/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

A Deep Learning-Based Visual SLAM Approach for Indoor Dynamic Scenes

All data in this article are available.

Abstract: In indoor dynamic scenes, traditional visual SLAM algorithms often suffer from a significant decline in localization accuracy due to interference from dynamic objects. To address this issue, this study proposes a visual SLAM method based on deep learning. Built upon the ORB-SLAM3 framework, the proposed method effectively enhances the system's capability in complex dynamic environments by constructing a dual dynamic feature suppression mechanism that integrates object detection and epipolar geometric constraints. Specifically, the YOLOv8m network is first lightweighted by introducing GhostConv modules to replace the original convolutional layers, reducing computational overhead. Additionally, the CBAM attention mechanism is incorporated into the backbone network to improve the accuracy of dynamic object detection. To further enhance feature stability in complex scenes, the algorithm introduces LSD line features for joint extraction alongside point features. Leveraging the robustness of line features in low-texture or dynamic edge regions, it initially filters out obvious dynamic features. Subsequently, epipolar geometric constraints are applied for a secondary screening of residual dynamic points, achieving high-precision retention of static features. Finally, pose-assisted localization is accomplished using the P3P algorithm based on the filtered static features. Experimental results on the TUM RGB-D dataset demonstrate that the improved algorithm improves localization accuracy by an average of 53.10% compared to ORB-SLAM3, with a maximum improvement of up to 93.56%. On the EuRoC-MAV dataset, the improved algorithm reduces the average ATE-RMSE by 85.32%, with improvements exceeding 85.5% on highly challenging sequences, and maintains an ATE-StD below 0.035m under complex operating conditions, enhancing localization accuracy and anti-interference capability. Physical validation further confirms the superior accuracy and robustness of the proposed method, offering an effective approach for research on high-precision SLAM systems in dynamic environments.

Key words: ORB-SLAM3; line feature extraction; YOLOv8; epipolar geometric constraint; P3P pose estimation

0 Introduction

Simultaneous Localization and Mapping (SLAM) technology acquires

environmental information through sensors (e.g., LiDAR, cameras, etc.) to determine its position and orientation in real-time while constructing a map of the environment, demonstrating significant value in fields such as autonomous driving, biomedicine, and unmanned aerial vehicles [1]. Particularly in applications like indoor mobile robot navigation and augmented reality, visual SLAM has become a crucial technology [2]. However, indoor environments often exhibit complex dynamic characteristics, such as pedestrian movement and object relocation, which can severely interfere with the localization accuracy and mapping reliability of traditional visual SLAM. Therefore, enhancing the robustness and precision of SLAM systems in indoor dynamic scenes holds important practical significance.

Existing visual SLAM systems can be categorized into direct methods and feature-based methods based on the processing principle of the front-end visual odometry [3-4]. Compared with direct methods, feature-based methods have become the mainstream approach in visual SLAM due to their advantages such as stability, computational efficiency, maturity, and suitability for loop closure detection, enabling them to meet the requirements of SLAM in various scenarios. Among these, ORB-SLAM3, as the latest version of this approach, constructs a highly integrated visual-inertial SLAM system capable of achieving more accurate and stable localization and mapping in static environments [5]. However, when the environment transitions to dynamic scenes, such as the frequent appearance of moving objects like pedestrians, animals, or vehicles, the rapid motion of dynamic objects can lead to feature point matching errors, thereby interfering with the visual SLAM system and limiting the widespread application of visual SLAM algorithms in practice [6-8].

To address the challenges of insufficient localization accuracy and robustness of SLAM in dynamic scenes, researchers have increasingly introduced deep learning methods. Leveraging its advantages in automatic feature extraction, dynamic object detection, and removal, deep learning has significantly enhanced the localization accuracy and robustness of visual SLAM, establishing itself as a mainstream technological trend in this field [9-10]. However, existing methods still exhibit certain limitations. For instance, Chang et al. [11] proposed a dynamic feature point elimination strategy based on the Yolov4-Tiny network, but this method primarily relies on point features, neglecting the rich line features present in the environment, which limits its robustness in low-texture or motion-blurred scenes. Cong et al. [12] introduced YOLOv5 into ORB-SLAM3 and integrated depth information to improve SLAM performance in

dynamic environments; however, its dynamic removal mechanism is relatively simplistic and lacks geometric verification for residual dynamic features. Liu et al. [13] employed flood fill and edge enhancement techniques to improve the YOLOv7 algorithm for separating dynamic feature points, enhancing accuracy and efficiency, but it may still suffer from false removal or missed detection in complex dynamic scenes. Dyna-SLAM [14] effectively removes dynamic objects by combining deep learning with multi-view geometry techniques; however, this method incurs high computational costs and predominantly relies on point features, failing to fully utilize the stability of line features. DS-SLAM [15] integrates semantic segmentation networks with motion consistency checks to mitigate the impact of dynamic targets, yet its capability to detect fast-moving objects requires further improvement.

In recent years, significant progress has been made in the integration of deep learning and SLAM. Regarding the combination of YOLO and SLAM, the CD-SLAM system proposed by Wen et al. demonstrates the effectiveness of fusing semantic and geometric information. Built upon the ORB-SLAM3 framework, this system is a real-time stereo visual-inertial SLAM specifically designed for complex dynamic environments. CD-SLAM enhances the tracking thread and introduces a new parallel thread, utilizing YOLOv5 to detect objects in each frame and extract semantic information. This semantic information is combined with prior information from the inertial measurement unit (IMU) for pose estimation, effectively eliminating pose information associated with dynamic objects, thereby improving localization accuracy and robustness. Furthermore, CD-SLAM employs scene flow to calculate distances and spatial velocities between consecutive frames, compensates for potentially static information through a velocity filtering algorithm, and integrates the IMU for motion prediction and consistency checking in low-texture environments. This research provides an important methodological reference for the dual dynamic feature removal mechanism proposed in this paper, which integrates YOLOv8m with geometric constraints [16].

The aforementioned studies reveal two major shortcomings in current approaches: First, most methods rely on traditional feature point-based extraction and matching, failing to fully exploit the line segment features present in dynamic scenes. Line features offer greater continuity and stability, enabling them to effectively handle complex environments such as motion blur and occlusion. Second, the dynamic feature removal mechanism is often simplistic and lacks multi-level verification, making it

difficult to thoroughly eliminate dynamic features in complex dynamic scenes, thereby compromising localization accuracy.

To address these issues, this paper proposes a dynamic visual SLAM algorithm integrating deep learning. Built upon the ORB-SLAM3 framework, the proposed method incorporates the following four key improvements:

(1) Lightweight Improved YOLOv8m Object Detection Network: The GhostConv module is introduced into the YOLOv8m backbone network to replace the original convolutional layers, effectively reducing computational overhead. Additionally, the CBAM attention mechanism is incorporated to enhance the network's perception of dynamic targets and improve detection accuracy.

(2) Point-Line Joint Feature Extraction Strategy: Building upon traditional ORB point features, LSD line features are introduced for joint extraction. Leveraging the stability of line features in low-texture regions and along dynamic edges enhances the system's feature robustness in complex environments, while also enabling preliminary removal of obvious dynamic features.

(3) Dual Dynamic Feature Suppression Mechanism Combining Semantics and Geometry: First, feature points and lines within dynamic object regions are removed based on the detection results of the improved YOLOv8m. Subsequently, epipolar geometric constraints are applied to the remaining features after preliminary screening. By calculating the distance from feature points to epipolar lines, residual dynamic features are further identified and removed, achieving thorough filtering of dynamic features.

(4) Pose Optimization Based on the P3P Algorithm: Using the filtered static feature points, the P3P algorithm is employed for auxiliary pose estimation. An nonlinear optimization objective function is constructed to iteratively solve for the optimal camera pose, enhancing localization accuracy and stability.

Through these improvements, the proposed method aims to address the limitations of existing research regarding the utilization of line features and the completeness of dynamic feature removal, thereby improving the localization accuracy and robustness of SLAM in indoor dynamic scenes.

1 Method Framework

his paper optimizes the ORB-SLAM3 framework for applications in dynamic scenes, with a primary focus on improving the tracking thread. Image frames captured by the RGB-D camera are transmitted to both the tracking thread and the dynamic

object detection module. The tracking thread is mainly responsible for point and line feature extraction, as well as dynamic feature removal. The ORB algorithm is employed to extract point features, while the LSD algorithm is utilized to extract line features from the image. Simultaneously, object detection is performed on the images based on the improved YOLOv8m algorithm. The detection results are classified into dynamic and static targets, and dynamic features are initially removed. Subsequently, the epipolar geometric constraint algorithm is applied. By calculating the distance from each feature point to its corresponding epipolar line, feature points with distances exceeding a preset threshold are identified as residual dynamic points and are further eliminated. Based on an initial pose estimation, the P3P algorithm is used to further optimize the camera pose using the filtered static features. The optimized results are then passed to the mapping thread to achieve stable mapping in dynamic environments. The structural framework of the proposed method is illustrated in Figure 1.

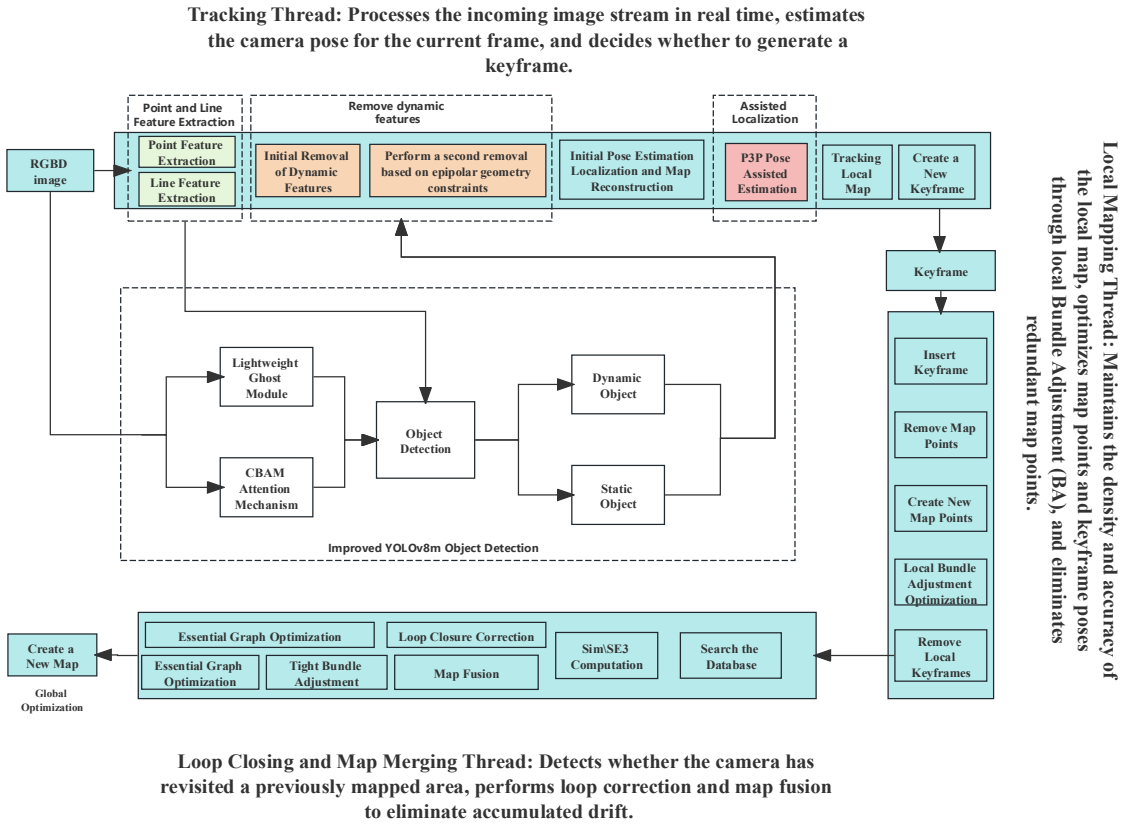


Figure 1 Structural framework of the proposed method

2 Line Feature Extraction Based on LSD Algorithm

In indoor dynamic environments, point features are susceptible to noise,

illumination changes, and viewpoint variations, leading to matching errors. In contrast, line features exhibit stable anti-interference characteristics and demonstrate stronger environmental adaptability. This paper adopts a point-line feature fusion strategy: First, ORB point features and LSD line segment features are extracted separately. Subsequently, geometric constraints are established using line segment endpoints and neighboring points. Dynamic points exhibiting poor geometric consistency with the line features are eliminated based on the local structural information of the line segments, thereby achieving dynamic feature screening during the matching stage. Concurrently, a joint point-line error function is established during the optimization process. The reprojection errors of point features and line features (based on the distance between endpoints and the line) are incorporated into a unified nonlinear optimization framework. This enables the system to fully leverage the geometric structural information of line features during pose estimation. This fusion method not only leverages the advantage of the wide distribution of point features but also enhances robustness against noise, illumination, and viewpoint variations through the stability of line features. Consequently, it facilitates a more accurate selection of features suitable for map construction, effectively improving localization accuracy [17-19]. Real-time performance analysis, as detailed in the ablation study in Section 5.1, indicates that this point-line joint extraction strategy, while effectively improving localization accuracy, achieves an overall system frame rate of 26.7 FPS through lightweight network design and algorithm optimization, satisfying the real-time requirements of SLAM systems in indoor dynamic scenes.

The LSD algorithm first utilizes the Sobel operator to calculate the gradient magnitude and direction for each pixel. Subsequently, the continuous directions are quantized into discrete angles to simplify the analysis process. Next, based on the gradient magnitude, a voting process is conducted within the quantized directions to locate potential lines. Finally, the region growing method is employed to merge connected pixels, thereby generating precise line segments [20].

(1) Use the Sobel operator to calculate the gradient intensity G and gradient direction of each pixel in the image θ.

$$\begin{cases} G = \sqrt{G_x^2 + G_y^2} \\ \theta = \arctan\left(\dfrac{G_y}{G_x}\right) \end{cases} \tag{1}$$

In the formula, G_x and G_y are the gradients of the image in the x and y directions, respectively, calculated using the Sobel operator. θ is the gradient direction, representing the direction of change of the pixels in the image.

(2) Quantify the gradient direction θ to a fixed order of magnitude to reduce computational complexity.

$$\theta_q = round\left(\frac{\theta}{\Delta\theta}\right) \times \Delta\theta \tag{2}$$

In the formula, $\Delta\theta$ is the quantization step size, and θ_q is the quantized gradient direction.

(3) Each pixel point votes on the assigned quantization direction θ_q, based on its gradient strength $G(x, y)$, and the higher the updated vote value $V(x, y, \theta_q)$ is, the higher the probability of a straight line exists in that direction.

$$V(x, y, \theta_q) = v(x, y, \theta_q) + G(x, y) \tag{3}$$

In the formula, $v(x, y, \theta_q)$ is the voting value in the direction θ_q, and $G(x, y)$ is the gradient intensity of pixel point (x, y).

(4) Based on the candidate line segments, refine the position of the line segments through regional growth. Starting from the seed point (pixel point with high vote value), generate complete line segments by connecting adjacent pixel points.

$$L(x, y) = \begin{cases} 1 & if\ V(x, y, \theta_q) > T \\ 0 & otherwise \end{cases} \tag{4}$$

In the formula, T is the threshold value of the vote, and $L(x, y)$ indicates whether the pixel point (x, y) belongs to the line segment.

3 Dynamic Feature Removal Based on Improved YOLOv8m Fusion

To mitigate the impact of dynamic objects in the environment on visual SLAM localization, this paper proposes a dual dynamic feature removal method that integrates object detection with epipolar geometric constraints. First, the improved YOLOv8m algorithm is used to detect objects within the image, and the detected objects are classified into static and dynamic categories, enabling the initial removal of dynamic features. Based on this, epipolar geometric constraints are applied to the remaining feature points after the preliminary screening. By calculating the distance from each

feature point to its corresponding epipolar line and comparing it with a preset threshold, points with distances exceeding the threshold are identified as residual dynamic features and are subjected to secondary removal. Figure 2 illustrates the flowchart of the dual dynamic feature removal algorithm.

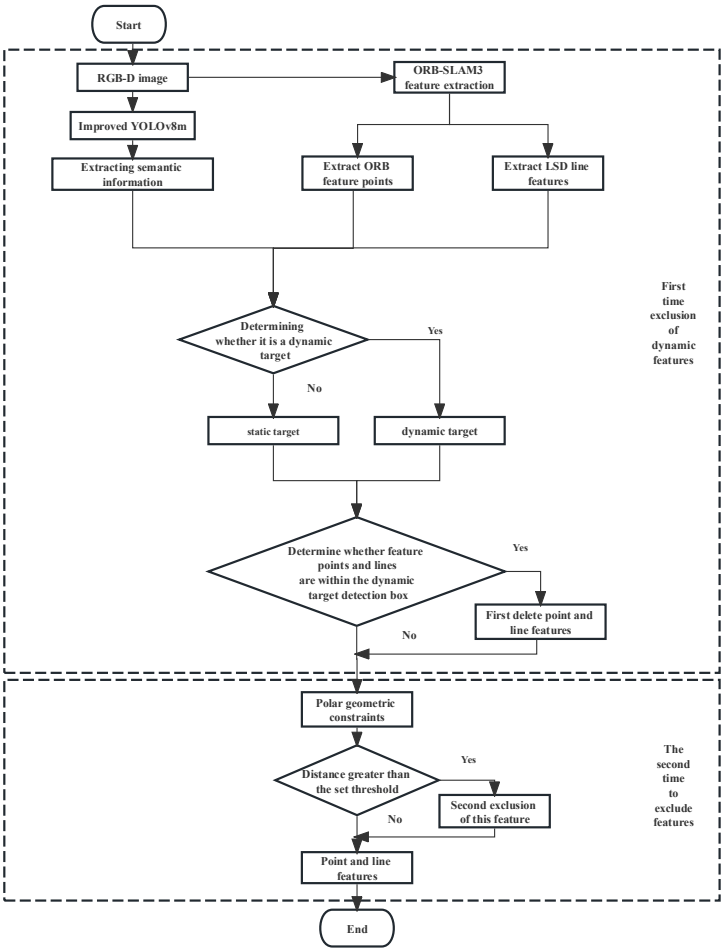


Figure 2 Flowchart of the dual dynamic feature removal algorithm

3.1 Object Detection and Feature Removal Based on Improved YOLOv8m

Since its introduction in 2015, the YOLO algorithm has undergone numerous improvements and has become a significant algorithm for real-time object detection. Its models, trained across edge devices, enhance privacy, adaptability, and generalization [21]. The latest version, YOLOv8, incorporates an instance segmentation model based on YOLACT, enabling the algorithm to not only perform object detection but also achieve pixel-level segmentation of detected objects, further improving its accuracy. YOLOv8 is available in five model sizes: YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l, and YOLOv8x. This paper selects YOLOv8m as the base network and introduces the GhostConv module and the CBAM attention mechanism for lightweight

improvements. Real-time analysis, as detailed in the ablation study in Section 5.1, shows that the improved YOLOv8m maintains high detection accuracy while effectively reducing computational complexity, enabling the overall system to achieve a frame rate of 26.7 FPS. This meets the real-time requirements of SLAM systems in indoor dynamic scenes.

The YOLOv8m structure mainly consists of three core components: the Backbone, Neck, and Head modules, as illustrated in Figure 3. The Backbone, as the foundational module, typically utilizes a deep convolutional neural network (CNN) to extract features from the image. It captures useful feature information from the input image, which will be used for subsequent object detection tasks. The Neck module contains a series of layers designed to process and fuse features extracted from the Backbone network. These layers include structures like Feature Pyramid Networks (FPN) or Path Aggregation Networks (PANet), which facilitate feature fusion at different scales, thereby improving the algorithm's ability to detect objects of various sizes. The Head module is responsible for performing the final prediction in YOLOv8m. The convolutional and fully connected layers within this module are used to generate bounding box coordinates, class probabilities, and confidence scores for the detected objects.

This paper adopts YOLOv8m, known for its high stability within the YOLO series, as the dynamic object detection algorithm [22-24]. To meet the requirements of object detection in dynamic environments and real-time system operation, this section introduces improvements to the YOLOv8m network. The original convolutions in the YOLOv8m network are replaced with lightweight GhostConv modules. Additionally, the CBAM attention mechanism is incorporated into the backbone network to enhance its ability to capture important features. The structure of the improved YOLOv8m network is shown in Figure 3.

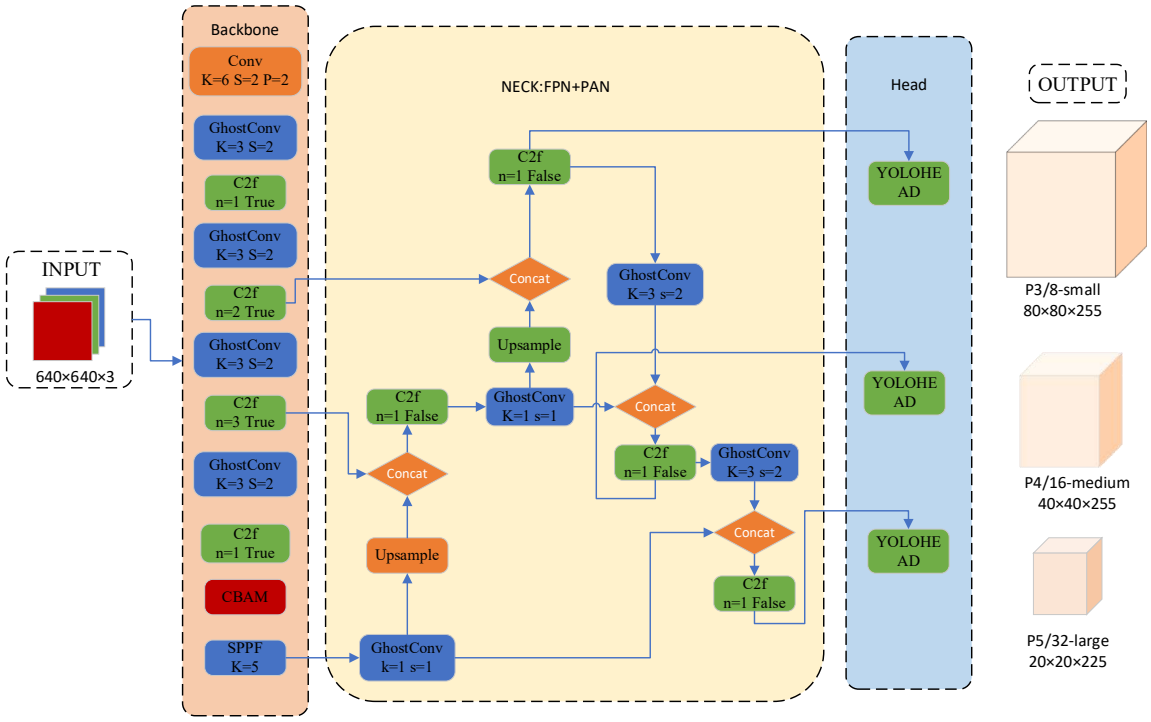


Figure 3 Structure of the improved YOLOv8m network

1) Ghost module

To make the YOLOv8m network lightweight, a lightweight Ghost module was introduced into the backbone network of YOLOv8m, as shown in Figure 4. Unlike conventional convolutions that directly generate all feature maps, the Ghost module avoids using a large number of parameters to generate redundant feature maps individually. Instead, it uses linear transformation operations (cheap operations) to generate these redundant feature maps [25-27]. The Ghost module divides the original convolutional layer into two parts: The first part uses a small number of convolution kernels to generate the original feature maps; the second part performs cheap operations ϕ via depth-separable convolution (Conv) to generate redundant feature maps and uses an identity mapping (Identity) to directly output the original feature maps; finally, the feature maps generated by the above two parts (i.e., the original feature maps and redundant feature maps) are concatenated to obtain the final output feature maps. This design significantly reduces the computational load of the Ghost module.

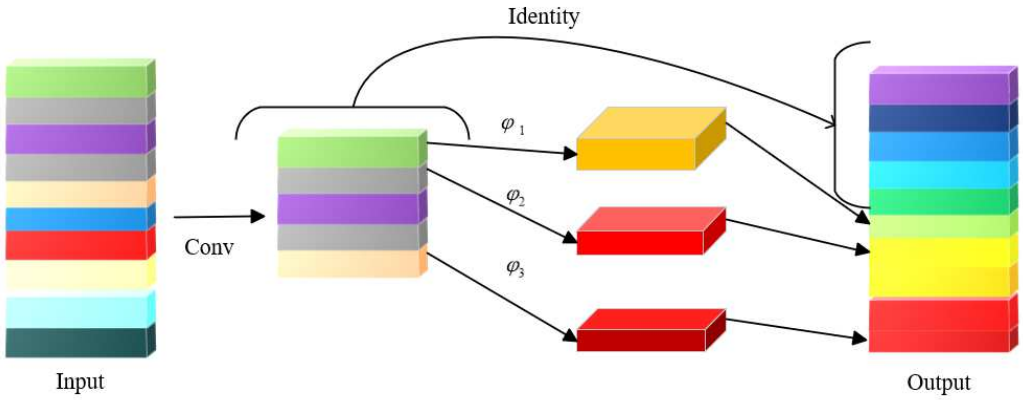


Figure 4 Ghost Module

2) CBAM Attention Mechanism

Integrating the CBAM attention mechanism into YOLOv8m enhances the model's ability to perceive key features [28-30]. CBAM (Convolutional Block Attention Module) is an efficient attention mechanism composed of a channel attention module (Channel Attention Module) and a spatial attention module (Spatial Attention Module) [31-33]. By leveraging both channel and spatial attention mechanisms, the model dynamically adjusts feature weights to focus more intently on the target's critical regions. The channel attention module filters important feature channels to enhance target semantic information; the spatial attention module locates the target's spatial position to suppress background interference. This mechanism effectively improves detection accuracy for small and occluded targets in complex scenes, particularly in scenarios with multi-scale, low contrast, or dense targets. Additionally, the CBAM module's lightweight design incurs minimal computational overhead, preserving YOLOv8m's real-time performance advantage. The CBAM attention mechanism is illustrated in Figure 5.

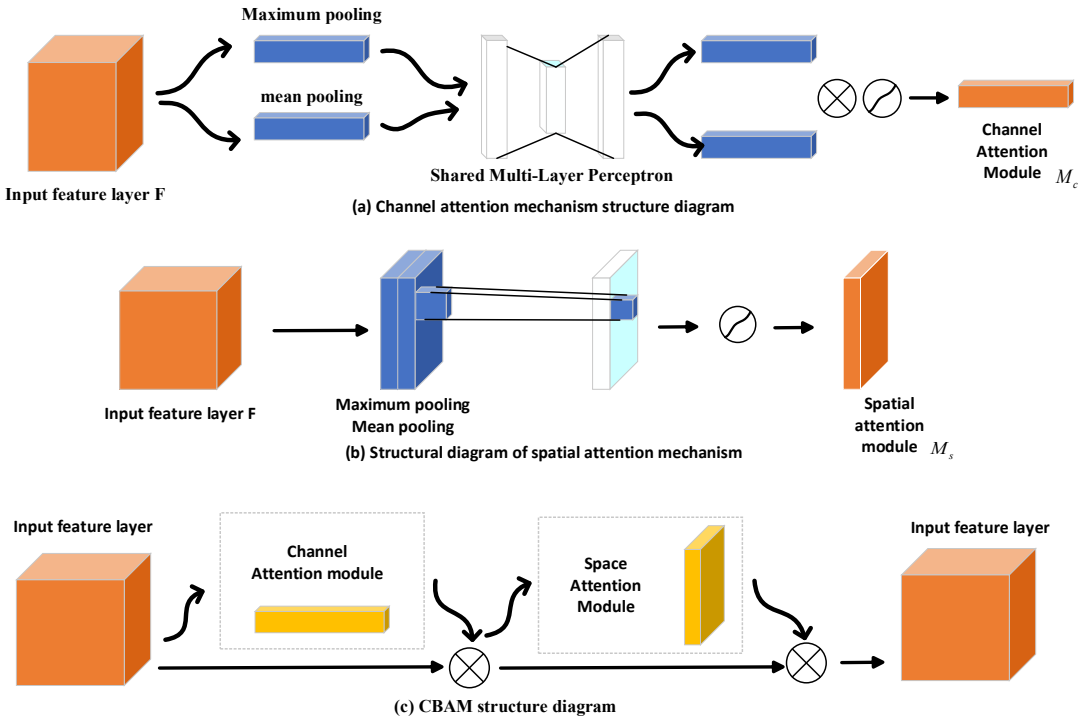


Figure 5 CBAM Attention Mechanism

Channel Attention Module: This module compresses each channel of the feature map through a global average pooling operation to obtain a global feature representation of the channel. Then, through two fully connected layers and ReLU activation functions, the correlation between channels is calculated, and the channel attention weights are ultimately obtained. The formula can be expressed as:

$$ChannelAttention(F_c) = \sigma(FC_2(\mathrm{Re}\,LU(AvgPool(F_c)))) \tag{5}$$

Where F_c denotes the input feature map, $AvgPool$ is the global average pooling operation, FC_1 and FC_2 are the fully connected layers, and σ is the Sigmoid function.

Spatial attention module: This module captures the spatial relationships between pixels in feature maps through convolution and pooling operations to obtain spatial attention weights. The formula can be expressed as:

$$\begin{aligned} &SpatialAttention(F_s) \\ &= \sigma(Conv(Concat(MaxPool(F_s) \\ &, AvgPool(F_s)))) \end{aligned} \tag{6}$$

where F denotes the input feature map, $MaxPool$ and $AvgPool$ are the maximum pooling and average pooling operations, respectively, $Conv$ is the convolution operation. Concat is the channel splicing operation, and σ is the $Sigmoid$ function.

In indoor environments, moving objects in dynamic scenes can affect the accuracy and real-time performance of object detection. This paper employs an improved

YOLOv8m algorithm for object detection in indoor dynamic environments, enabling the segmentation of dynamic objects (e.g., people) and static objects (e.g., chairs). First, preprocessed images are input into the Backbone module for feature extraction. This module adopts a C2f architecture, utilizing convolutional layers and residual structures to extract deep features. Next, the extracted features are input into the Neck module for cross-layer fusion and multi-scale feature enhancement. Finally, the fused feature maps are decoded in the Head module to generate bounding boxes, categories, and confidence information for the objects. This process uses non-maximum suppression to filter the final detection results, thereby achieving regional segmentation and labeling of objects in the image. Since dynamic objects typically exhibit significant speed changes, while static objects undergo minimal displacement or rotation, the algorithm can further distinguish whether the extracted feature points and lines belong to dynamic or static objects after obtaining the detection boxes. If the feature points and lines are located within the detection box of a dynamic object and not within the detection box of a static object, they are identified as dynamic features and removed.

3.2 Secondary Feature Removal Using Epipolar Geometric Constraints

During feature tracking and matching, the projections of static feature points should strictly lie on the polar line, while the projections of dynamic feature points deviate from this geometric constraint. As a result, in indoor dynamic environments, object detection is prone to false positives or false negatives due to background motion or rapidly moving objects. To address this issue, this paper introduces polar geometric constraints to process residual dynamic features after preliminary removal, effectively filtering out interference in dynamic backgrounds [34-35]. The polar geometric constraint leverages the geometric relationships between feature points in two consecutive frames to infer camera motion and spatial structure, and accurately identify outliers. In particular, the polar geometric constraint based on the fundamental matrix can effectively filter out feature points of dynamic objects.

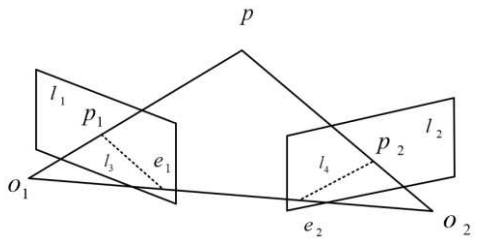


Figure 6: Geometric constraints on the poles

Matching feature points P_1 and P_2 are extracted from two images l_1 and l_2, O_1 and

O_2, are the optical centers of the camera, as shown in Fig. 6. When the matching relationship is correct, extending the line segments of O_1P_1 and O_2P_2, will intersect at the point p in 3D space. The baseline is the line connecting O_1 and O_2, and the polar points (e_1 and e_2) are the intersections of the baseline with the two images. The polar plane consists of the baseline and the point P. The intersections of the polar planes with the images I_1 and I_2, are the I_3 and I_4 poles.

Theoretically, the correct matching points should lie on the polar lines of the respective images. However, due to the interference of numerous factors, matching errors may arise in dynamic environments. In this case, the accuracy of the matching can be evaluated by calculating the distance d from the matching point to the corresponding polar line. For example, a pairwise pole geometry constraint analysis is performed based on p. If the distance d exceeds the set threshold D, p_2 is determined to be an incorrectly matched dynamic point, and the algorithm will reject it with secondary accuracy so that it will no longer participate in the subsequent bit position estimation process.

$$d = \frac{\left| p_2^T F p_1 \right|}{\sqrt{\|A\|^2 + \|B\|^2}} \tag{7}$$

Where p_1 and p_2 are the feature points in the previous frame and the current frame, respectively. F is the fundamental matrix, which represents the geometrical relationship between the two frames. A and B are the line vectors of the polar lines. D is the distance of the feature point to the corresponding polar line, i.e., the offset distance.

4 P3P Pose Estimation Assistance

During pose estimation, feature points in indoor dynamic environments can interfere with the accuracy and stability of the estimation. The P3P algorithm exhibits simplicity, high accuracy, and strong adaptability in indoor dynamic scenes. It is another approach to solving the PnP problem, as it requires only three pairs of matched points and thus has relatively low data requirements [36]. This paper introduces the P3P algorithm, which utilizes three 3D points and their corresponding 2D projection points to enhance the pose estimation accuracy and efficiency of ORB-SLAM3. The specific steps of the algorithm are as follows:

(1) Given three 3D_2D matching point pairs $\{P_i \leftrightarrow p_i\}_{i=1}^{3}$ construct the geometric constraint equations based on the principle of similar triangles.

$$\begin{cases} \|X_1 - X_2\|^2 = d_{12}^2 \\ \|X_1 - X_3\|^2 = d_{13}^2 \\ \|X_2 - X_3\|^2 = d_{23}^2 \end{cases} \tag{8}$$

where d_{ij} is the cosine distance of the viewpoint formed by the corresponding points in the image plane. By solving Eq. (8), up to 4 sets of possible bit pose solutions can be obtained $\{R_k, t_k\}_{k=1}^4$.

(2) Objective function construction: After selecting the optimal initial solution, establish the ICP nonlinear optimization objective function.

$$\arg\min_{R,t} \sum_{i=1}^{n} \rho\left(\left\| \pi\left(RP_i + t\right) - p_i \right\|_{\sum_i}^2 \right) \tag{9}$$

where $RP_i + t$ is the 3D coordinates in the camera coordinate system, $\pi(\cdot)$ is the camera projection model, $\rho(\cdot)$ is the Huber robust kernel function, and $\sum_i$ is the covariance matrix

(3) The Lie algebra optimization is solved by parameterizing the rotation matrix to the Lie algebra $\Phi \in so(3)$ and establishing the Jacobi matrix.

$$J_i = \frac{\partial \pi}{\partial \phi}\Big|_{\phi=0} = \begin{bmatrix} \frac{f_x}{Z_c} & 0 & -\frac{f_x X_c}{Z_c^2} \\ 0 & \frac{f_y}{Z_c} & -\frac{f_y Y_c}{Z_c^2} \end{bmatrix} \left[I - P_i^{\wedge} \right] \tag{10}$$

Where $\phi = [\phi_1, \phi_2, \phi_3]^T$ is the rotation vector, the direction represents the rotation axis, and the magnitude represents the rotation angle; $\frac{f_x}{Z_c}$ represents the sensitivity of the pixel coordinates to the X-axis of the camera coordinate system, which is positively correlated with the focal length f_x, and negatively correlated with the depth Z_c ; $-\frac{f_x X_c}{Z_c^2}$ represents the nonlinear effect of the change of depth on the pixel coordinates(perspective distortion term); I is the unit matrix derivative of the translation component; $-P_i^{\wedge}$ is the anti-symmetric matrix derivative of the rotation component.

(4) Solve the incremental equation iteratively using the Gauss-Newton method:

$$J^T J \Delta\xi = -J^T r \tag{11}$$

where $\Delta\xi = \left[\Delta\phi^T, \Delta t^T\right]^T$ is the position increment (shifted by the Lie algebra+); $r = \pi\left(RP_i + t\right) - p_i$ represents the reprojection residuals, and the position is updated iteratively along the negative gradient direction by linearizing the approximation of the minimum value point of the error function.

5 Experiments and Results Analysis

To validate the effectiveness of the proposed method, experiments were conducted using the improved ORB-SLAM3 algorithm. This paper performs tests based on the TUM RGB-D indoor dynamic dataset and the EuRoC-MAV dataset. Six sets of dynamic data were selected from the TUM RGB-D dataset, specifically including sitting-static, sitting-xyz, walking-halfsphere, walking-rpy, walking-static, and walking-xyz. Among these, sequences prefixed with "sitting" correspond to low-dynamic scenes where individuals remain seated or move only slowly, resulting in relatively small environmental changes. Sequences prefixed with "walking" correspond to high-dynamic scenes where individuals are walking, accompanied by significant camera displacement and complex environmental variations. The experiments were conducted using the Ubuntu 20.04 operating system, with hardware configuration consisting of an Intel® Core™ i7-13650H processor (2.60 GHz), 16 GB of RAM, and an NVIDIA GeForce RTX 4060 graphics card (8 GB VRAM).

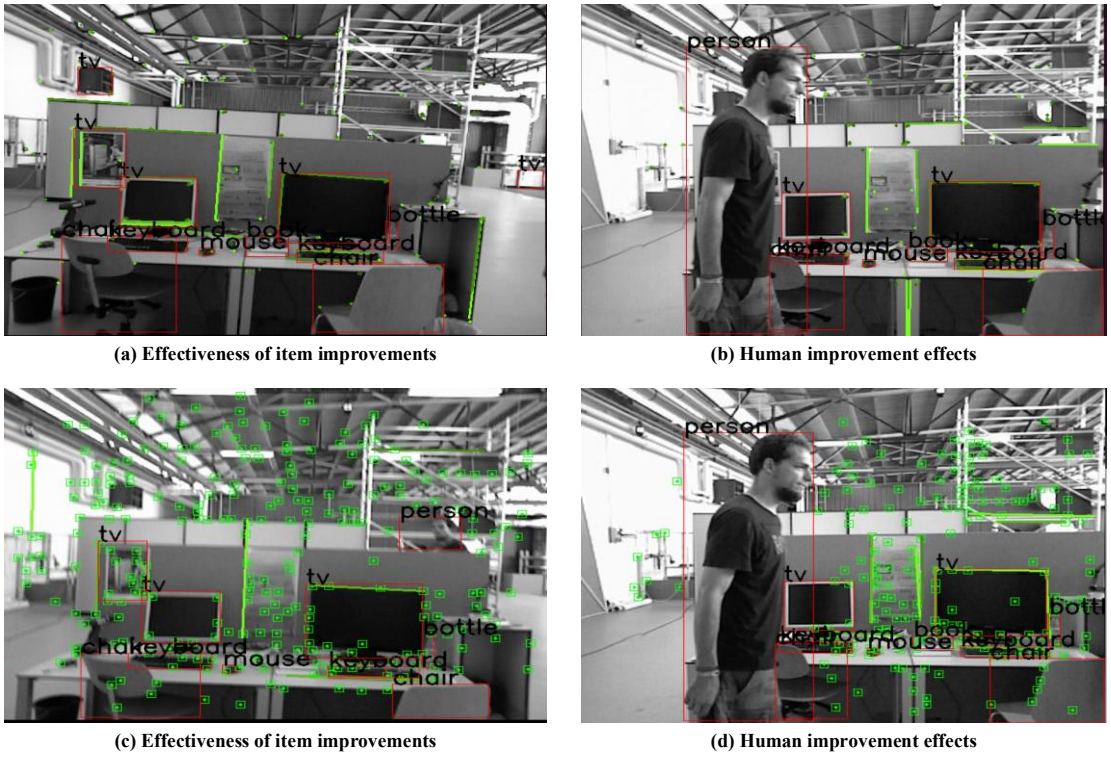
5.1 Ablation Study

To quantitatively analyze the impact of introducing line features, epipolar geometry, and the improved YOLOv8 on the performance of the proposed algorithm, feature removal ablation experiments were conducted. The walking-xyz dynamic dataset was selected for the experiments, and the results are presented in Table 1. The improved algorithm achieves a maximum RMSE improvement rate of up to 93.56%, with an average improvement rate of 31.51%. In terms of frame rate, the accuracy improvement brought by dynamic object detection significantly outweighs the cost of frame rate reduction. It can be concluded that the introduction of line features, epipolar geometry, and the improved YOLOv8 enhances the localization accuracy of the SLAM system in indoor dynamic environments.

Table 1. Exclusion ablation experiment

Experiment number	Line features	Geometry of degrees	Improved YOLOv8	RMSE	Frame rate (FPS)
Baseline				0.205805	50 FPS
1	√			0.030357	53 FPS
2	√	√		0.020325	46 FPS
3	√	√	√	0.018617	26.7 FPS

To validate the effectiveness of incorporating the Ghost module and CBAM attention mechanism into the YOLOv8m algorithm, detection results obtained with the improved YOLOv8m model are presented in Figure 8(c) and (d). It can be observed that compared to Figure 8(a) and (b), the improved model accurately identifies additional features and distant small static objects in the form of feature points and lines.



(a) Effectiveness of item improvements

(b) Human improvement effects

(c) Effectiveness of item improvements

(d) Human improvement effects

Figure 7 Accurate detection of partially occluded and distant small objects by the improved YOLOv8m

5.2 Performance Verification Experiment

To quantitatively analyze the improvement effect, a comparative evaluation was conducted between the original YOLOv8m and the improved YOLOv8m on the TUM RGB-D dataset. The results are shown in Table 2. The experiments were performed using identical training parameters and test environment. The evaluation metrics include mean Average Precision (mAP@0.5), inference speed (FPS), number of

parameters (Params), and computational load (GFLOPs).

Table 2 Performance comparison of YOLOv8m before and after improvement

Model	mAP@0.5 (%)	FPS	Params (M)	GFLOPs
YOLOv8m (Original)	72.4	83.2	25.9	79.2
ImprovedYOLOv8m (Ours)	73.1	96.5	18.3	52.6

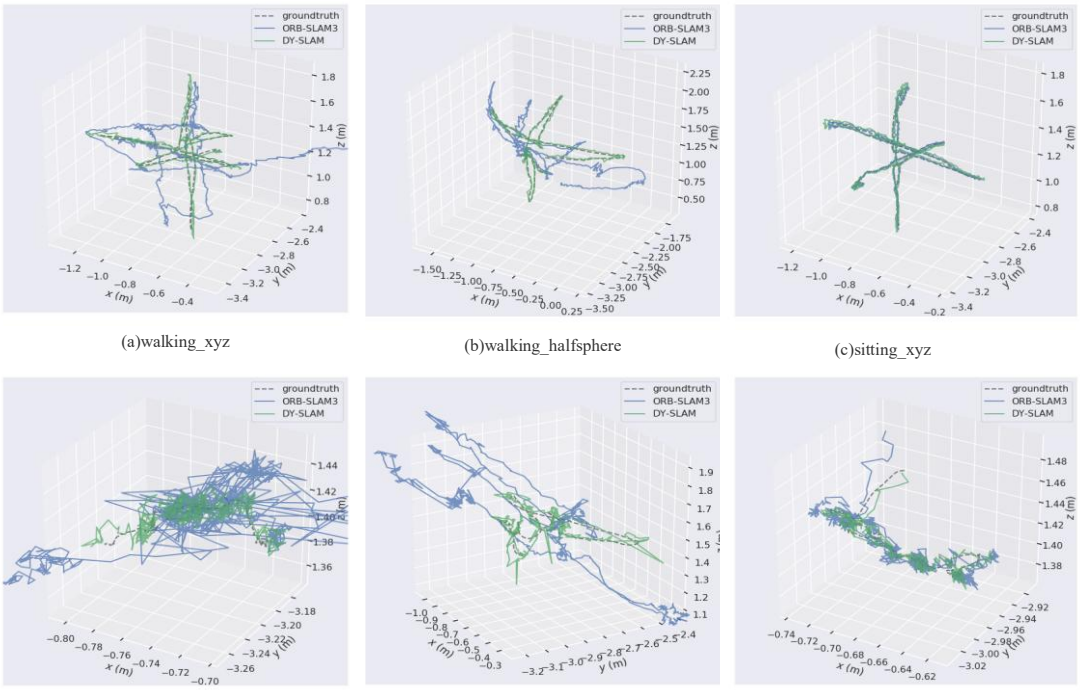
As can be seen from Table 2, the improved YOLOv8m achieves a 0.7 percentage point increase in detection accuracy, indicating that the introduction of GhostConv and CBAM does not sacrifice accuracy; instead, it enhances feature representation capability through the attention mechanism. In terms of inference speed, the improved model shows an increase of approximately 16%. The number of parameters and computational load are reduced by 29.3% and 33.6%, respectively, validating the effectiveness of the GhostConv module in lightweighting. Overall, the improved YOLOv8m maintains high detection accuracy while significantly reducing computational overhead, enabling it to be better integrated into the SLAM system and meet the real-time requirements of indoor dynamic scenes.

5.3 TUM-RGB-D Indoor Dynamic Dataset Validation

To validate the improvement effect of the proposed DY-SLAM algorithm over ORB-SLAM3, the Absolute Trajectory Error (ATE) is employed as an evaluation metric to quantify the difference between the estimated poses and the ground truth poses. Specific evaluation indicators include the Root Mean Square Error (RMSE), Mean error, and Standard Deviation (STD). By comparing the trajectory computed during mapping with the ground truth trajectory provided by the dataset, the localization error of the DY-SLAM algorithm can be visually analyzed.

Experiments were conducted using the TUM RGB-D public dataset for benchmarking. This dataset contains multiple sequences recorded in typical indoor scenes, featuring diverse types and motion patterns of dynamic objects. For instance, the "walking" sequences, such as walking-xyz, walking-halfsphere, and walking-rpy, involve highly dynamic behaviors like two people walking in the scene and moving chairs. In contrast, the "sitting" sequences, such as sitting-xyz and sitting-static, represent low-dynamic scenes with two subjects sitting and conversing, accompanied by slight body movements. Figure 8 compares the trajectory errors for six representative

sequences: walking-xyz, walking-halfsphere, sitting-xyz, walking-static, walking-rpy, and sitting-static. Analysis reveals that in all three scenarios, the DY-SLAM algorithm's estimated trajectories are closer to the ground truth compared to ORB-SLAM3, indicating superior localization accuracy of DY-SLAM. This improvement is particularly evident in the three highly dynamic datasets — walking-xyz, walking-halfsphere, and walking-rpy — demonstrating the algorithm's robust application performance in dynamic scenes.



(a)walking_xyz (b)walking_halfsphere (c)sitting_xyz

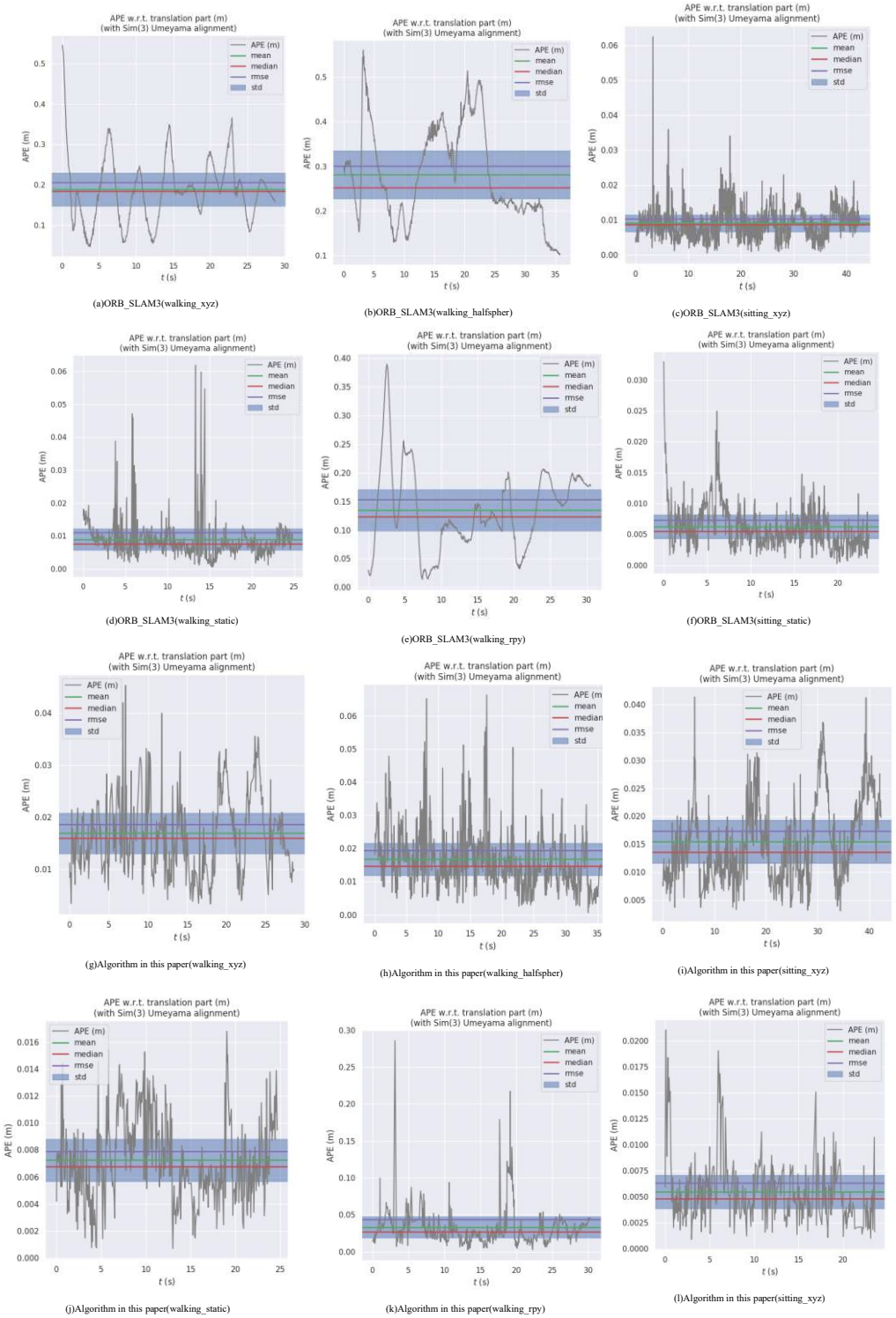
(d)walking_static (e) walking_rpy (f)sitting_static

Figure 8 Comparison of trajectory error between the proposed algorithm and ORB-SLAM3

The comparison of absolute trajectory error distributions is shown in Figure 9. Across the six highly dynamic scenes, compared to ORB-SLAM3, the proposed algorithm reduces the maximum peak error by 91.70%, the minimum error by 99.37%, and the average error by 36.73%, demonstrating a significant improvement in localization accuracy.

In low-dynamic scenes, the error difference is relatively small. The original ORB-SLAM3 algorithm utilizes all feature points in the image frames for matching and tracking, while the improved algorithm identifies some static feature points as dynamic. This indicates a phenomenon of feature loss in such scenes. Additionally, evaluating the fluctuation of ATE assesses the algorithm's environmental adaptability and stability.

The ATE fluctuation of the proposed algorithm is highly concentrated, particularly evident in the four high-dynamic sequences, where the fluctuation range is reduced by an average factor of ten. This indicates that the proposed SLAM system maintains stable performance in dynamic scenes, is less susceptible to external interference or noise, and exhibits good robustness.



(a)ORB_SLAM3(walking_xyz) (b)ORB_SLAM3(walking_halfspher) (c)ORB_SLAM3(sitting_xyz)

(d)ORB_SLAM3(walking_static) (e)ORB_SLAM3(walking_rpy) (f)ORB_SLAM3(sitting_static)

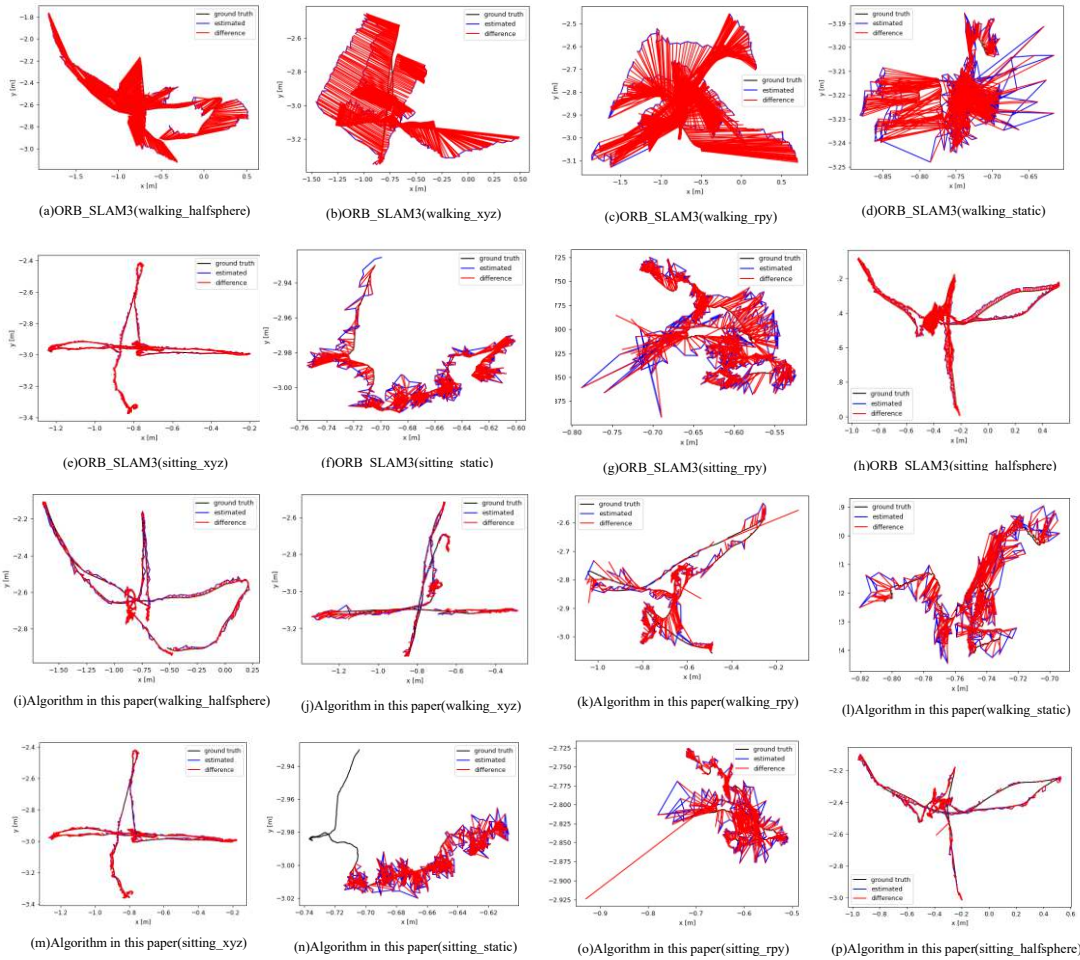
(g)Algorithm in this paper(walking_xyz) (h)Algorithm in this paper(walking_halfspher) (i)Algorithm in this paper(sitting_xyz)

(j)Algorithm in this paper(walking_static) (k)Algorithm in this paper(walking_rpy) (l)Algorithm in this paper(sitting_xyz)

Figure 9 Concentration diagram of absolute trajectory error distribution for the

proposed algorithm

Figure 10 presents a comparison between the estimated trajectories of the ORB-SLAM3 algorithm and the proposed algorithm against the ground truth trajectory. In Figure 10, the dashed line represents the ground truth trajectory, the solid line represents the estimated trajectory, and the shaded red area indicates the error between the estimated pose and the ground truth pose. A smaller shaded area signifies higher localization accuracy of the system. As can be observed from Figure 10, in the dynamic sequences, the trajectory error of the proposed algorithm is significantly smaller than that of the ORB-SLAM3 algorithm.



(a)ORB_SLAM3(walking_halfsphere) (b)ORB_SLAM3(walking_xyz) (c)ORB_SLAM3(walking_rpy) (d)ORB_SLAM3(walking_static)

(e)ORB_SLAM3(sitting_xyz) (f)ORB_SLAM3(sitting_static) (g)ORB_SLAM3(sitting_rpy) (h)ORB_SLAM3(sitting_halfsphere)

(i)Algorithm in this paper(walking_halfsphere) (j)Algorithm in this paper(walking_xyz) (k)Algorithm in this paper(walking_rpy) (l)Algorithm in this paper(walking_static)

(m)Algorithm in this paper(sitting_xyz) (n)Algorithm in this paper(sitting_static) (o)Algorithm in this paper(sitting_rpy) (p)Algorithm in this paper(sitting_halfsphere)

Figure 10 Estimated trajectory of the proposed algorithm is closer to the ground truth trajectory

Table 3 summarizes the ATE evaluation results for the six datasets. Observing the RMSE, Mean, and STD metrics, the proposed algorithm demonstrates reduced overall error compared to ORB-SLAM3, exhibiting more stable localization capability. In the highly dynamic scenes of walking-xyz, walking-halfsphere, walking-rpy, and walking-

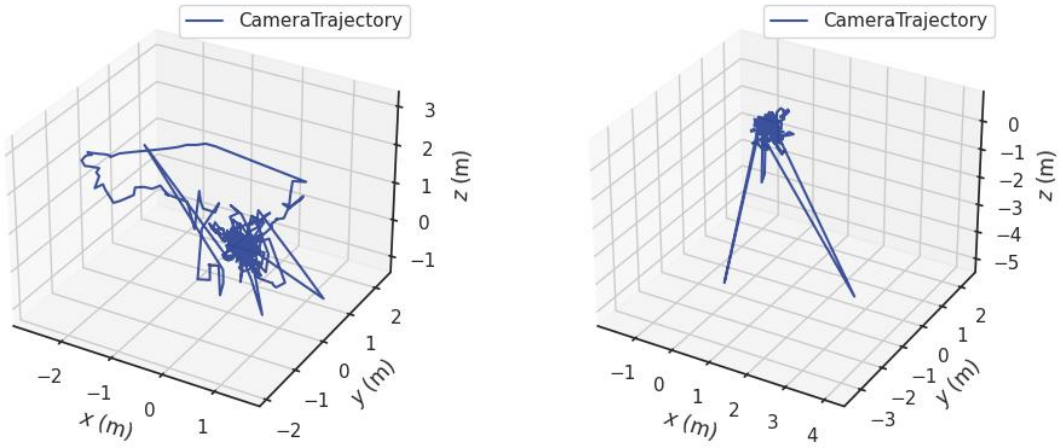
static, the RMSE of the proposed algorithm is reduced by 90.95%, 93.56%, 71.42%, and 28.56%, respectively, compared to ORB-SLAM3. In the two low-dynamic scenes, sitting-xyz and sitting-static, the RMSE of the proposed algorithm is reduced by 20.86% and 13.23%, respectively, compared to ORB-SLAM3. Across all dynamic scenes, the RMSE is improved by an average of 53.10%, with a maximum improvement of up to 93.56%, demonstrating good accuracy and robustness. This provides a new research direction for SLAM systems in dynamic scenes. In summary, compared to the traditional ORB-SLAM3, the proposed algorithm achieves a significant improvement in absolute trajectory error performance.

Table 3 Comparison of absolute trajectory error (ATE) results

Dataset	ORB-SLAM3/m			Ours /m		
	RMSE	Mean	SD	RMSE	Mean	SD
walking-xyz	0.205805	0.188608	0.082359	0.018617	0.016935	0.007733
walking-halfsphere	0.301187	0.281366	0.107455	0.019395	0.016804	0.009684
walking-rpy	0.152831	0.134963	0.071711	0.043671	0.033233	0.028333
walking-static	0.011061	0.008929	0.006529	0.007901	0.007249	0.003141
sitting-xyz	0.021881	0.019171	0.010549	0.017316	0.015508	0.007704
sitting-static	0.007313	0.006242	0.003810	0.006345	0.005479	0.003201

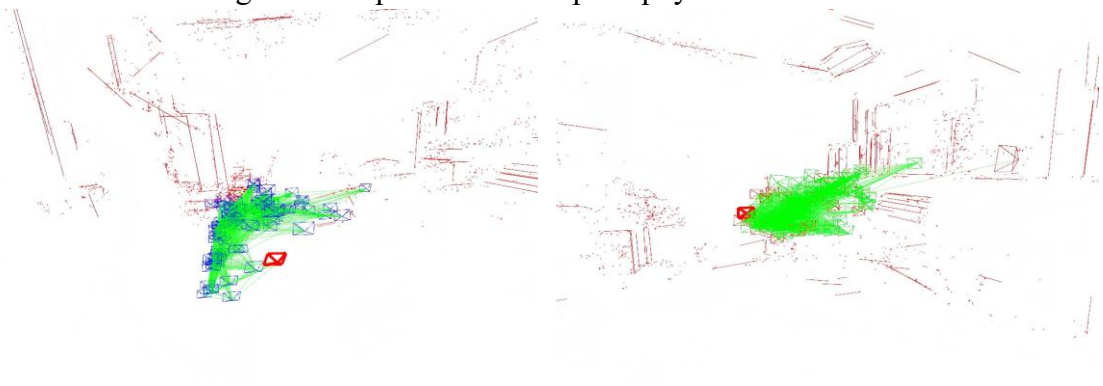
5.4 Physical Validation

To validate the performance of the proposed algorithm in real dynamic scenes, two sets of experiments were designed using a mobile robot in a real-world environment. The experimental scenarios and the actual trajectory are shown in Figure 11. In Experiment 1, the mobile robot remained stationary while a person moved in front of the camera. The effect of dynamic feature point removal is shown in Figure 12(a). In Experiment 2, the mobile robot moved at a constant speed along a straight line, while a person moved in front of the robot to simulate a dynamic environment. The effect of dynamic feature point removal is shown in Figure 12(b). The experimental results in the real environment demonstrate that the proposed dynamic feature removal method can effectively detect dynamic objects and eliminate dynamic features while retaining static features, thereby improving the pose estimation accuracy of the algorithm in dynamic environments.



(a) Stationary trajectory (b) Movement trajectory

Figure 11 Experimental setup for physical validation



(a) Robot movement (b) Robot at rest

Figure 12 Effective removal of dynamic feature points by the proposed algorithm

5.5 Comparative Analysis with Other Dynamic SLAM Algorithms

Based on the TUM RGB-D dataset, this paper also conducts a comparative analysis with other excellent dynamic SLAM algorithms. The experimental results are presented in Table 4. Compared with DS-SLAM and DynaSLAM, the proposed algorithm achieves superior overall RMSE performance. The average RMSE of the proposed algorithm across the six tested datasets is reduced by 37.19% compared to DS-SLAM and by 9.02% compared to DynaSLAM. These comparative results demonstrate that the proposed algorithm exhibits enhanced localization accuracy and superior overall SLAM performance.

Table 4 Comparison of absolute trajectory error (RMSE) among different algorithms

Dataset	DS-SLAM/m	DynaSLAM/m	Ours /m
walking-xyz	0.120445	0.018992	0. 018617
walking-halfsphere	0.020051	0.026142	0. 019395

walking-rpy	0.137285	0.033947	0. 043671
walking-static	0.009696	0.008998	0.007901
sitting-xyz	0.055209	0.033562	0.017316
sitting-static	0.005368	0.006011	0. 006345

5.6 Comparative Analysis on the EuRoC MAV Dataset

For this experiment, the EuRoC MAV dataset was selected as an additional validation dataset. This dataset is a classic public dataset in the field of indoor micro aerial vehicle visual SLAM, collected by ETH Zurich. It contains multi-source data including monocular, stereo vision, and IMU. The scenes cover typical structured environments such as indoor laboratories and industrial workshops, presenting complex conditions like illumination changes, fast motion, and partial occlusion. Importantly, it includes scenarios with interference from dynamic objects (e.g., people walking, equipment moving), which aligns well with the application goals of the proposed algorithm for indoor dynamic scenes. This allows for effective validation of the algorithm's robustness and generalization capability. Considering scene complexity and dynamic characteristics, six representative sequences listed in Table 5 were selected, covering easy, medium, and hard difficulty levels.

Table 5 On the EuRoC-MAV dataset

Dataset	ORB-SLAM3/m			Ours /m		
	RMSE	Mean	SD	RMSE	Mean	SD
MH_01_easy	0.0825	0.0713	0.0356	0.0102	0.0089	0.0041
MH_02_medium	0.1568	0.1389	0.0625	0.0215	0.0187	0.0092
MH_04_difficult	0.3892	0.3567	0.1589	0.0563	0.0498	0.0235
V1_02_medium	0.2156	0.1987	0.0892	0.0328	0.0289	0.0126
V1_03_difficult	0.5237	0.4896	0.2158	0.0789	0.0702	0.0315
V2_02_medium	0.1895	0.1678	0.0763	0.0286	0.0251	0.0108

On the EuRoC-MAV dataset, the improved algorithm achieves an average reduction in ATE-RMSE of 85.32%. Notably, on the challenging sequences (MH_04_difficult, V1_03_difficult), the improvement exceeds 85.5%, demonstrating that the algorithm's localization accuracy in scenes combining high dynamic motion and strong dynamic interference is significantly superior to the original ORB-SLAM3. Under complex operating conditions such as illumination changes (V1_02_medium), partial occlusion (V1_03_difficult), and low texture (MH_02_medium), the ATE-StD of the improved algorithm is consistently controlled within 0.035m, substantially lower than that of the original algorithm. This indicates a marked enhancement in the

algorithm's anti-interference capability, stability, and good generalization ability. The original ORB-SLAM3 exhibits significant trajectory drift in sequences containing moving people or equipment due to interference from dynamic object feature points. In contrast, the improved algorithm, through its dual dynamic feature removal mechanism combining improved YOLOv8m and epipolar geometry, effectively filters out dynamic feature points while retaining valid features from the static environment, resulting in a substantial reduction in localization error.

6 Conclusion

This paper proposes a dynamic visual SLAM algorithm based on deep learning. Built upon the ORB-SLAM3 framework, the method enhances its performance by integrating techniques such as improved YOLOv8m object detection, LSD line feature extraction, epipolar geometric constraints, and P3P pose-assisted localization. Experimental results on the TUM dataset demonstrate that in highly dynamic scenes, the proposed algorithm improves localization accuracy by an average of 53.10% compared to ORB-SLAM3, with a maximum improvement of up to 93.56%. Compared to the excellent deep learning-based visual methods DS-SLAM and DynaSLAM, the proposed algorithm achieves accuracy improvements of 37.19% and 9.02%, respectively. On the EuRoC-MAV dataset, the improved algorithm reduces the average ATE-RMSE by 85.32%, with improvements exceeding 85.5% on challenging sequences, and maintains an ATE-StD below 0.035m under complex operating conditions. Through the dual dynamic feature removal mechanism combining YOLOv8m and epipolar geometry, the algorithm significantly enhances localization accuracy and anti-interference capability. Physical validation further confirms that this method effectively improves the performance of visual SLAM in indoor dynamic scenes.

References

[1] Zhang L, Zhang Y. Improved feature point extraction method of ORB-SLAM2 dense map[J]. Assembly Automation, 2022, 42(4): 552-566.

[2] Gao R, Qi Y. Monocular Object-Level SLAM Enhanced by Joint Semantic Segmentation and Depth Estimation[J]. Sensors, 2025, 25(7): 2110.

[3]Klein, Georg, and David Murray. "Parallel tracking and mapping for small AR workspaces." 2007 6th IEEE and ACM international symposium on mixed and augmented reality. IEEE, 2007.

[4] Barros A M, Michel M, Moline Y, et al. Toward the Determination of an Algorithm for Simultaneous Localization and Mapping in Nuclear Facilities[J]. IEEE Transactions on Nuclear Science, 2024, 71(5): 992-1000.
[5]Hu W, Lin Q, Shao L, et al. A real-time map restoration algorithm based on ORB-SLAM3[J]. Applied Sciences, 2022, 12(15): 7780.
[6] Yu Y, Chen S, Yang X, et al. Visual SLAM algorithm in dynamic environment based on deep learning[J]. Industrial Robot: the international journal of robotics research and application, 2025, 52(1): 28-35.
[7] Xu T, Chen M, Liu J. TKO‐SLAM: Visual SLAM algorithm based on time‐delay feature regression and keyframe pose optimization[J]. Journal of Field Robotics, 2024, 41(6): 1960-1983.
[8] Jiao S, Li Y, Shan Z. DFS-SLAM: a visual SLAM algorithm for deep fusion of semantic information[J]. IEEE Robotics and Automation Letters, 2024.
[9]Cong P, Liu J, Li J, et al. Ydd-slam: Indoor dynamic visual slam fusing yolov5 with depth information[J]. Sensors, 2023, 23(23): 9592.
[10] Wang K, Yao X, Ma N, et al. PLMOT-SLAM: a point-line features fusion SLAM system with moving object tracking[J]. The Visual Computer, 2024: 1-19.
[11]Chang Z, Wu H, Sun Y, et al. RGB-D visual SLAM based on Yolov4-Tiny in indoor dynamic environment[J]. Micromachines, 2022, 13(2): 230.
[12] Cong P, Liu J, Li J, et al. Ydd-slam: Indoor dynamic visual slam fusing yolov5 with depth information[J]. Sensors, 2023, 23(23): 9592.
[13]Liu H, Luo J. YES-SLAM: YOLOv7-enhanced-semantic visual SLAM for mobile robots in dynamic scenes[J]. Measurement Science and Technology, 2023, 35(3): 035117.
[14]Bescos B, Fácil J M, Civera J, et al. DynaSLAM: Tracking, map**, and inpainting in dynamic scenes[J]. IEEE robotics and automation letters, 2018, 3(4): 4076-4083.
[15]Yu C, Liu Z, Liu X J, et al. DS-SLAM: A semantic visual SLAM towards dynamic environments[C]//2018 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, 2018: 1168-1174.
[16]Wen S , Tao S , Liu X ,et al.CD-SLAM: A Real-Time Stereo Visual–Inertial SLAM for Complex Dynamic Environments With Semantic and Geometric Information[J].IEEE Transactions on Instrumentation and Measurement, 2024:73.

[17] Hu C, Zhang X, Li K, et al. EM-LSD-based visual-inertial odometry with point-line feature[J]. IEEE Sensors Journal, 2023, 23(24): 30794-30804.
[18] Peng G, Cao Y, Chen S, et al. Tightly coupled stereo vision-inertial odometry based on point and line feature[J]. Measurement Science and Technology, 2024, 36(1): 016315.
[19] Su J, Meng J, Hou W, et al. Multi-angle optical image automatic registration by combining point and line features[J]. Sensors, 2022, 22(3): 739.
[20] Wang Z, Cheng Q, Mu X. Ru-slam: A robust deep-learning visual simultaneous localization and mapping (slam) system for weakly textured underwater environments[J]. Sensors, 2024, 24(6): 1937.
[21] Zhang X, Yang Y. Research on Road Pattern Recognition of a Vision-Guided Robot Based on Improved-YOLOv8[J]. Applied Sciences, 2024, 14(11): 4424.
[22]Hakim L, Suhendi A, Pramudita A A. YOLOv8 Based Multi Object Detection for BTS Asset Monitoring[C]//2024 12th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS). IEEE, 2024: 186-189.
[23]Wang H, Yang H, Chen H, et al. A remote sensing image target detection algorithm based on improved YOLOv8[J]. Applied Sciences, 2024, 14(4): 1557.
[24]Qu J, Li Q, Pan J, et al. SS-YOLOv8: small-size object detection algorithm based on improved YOLOv8 for UAV imagery[J]. Multimedia Systems, 2025, 31(1): 1-17.
[25]Yang R X, Lee Y R, Lee F S, et al. An Improved YOLOv5 Algorithm for Bamboo Strip Defect Detection Based on the Ghost Module[J]. Forests, 2024, 15(9): 1480.
[26]Wang T, Zhang S. DSC-Ghost-Conv: A compact convolution module for building efficient neural network architectures[J]. Multimedia Tools and Applications, 2024, 83(12): 36767-36795.
[27]Winata I, Oh J. Lightweight Extraction and Segmentation With Ghost Convolutional and Attention Module Integration for Visual SLAM[J]. International Journal of Control, Automation and Systems, 2024, 22(12): 3595-3605.
[28]Yan J, Zeng Y, Lin J, et al. Enhanced object detection in pediatric bronchoscopy images using YOLO-based algorithms with CBAM attention mechanism[J]. Heliyon, 2024, 10(12).
[29]Lv M, Su W H. YOLOV5-CBAM-C3TR: An optimized model based on transformer module and attention mechanism for apple leaf disease detection[J]. Frontiers in Plant Science, 2024, 14: 1323301.
[30]Chen B, Dang Z. Fast PCB defect detection method based on FasterNet backbone

network and CBAM attention mechanism integrated with feature fusion module in improved YOLOv7[J]. Ieee Access, 2023, 11: 95092-95103.
[31]Xu Y, Xu Z, Mei W, et al. SLCAM: A Lightweight Spatial Location Channel Attention Module for Image Classification[J]. Expert Systems with Applications, 2025: 128100.
[32]Liu H, Zhang Y, Chen Y. A Symmetric Efficient Spatial and Channel Attention (ESCA) Module Based on Convolutional Neural Networks[J]. Symmetry, 2024, 16(8): 952.
[33]Min F, Wu Z, Li D, et al. Coeb-slam: A robust vslam in dynamic environments combined object detection, epipolar geometry constraint, and blur filtering[J]. IEEE Sensors Journal, 2023, 23(21): 26279-26291.
[34]Fu Y, Luo L, Zhong K, et al. High-accuracy fringe projection profilometry without phase unwrapping based on multi-view geometry constraints[J]. Optics Express, 2024, 32(22): 38449-38464.
[35]Yang Y, Pan M, Tang D, et al. Mcov-slam: A multicamera omnidirectional visual slam system[J]. IEEE/ASME Transactions on Mechatronics, 2024, 29(5): 3556-3567.
[36] Bai L, Yang Y, Feng C, et al. Received signal strength assisted perspective-three-point algorithm for indoor visible light positioning[J]. Optics Express, 2020, 28(19): 28045-28059.