

Robot Gaussian-historical relocalization: inertial measurement unit-LiDAR likelihood field matching

Ye-Ming Shen¹, Min Kang¹, Jia-Qiang Yang^{1,2}, Zhong-Hou Cai²

¹School of Automation and Electrical Engineering, Zhejiang University of Science and Technology, Hangzhou, Zhejiang, China

²School of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang, China

Article Info

Article history:

Received Jan 20, 2025

Revised Jul 19, 2025

Accepted Aug 26, 2025

Keywords:

Adaptive likelihood field
LiDAR-inertial measurement
unit fusion
Localization losses
Relocalization
ROS robots

ABSTRACT

Robot localization is a foundational technology for autonomous navigation, enabling task execution and adaptation to dynamic environments. However, failure to return to the correct pose after power loss or sudden displacement (the “kidnapping” problem) can lead to critical system failures. Existing methods often suffer from slow relocalization, high computational cost, and poor robustness to dynamic obstacles. We propose a novel inertial measurement unit (IMU)-LiDAR fusion relocalization framework based on Gaussian historical constraints and adaptive likelihood field matching. By incorporating IMU-derived yaw constraints and modeling historical poses within a 3σ Gaussian region, our method effectively narrows the LiDAR search space. Curvature and normal vector-based feature extraction reduces point cloud volume by 50–70%, while dynamic obstacle filtering via multi-frame differencing and neighborhood validation enhances robustness. An adaptive spiral search strategy further refines pose estimation. Compared to ORB-SLAM3 and adaptive Monte Carlo localization (AMCL), our method maintains comparable accuracy while significantly reducing relocalization time and CPU usage. Experimental results show a relocalization success rate of 84%, average time of 1.68 seconds, and CPU usage of 38.4%, demonstrating high efficiency and robustness in dynamic environments.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Min Kang

School of Automation and Electrical Engineering, Zhejiang University of Science and Technology

Hangzhou, Zhejiang, China

Email: kangmin@zust.edu.cn

1. INTRODUCTION

A key prerequisite for mobile robots to accomplish autonomous navigation is positioning, which entails figuring out the robot’s posture in its operational environment. Tasks such as mapping, obstacle avoidance, area coverage, and target tracking rely on accurate localization. However, one major challenge is the “robot kidnapping” problem [1], where a robot is suddenly relocated to an unknown position without any prior information, causing a loss of localization. To ensure the safe and effective operation of robots in real-world applications such as industrial automation, service robotics, and autonomous driving, it is crucial to address the relocalization problem after the robot loses its positioning [2].

Particle filtering and visual simultaneous localization and mapping (Visual SLAM) are two popular methods in robot relocalization. Visual SLAM uses camera image data to locate robots and create maps, and is strongly affected by changes in lighting [3]–[5]. Particle filtering uses a group of particles to estimate the robot’s state, but it is computationally demanding, particularly in large-scale environments [6]–[10]. These limitations, such as Visual SLAM’s sensitivity to illumination changes and particle filtering’s high computational costs, underscore the need for more reliable and effective relocalization techniques.

At present, many scholars have researched relocalization and achieved fruitful results. For example, Hu *et al.* [11] proposed a fusion positioning method combining Wi-Fi, LiDAR, and maps based on Kalman filtering. This method addresses two issues: the poor performance of single Wi-Fi positioning in indoor environments and incorrect LiDAR positioning due to ineffective feature extraction in specific scenarios. However, it requires substantial computational resources. Wang *et al.* [12] proposed a global positioning method based on the principles of laser odometry and improved adaptive Monte Carlo. They used laser odometry based on the point-to-line iterative closest point (PL-ICP) method to replace the traditional odometry and introduced the idea of deoxyribonucleic acid (DNA) crossover and mutation in genetics into the particle iteration process of AMCL. However, the iterative closest point (ICP) obtains the optimal approximate position solution through iteration. Its heavy calculation often causes the robot's state to change before getting the position, resulting in a large cumulative error. Lan *et al.* [13] proposed narrow field of view (NFOV) error recognition to determine the status of base stations through the sliding window technique and standard deviation threshold, eliminating abnormal data. Graph optimization fusion positioning combines ultra-wide band (UWB) measurement values and odometer information to obtain accurate robot trajectories through optimization algorithms. UWB signals may be affected by environmental interference, affecting the reliability of measurement results. Wang *et al.* [1] suggested a residual network (ResNet)-based robot relocalization technique that combined coarse and fine matching to greatly increase positioning success rate and efficiency. Even though current approaches have some benefits, they still have drawbacks like a slow relocalization response and high computational complexity. Conventional approaches show limited accuracy in dynamic obstacle detection under challenging conditions such as lighting variations and object occlusion, which prevents them from meeting operational requirements in complex environments. Moreover, they often fail to effectively integrate historical pose data and inertial measurement unit (IMU) measurements to enable robust relocalization in dynamic scenarios, resulting in reduced adaptability and slow recovery after disruptions.

In order to overcome these limitations, this study proposes a novel LiDAR-IMU fusion-based relocalization technique. The method fuses IMU yaw angles with historical pose data to effectively compress the search space, and it utilizes curvature and normal vector analysis for feature extraction, significantly reducing data volume while preserving most critical environmental information. Furthermore, we propose a frame-difference-based dynamic obstacle detection approach for accurate dynamic environment discrimination, along with an adaptive likelihood field matching algorithm that dynamically optimizes computational resource allocation according to environmental complexity for LiDAR-to-map matching. The experimental results demonstrate that the proposed method substantially improves system responsiveness and environmental adaptability, providing an effective solution for robot relocalization in complex dynamic scenarios.

2. RELOCALIZATION APPROACH

The relocalization implementation block diagram is shown in Figure 1. It mainly consists of a data input layer, processing module, and output layer. The data input layer includes data sources such as historical coordinates, grid maps, IMU, and LiDAR, responsible for collecting various raw data to provide basic information for the system. The processing module is the core part of the relocalization implementation, it performs analytical processing and data fusion on the input signals to determine the optimal robot pose estimation. The output layer then produces optimal pose estimation based on the processing module's results, determining the robot's position and orientation.

2.1. Multi-source information fusion combining

The multi-source information fusion module deeply integrates historical pose data, grid map information, and feature-extracted LiDAR and IMU measurements. The historical poses and grid map provide macroscopic position and environmental information, offering a global reference framework. Meanwhile, the fused LiDAR and IMU data contribute precise local environmental features and motion state information. This comprehensive integration enhances the robustness and accuracy of the relocalization process in complex dynamic environments.

2.1.1. Gaussian 3σ constraint modeling for historical poses

Data collection involves obtaining the pose data of the latest 10 successful localizations from the robot localization system. Each pose is typically represented as a vector $p = [x, y, \theta]^T$, where x and y are the robot's two-dimensional position coordinates and θ is its yaw angle. Denote these 10 poses as p_1, p_2, \dots, p_{10} . To calculate the mean vector μ , given $n = 10$ data points, for the translation part (x and y) and rotation part (θ), we use $\mu = \frac{1}{n} \sum_{i=1}^n p_i$. In the two-dimensional translation and one-dimensional rotation case, $\mu = [\mu_x, \mu_y, \mu_\theta]^T$ with $\mu_x = \frac{1}{n} \sum_{i=1}^n x_i$, $\mu_y = \frac{1}{n} \sum_{i=1}^n y_i$, and $\mu_\theta = \frac{1}{n} \sum_{i=1}^n \theta_i$.

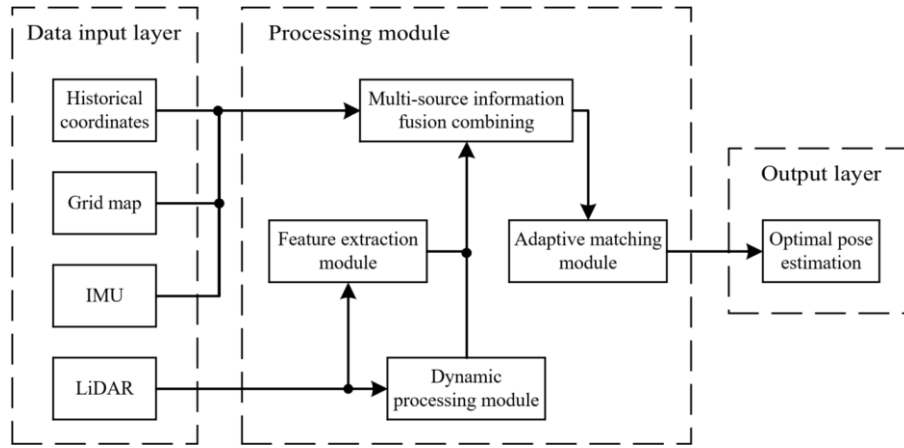


Figure 1. Block diagram of the relocalization implementation

The covariance matrix Σ , a 3×3 matrix for pose data, describes data distribution and inter-dimensional correlations. Its elements are calculated as $\Sigma_{jk} = \frac{1}{n-1} \sum_{i=1}^n (p_{ij} - \mu_j)(p_{ik} - \mu_k)$ for $j, k \in \{1, 2, 3\}$ corresponding to x, y , and θ . Expanded, the covariance matrix is given by (1),

$$\Sigma = \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xy} & \sigma_{x\theta} \\ \sigma_{yx} & \sigma_{yy}^2 & \sigma_{y\theta} \\ \sigma_{\theta x} & \sigma_{\theta y} & \sigma_{\theta\theta}^2 \end{bmatrix} \quad (1)$$

where σ_{xx}^2 and σ_{yy}^2 are x and y variances, $\sigma_{\theta\theta}^2$ is the θ variance, and $\sigma_{xy} = \sigma_{yx}$, $\sigma_{x\theta} = \sigma_{\theta x}$, $\sigma_{y\theta} = \sigma_{\theta y}$ are the respective covariances. The covariance matrix is dynamically adjusted to ensure $\sigma_{xy}^2 = \max(\sigma_{xx}^2, \sigma_{yy}^2) \leq 0.25m^2$ for translation variance and $\sigma_{\theta}^2 = \sigma_{\theta\theta}^2 \leq 0.09rad^2$ for rotation variance, with adjustments $\sigma_{xx}^2 = \min(\sigma_{xx}^2, 0.25)$, $\sigma_{yy}^2 = \min(\sigma_{yy}^2, 0.25)$, $\sigma_{\theta\theta}^2 = \min(\sigma_{\theta\theta}^2, 0.09)$. In a Gaussian distribution, the 3σ range, covering about 99.7% of the data, defines the effective search region. For two-dimensional translation (x and y), it's an ellipse obtained from the covariance matrix's eigenvalue decomposition, and for one-dimensional rotation (θ), it is the interval $[\mu_{\theta} - 3\sqrt{\sigma_{\theta\theta}^2}, \mu_{\theta} + 3\sqrt{\sigma_{\theta\theta}^2}]$. When a new pose $p_{new} = [x_{new}, y_{new}, \theta_{new}]^T$ is observed, the Mahalanobis distance d_M is calculated to measure its deviation from the historical Gaussian model. The formula for the Mahalanobis distance is given by (2).

$$d_M = \sqrt{(p_{new} - \mu)^T \Sigma^{-1} (p_{new} - \mu)} \quad (2)$$

The system initiates the relocalization process if the Mahalanobis distance d_M surpasses a predetermined threshold T , signifying a notable departure from the historical distribution.

2.1.2. IMU pre-integration for relocalization

When using information from an IMU for localization or relocalization, directly integrating raw accelerometer and gyroscope data often suffers from high-frequency noise and accumulated drift. Moreover, since the sampling frequency of the IMU is typically much higher than that of LiDAR, repeatedly integrating IMU data from the initial moment to the current frame incurs considerable computational cost. To address these issues, the pre-integration method has been proposed to efficiently aggregate high-frequency IMU data over the lower-frequency interval between two consecutive keyframes, thereby providing a stable relative motion constraint. The IMU pre-integration method was first formalized by Forster *et al.* [14], enabling efficient incorporation of high-frequency inertial data into optimization-based estimation frameworks without redundant re-integration.

The core idea of IMU pre-integration is to integrate the angular velocity and linear acceleration over the time interval between two frames to obtain a relative pose increment. This approach avoids redundant re-integration from the initial state during each optimization. In addition, to accommodate the updates of state variables (e.g., orientation) during the nonlinear optimization process, the Jacobians of the pre-integrated quantities with respect to the initial states are also computed. This facilitates the incorporation of IMU factors into optimization-based frameworks.

The resulting pre-integrated quantities include the relative rotation increment ΔR_{ij} , the relative velocity increment Δv_{ij} , and the relative position increment Δp_{ij} . After removing the effects of gravity and sensor bias, the IMU pre-integration can be approximated by (3) to (5).

$$\Delta R_{ij} \approx \prod_{k=i}^{j-1} \exp((\omega_k - b_g)\Delta t) \quad (3)$$

$$\Delta v_{ij} \approx \sum_{k=i}^{j-1} R_k(a_k - b_a)\Delta t \quad (4)$$

$$\Delta p_{ij} \approx \sum_{k=i}^{j-1} [v_k\Delta t + \frac{1}{2}R_k(a_k - b_a)\Delta t^2] \quad (5)$$

Here, ω_k and a_k denote the gyroscope and accelerometer measurements at time k , respectively, b_g and b_a represent the gyroscope and accelerometer biases, and R_k is the rotation matrix at time k .

In situations such as robot kidnapping, where the robot's prior pose estimate becomes invalid, IMU pre-integration can provide a relatively stable motion prior to assist LiDAR in pose initialization. This enables rapid relocalization. Furthermore, the high-frequency nature of IMU measurements allows the system to continuously capture orientation changes over short time intervals, thereby improving its responsiveness to abrupt motion.

2.1.3. Dynamic processing module

With its data serving as the foundation of the adaptive likelihood field for assessing position probability distributions, LiDAR is essential for localization and environmental perception. However, there are two main issues with LiDAR data. First, even though 2D point cloud data is rich, dynamic obstacles interference and sensor limitations introduce measurement errors. Second, the massive amount of LiDAR data raises storage requirements and decreases computational efficiency.

The dynamic processing module is mainly responsible for processing LiDAR data. Dynamic objects, such as moving vehicles and pedestrians, are common in real-world applications. Accurate positioning may be compromised by dynamic object interference in LiDAR point clouds [15]. To counteract this interference, the dynamic processing module processes radar data using a multi-frame differential detection technique.

Let the current frame be the c -th frame, with its LiDAR scan distance data sequence represented as $R_c = [r_{c1}, r_{c2}, \dots, r_{cN}]$, where N denotes the number of LiDAR scan points. For M historical frames, the distance data sequence of the m -th historical frame is $R_{hm} = [r_{hm1}, r_{hm2}, \dots, r_{hmN}]$, where $m = 1, 2, \dots, M$. For the i -th scan point, the inter-frame difference Δr_{im} between the current frame and the m -th historical frame is calculated as (6).

$$\Delta r_{im} = |r_{ci} - r_{hmi}| \quad (6)$$

The neighborhood point set for the i -th scan point is defined as \mathcal{N}_i . For each neighboring point $j \in \mathcal{N}_i$, we employ an indicator function $\mathbb{I}(\Delta r_{jm} > T)$ to evaluate whether the inter-frame difference Δr_{jm} exceeds threshold T . The function returns 1 when $\Delta r_{jm} > T$, and 0 otherwise.

If $\sum_{j \in \mathcal{N}_i} \mathbb{I}(\Delta r_{jm} > T) \geq 1$, the i -th scan point is preliminarily identified as potentially dynamic. However, a special case requires consideration before final determination.

In real-world scenarios, when a LiDAR beam switches from a dynamic obstacle to a static background due to object movement, the point may falsely trigger dynamic detection. To resolve this, we analyze both distance change patterns (sudden increase followed by stabilization) and scan angles to distinguish true dynamic points from revealed static features. This approach prevents misclassification when $\sum_{j \in \mathcal{N}_i} \mathbb{I}(\Delta r_{jm} > T) \geq 1$ occurs due to obstacle displacement rather than actual dynamics, substantially reducing false positives in the detection system.

2.1.4. Feature extraction module

Feature extraction techniques address the problem of massive LiDAR data by identifying key feature points and shape information, significantly reducing data volume while retaining critical environmental details. This approach accelerates processing and improves analysis accuracy [16].

Finding jump points—points in the LiDAR data where the distance between adjacent measurements substantially deviates from the expected range, indicating potential object boundaries or data anomalies [17]. the point clouds are segmented using the proper thresholds and rules, and Each group of segmented points is regarded as a local neighborhood. Formula (4) yields the covariance matrix of the local neighborhood, assuming that each local neighborhood contains k point clouds.

$$C = \frac{1}{k} \sum_{i=1}^k (p_i - \bar{p})(p_i - \bar{p})^T \quad (7)$$

Among them, p_i is the point in the neighborhood, and \bar{p} is the centroid of the neighborhood. Then, perform eigenvalue decomposition on the covariance matrix C to obtain three eigenvalues λ_1 , λ_2 , and λ_3 . The curvature is calculated by the ratio of the minimum eigenvalue to the sum of the eigenvalues, as shown in (5). Points with larger curvature usually correspond to corner points or edges in the environment.

$$k = \frac{\min(\lambda_1, \lambda_2, \lambda_3)}{\lambda_1 + \lambda_2 + \lambda_3} \quad (8)$$

The parameters of a circle are fitted by the least squares method [18]. Suppose the equation of a circle is $(x - a)^2 + (y - b)^2 = r^2$. Where (a, b) are the coordinates of the center of the circle and r is the radius. For the points $(x_{z_t^i}, y_{z_t^i})$ (represents the global coordinate system coordinates of the points projected onto the map from the points scanned by the laser) in the given point cloud data. Substitute them into the equation of the circle to construct a series of equations about a , b and r . Then, minimize the sum of the squared distances from the points to the circle, that is, solve for the values that make (6) reach its minimum.

$$\min_{a,b,r} \sum_i^n \left[\sqrt{(x_{z_t^i} - a)^2 + (y_{z_t^i} - b)^2} - r \right]^2 \quad (9)$$

To mitigate the excessive influence of long straight lines during data processing, a strategy of weakening their weight is adopted. This method guarantees the efficient retention of environmental characteristic information. The positioning algorithm can more effectively identify potential positions by using the processed data for probability inference, which improves the system's overall accuracy and robustness.

2.2. Adaptive likelihood field method

The beam model is prone to local optima and high computational costs. The adaptive likelihood field approach overcomes the beam model's drawbacks, particularly its non-smoothness in cluttered environments [19], [20]. The likelihood field model by reducing computational complexity and blurring obstacles provides smoother and more efficient results than the beam model [21], [22]. The probability distribution of the likelihood field [23] can be represented by (7).

$$p(z_t | x_t, m) = \prod_{i=1}^n p(z_t^i | x_t, m) \quad (10)$$

Among them, z_t^i is the i -th LiDAR measurement value at time, x_t is the pose of the robot at time, and m is the map. The likelihood field model blurs the obstacles in the workspace, making it applicable to various spatial situations with smoother and more efficient results. Its core idea is to regard the points on the grid as forming a magnetic field that attracts the surrounding point clouds, and the attraction decays with the square of the distance (or Gaussian decay can be used) [24], [25]. The endpoints obtained by LiDAR measurement in the global coordinate system are projected. At time t , the posture of the robot is $x_t = (x, y, \theta)^T$, the installation position of the LiDAR relative to the center coordinates of the robot is $(x_{k,sens}, y_{k,sens})^T$, the angle of the laser beam of the LiDAR relative to the orientation of the robot is $\theta_{k,sens}$, the coordinates of the laser-measured endpoints relative to the center of the LiDAR is z_t^k , and the coordinates of the points scanned by the laser projected onto the global coordinate system of the map is $(x_{z_t^k}, y_{z_t^k})$, as shown in Figure 2, their relationships are described by (8). This technique offers a more dependable method for robot relocalization while successfully addressing the drawbacks of the beam model. The robot uses this formula to update its position estimate in the global coordinate system based on its current posture and sensor measurements, achieving precise positioning.

$$\begin{pmatrix} x_{z_t^k} \\ y_{z_t^k} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x_{k,sens} \\ y_{k,sens} \end{pmatrix} + z_t^k \begin{pmatrix} \cos(\theta + \theta_{k,sens}) \\ \sin(\theta + \theta_{k,sens}) \end{pmatrix} \quad (11)$$

During relocalization initialization, the system constructs a 3σ -constrained search region on the 2D plane by leveraging real-time IMU yaw angle measurements (θ). This approach reduces the LiDAR heading search range to $\pm 30^\circ$ (an 83.3% reduction compared to full 360° search) while initializing the positional search space using historical trajectory variance constraints ($\sigma_{xx}^2, \sigma_{yy}^2 \leq 0.25m^2$)

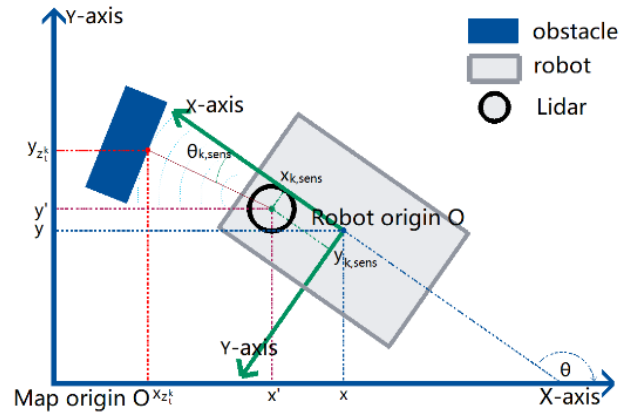


Figure 2. Robot coordinate transformation

For each candidate pose $[x, y, \theta]^T$, the system processes LiDAR data through dynamic obstacle removal and static feature extraction, then calculates the likelihood of matching between environmental features (edges/planes) and pre-built map elements. The iterative optimization maximizes the likelihood function $p(z_t | x_t, m)$, where z_t denotes current observations, x_t represents candidate poses, and m is the static map.

If the maximum matching score within the initial search region falls below the predefined threshold, the system activates an adaptive spiral expansion strategy as shown in Figure 3. Taking the initial search center as the origin, it dynamically adjusts the search step size using $L_k = L_0 \cdot r^{-k}$ (where $L_0 = 0.5 \text{ m}$ is the initial step size and $r = 0.8$ is the decay rate), extending the search area in a spiral pattern outward. The feature matching and optimization procedure is repeated following each expansion until a pose that satisfies the requirements is discovered. The final candidate pose is validated twice: first, it is examined for validity within the boundaries of the map and for the absence of collisions; second, IMU pre-integration is used to confirm that the motion constraints are consistent. Pose validation completes the relocalization process by publishing the pose to the output layer.

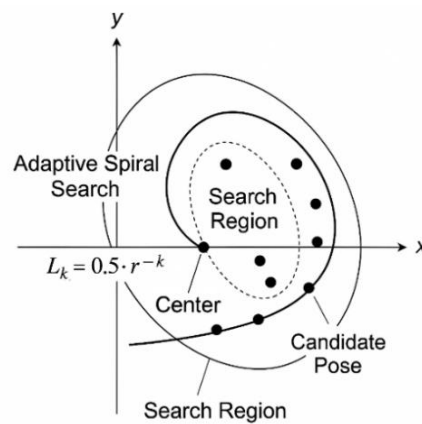


Figure 3. Expansion map of search scope

3. RESULTS AND DISCUSSION

This section explains the results of research and at the same time is given comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily [14], [15]. The discussion can be made in several sub-sections.

To establish the technical foundation for subsequent evaluation, we first analyze the core perception module's performance. This initial validation focuses on real-time dynamic obstacle detection using LiDAR point cloud processing, and robust feature extraction for environmental characterization. The verified performance of these subsystems directly enables the relocalization capabilities demonstrated in later benchmark and field tests.

3.1. LiDAR-based dynamic obstacle detection and feature extraction

The system is implemented on a real robotic platform equipped with an RPLIDAR A2 LiDAR sensor (model: RPLIDAR A2, 10 Hz scanning frequency, 16 m maximum range, 0.15 m minimum range, 0.0032 rad angular resolution, and 0.0858 s scan duration), which is used for 2D environmental scanning. The experimental setup includes dynamic human subjects walking at different speeds. The system operates on Ubuntu 20.04 with the robot operating system (ROS), and the resulting 2D scans are visualized in rviz, as illustrated in Figure 4.

Figure 4(a) shows the processed LiDAR data after dynamic obstacle detection and feature extraction, where the red point cloud represents the detected dynamic obstacles and the green point cloud corresponds to the extracted static features. Compared to the raw LiDAR data shown in Figure 4(b), which is visualized in yellow, the processed data is significantly reduced in volume. In particular, the number of points is reduced by approximately 50% to 70%, indicating a substantial decrease in redundant or non-essential data. Despite this reduction, key environmental features such as corners and arcs are largely preserved, ensuring that the structural integrity of the scene is maintained for relocalization.

The experimental results demonstrate that the proposed method effectively filters dynamic elements while preserving essential geometric features of the environment. This not only improves data efficiency but also enhances the robustness and accuracy of subsequent relocalization modules.

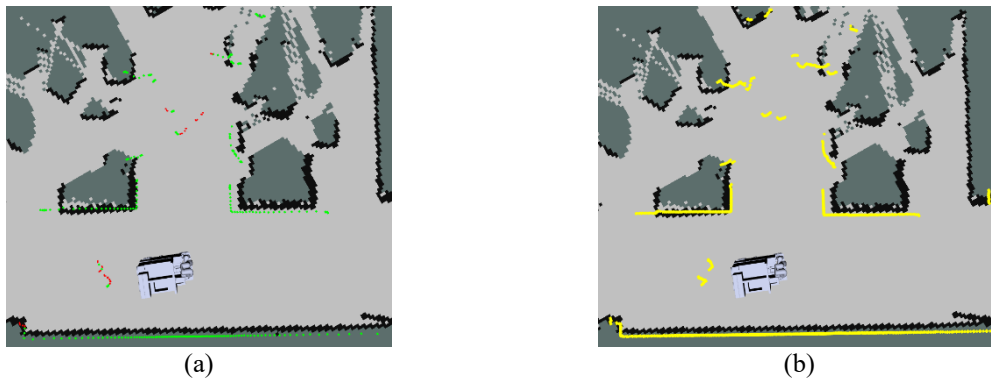


Figure 4. LiDAR data comparison: (a) LiDAR data after dynamic obstacle detection and feature extraction and (b) raw LiDAR data

3.2. Benchmark testing on OpenLORIS-Scene Dataset

This study uses the OpenLORIS-Scene public dataset to test the relocalization system's localization accuracy in indoor environments. The dataset was acquired by mobile robots in real-world environments, which included ground truth trajectories from motion capture devices or high-precision LiDAR as well as multimodal sensor data from a variety of settings, such as cafes, corridors, and offices.

Figure 5 illustrates the selection of four data sequences from two distinct scenarios in the OpenLORIS-Scene dataset used for evaluation. Specifically, Figure 5(a) and Figure 5(b) correspond to the Cafe1-1 and Cafe1-2 sequences, which capture dynamic indoor environments with frequent human activity. In contrast, Figure 5(c) and Figure 5(d) show the Corridor1-1 and Corridor1-2 sequences, which present challenges such as strong glass reflections and illumination changes. These representative sequences highlight the diverse environmental conditions under which our relocalization method is tested. The experimental platform consisted of an upper computer based on an Intel Core i5 processor running Ubuntu 20.04 with the ROS integrated into it. An offline sparse semantic map of the target environment was generated using ORB-SLAM3, running in Monocular-Inertial mode with a custom configuration file adapted to the dataset. This map included feature points, keyframes, and covisibility graphs. Both our algorithm and AMCL utilize the high-precision grid maps created with Lidar Inertial Odometry via Smoothing and Mapping (LIO-SAM) as ground truth references. Each dataset had 20 experimental trials for testing each algorithm, and the relocalization error was defined as (12) and (13).

$$P_{err} = \sqrt{(\hat{x} - x)^2 + (\hat{y} - y)^2} \quad (12)$$

$$\theta_{err} = |\hat{\theta} - \theta| \times \frac{180}{\pi} \quad (13)$$

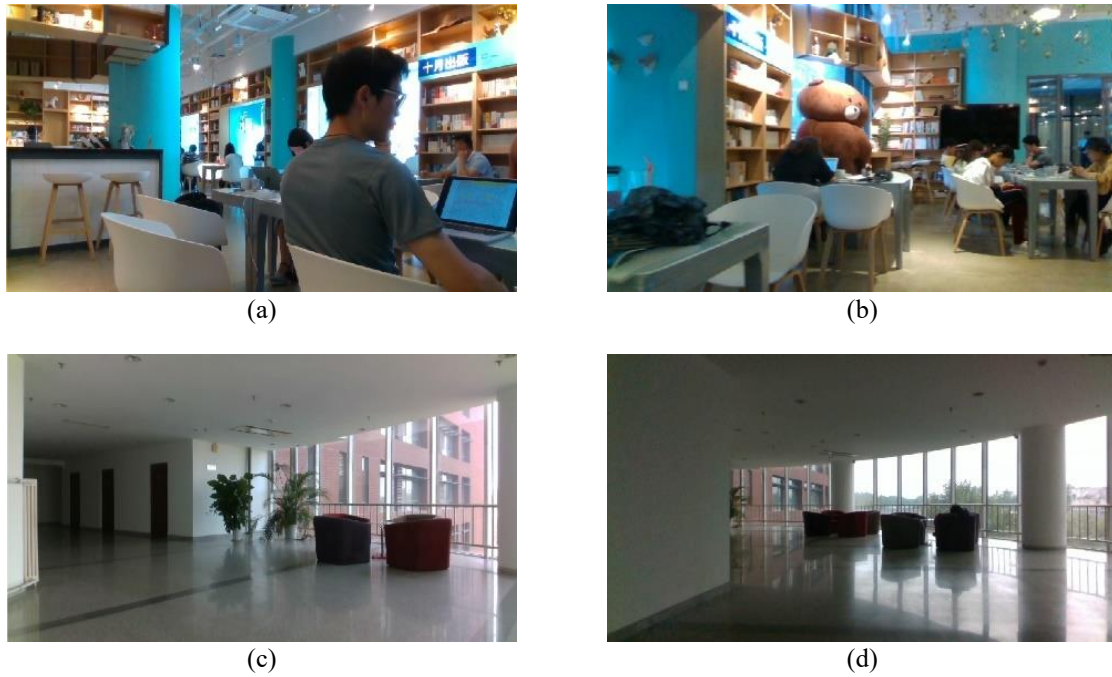


Figure 5. OpenLORIS-Scene dataset environments: (a) Cafe1-1 series, (b) Cafe1-2 series, (c) Corridor1-1 series, and (d) Corridor1-2 series

Figure 6 shows box plots of relocalization experiment results comparing different algorithms across multiple evaluation metrics. A box-plot is a statistical graph for describing the discrete degree of a group of data. The stability of the relocalization can be reflected by the box-plot. The horizontal line inside the box of a box plot represents the average value. As shown in Figure 6(a), the box plots of our algorithm demonstrate a relatively lower overall position (average error: 0.146 m) and smaller data dispersion. According to comparative analysis, our approach outperforms the LiDAR-dependent AMCL algorithm (0.321 m average error) by 54.7% in terms of overall positional error while achieving positioning accuracy on par with ORB-SLAM3 (0.150 m average error). But as Figure 6(b) illustrates, our algorithm's angular error noticeably rises when compared to ORB-SLAM3, especially in corridor datasets. The main causes of this performance degradation are (1) the LiDAR system's 180° scanning range in the OpenLORIS-Scene dataset and (2) the glass surfaces that are common in corridor environments, which significantly reduce the number of detectable LiDAR feature points, consequently increasing the dispersion of angular measurements.

Further analysis incorporating Figure 6(c) and Figure 6(d) reveals that our algorithm achieves an average CPU utilization of just 0.40 units, representing an 84.7% reduction compared to ORB-SLAM3 (2.60 units) and a 58.9% improvement over AMCL (0.97 units). In terms of processing time, our algorithm completes relocalization in 1.30 seconds on average; only 15% slower than ORB-SLAM3's 1.13 seconds while being 74.4% faster than AMCL's 5.08-second runtime. The lower position and compact dispersion of its box plots clearly demonstrate that the proposed method consumes fewer computational resources while maintaining stable performance during relocalization processes. These characteristics confirm the algorithm's higher computational efficiency and faster relocalization capability.

3.3. Field validation in real-world environments

To verify the effectiveness and robustness of the proposed method, experiments were conducted in an actual room environment using a comprehensive sensor suite. The real robot platform was equipped with an RPLIDAR A2 LiDAR sensor for 2D environmental scanning, a WHEELTEC N100 IMU for precise motion tracking and orientation estimation, and an upper computer based on Intel Core i5 processor. The system operated under the Ubuntu 20.04 environment with the ROS framework. For comparison, ORB-SLAM3 was run in Monocular-Inertial mode using a monocular camera and the WHEELTEC N100 IMU, with image-IMU synchronization achieved through timestamp alignment. Figure 7 presents the experimental setup, where Figure 7(a) shows the physical test environment, Figure 7(b) displays the corresponding test environment map, and Figure 7(c) illustrates the complete mobile robot platform with all integrated sensors.

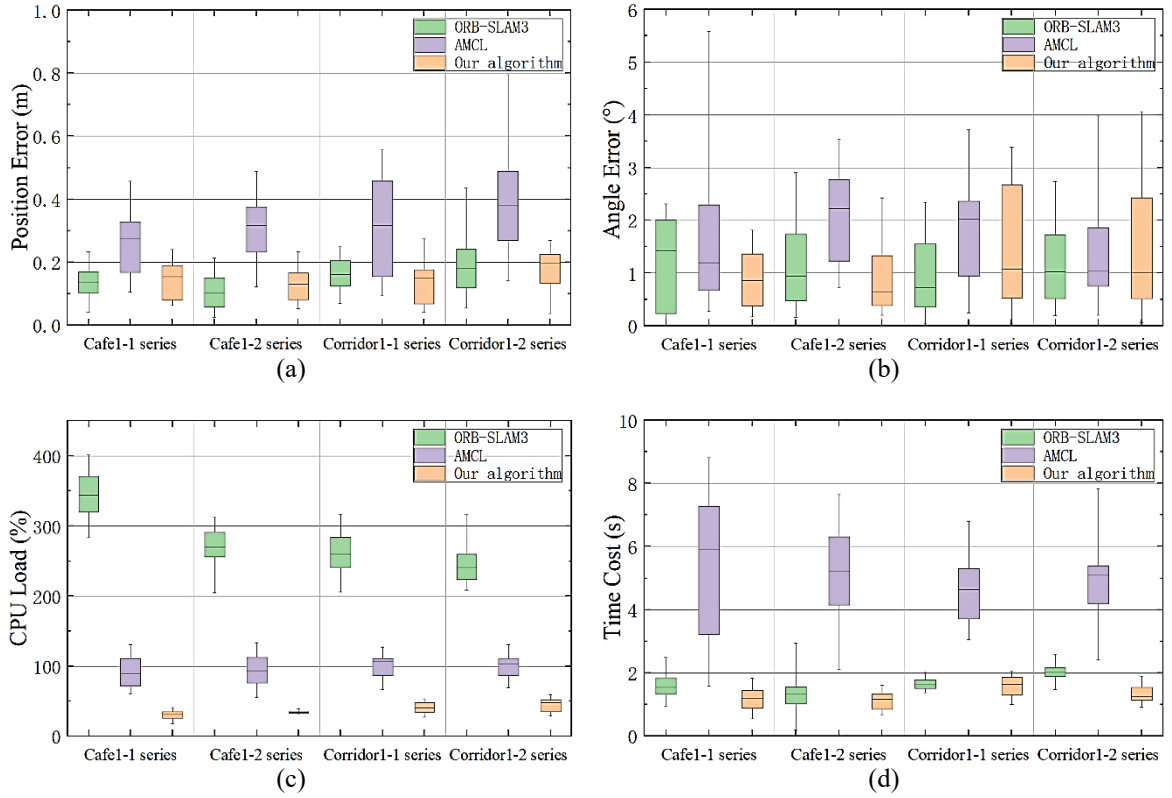


Figure 6. Relocalization experiment data of datasets-comparison of different algorithms: (a) position error, (b) angle error, (c) CPU load, and (d) time cost

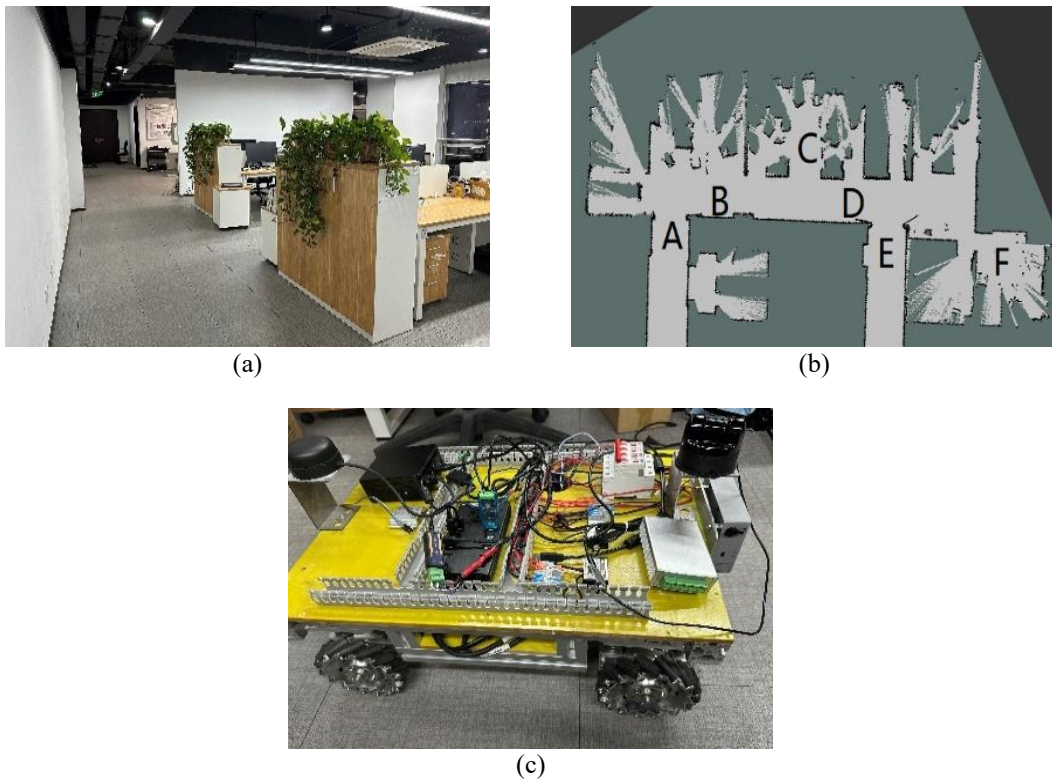


Figure 7. Relocalization test environment: (a) test environment, (b) test environment map, and (c) robot mobile

This study conducted a series of “robot kidnapping” tests to evaluate the performance of three algorithms: the proposed method, ORB-SLAM3, and AMCL. To enhance the comparative analysis under dynamic conditions, the experimental environment incorporated moving obstacles that accounted for approximately 15% of both LiDAR scans and camera frames. Additional challenging scenarios were introduced at specific locations: glass interference at Point C and reduced illumination at Point F. The experimental evaluation involved conducting comprehensive robot kidnapping tests across multiple scenarios (A→B, B→C, C→D, D→E, and E→F), with systematic performance comparisons measured through four key metrics: positioning accuracy (mean error and success rate documented in Table 1) and computational efficiency (processing time and CPU utilization presented in Table 2).

Table 1. Position error and success rate comparison by algorithm

Experimental points	Our algorithm		ORB-SLAM3		AMCL	
	Mean error	Success rate	Mean error	Success rate	Mean error	Success rate
A →B	0.073m, 0.58°	100%	0.074m, 0.57°	95%	0.210m, 9.25°	60%
B →C	0.137m, 0.70°	60%	0.108m, 0.69°	80%	0.394m, 11.32°	10%
C →D	0.109m, 0.72°	80%	0.112m, 0.62°	85%	0.224m, 10.49°	45%
D →E	0.113m, 0.92°	95%	0.090m, 0.76°	90%	0.262m, 1.56°	30%
E →F	0.101m, 0.61°	85%	0.137m, 1.10°	55%	0.527m, 1.31°	40%
Mean	0.106m, 0.65°	84%	0.104m, 0.74°	81%	0.323m, 6.78°	37%

Table 2. Execution time and CPU utilization comparison by algorithm

Experimental points	Our algorithm		ORB-SLAM3		AMCL	
	Mean time	CPU load	Mean time	CPU load	Mean time	CPU load
A →B	0.68s	35.4%	2.82s	278.2%	4.55s	100.6%
B →C	2.67s	37.3%	2.17s	265.7%	7.58s	99.6%
C →D	1.74s	39.9%	1.93s	246.4%	4.62s	98.4%
D →E	1.18s	41.9%	2.20s	277.4%	4.48s	101.7%
E →F	2.15s	37.6%	2.66s	264.1%	5.15s	88.9%
Mean	1.68s	38.4%	2.34s	266.3%	5.27s	97.9%

The proposed algorithm demonstrates superior relocalization accuracy due to IMU-based heading constraints, achieving an 89.6% reduction in mean yaw error compared to AMCL. Furthermore, the overall success rate is improved by 47% relative to AMCL. The conventional AMCL approach is prone to premature particle convergence, where particles erroneously concentrate in locally optimal—but globally incorrect—regions, leading to suboptimal pose estimation.

In Scenario C with glass interference, the proposed method maintains a positional error of merely 0.109 m with a 60% success rate, whereas AMCL exhibits significantly higher errors and a drastically reduced success rate of 10%, rendering it nearly ineffective. These results underscore the critical role of the Gaussian 3σ constraint modeling for historical poses in enhancing robustness.

While ORB-SLAM3 achieves an 80% success rate in Scenario C, its performance significantly degrades under low-light conditions (Scenario F), exhibiting a 26.9% increase in positional error and a 31.3% decline in success rate (from 80% to 55%). In contrast, the proposed algorithm maintains stable performance in Scenario F, demonstrating its illumination-invariant reliability. This consistent performance effectively highlights the inherent advantage of LiDAR’s insensitivity to lighting variations.

The computational efficiency evaluation demonstrates that the proposed method achieves significant improvements across all metrics compared to baseline algorithms. Relative to AMCL (97.9% CPU utilization), the proposed solution shows a 68.1% improvement in localization speed (1.68 s vs 5.27 s) while reducing CPU usage by 60.8% (38.4% vs 97.9%). When compared to ORB-SLAM3 (266.3% CPU utilization), it maintains a 28.5% speed advantage with an 85.6% reduction in computational load (38.4% vs 266.3%). Although processing demands increase in feature-dense scenarios (C and F) due to extensive feature extraction and long-range relocalization requirements, the algorithm’s optimized LiDAR processing pipeline and efficient spiral search strategy enable it to consistently outperform AMCL in both accuracy and resource utilization. These results validate the system’s ability to maintain high computational efficiency while delivering robust performance across diverse operating conditions.

The experimental results demonstrate the effectiveness of the proposed IMU-LiDAR likelihood field matching framework, which achieves three key performance benchmarks: i) an 84% success rate in robot kidnapping recovery scenarios, ii) an average relocalization time of 1.68 seconds, and iii) a CPU utilization rate of only 38.4%. These quantitative metrics collectively verify that the proposed method significantly improves both localization accuracy and system robustness in challenging real-world operating conditions, while maintaining exceptional computational efficiency.

4. CONCLUSION

Resolving the initial pose estimation and “robot kidnapping” issues is fundamental to enabling autonomous robot navigation, a critical capability for applications in industrial automation, service robotics, and autonomous driving. This study proposes a Robot Gaussian-Historical Relocalization algorithm based on IMU-LiDAR likelihood field matching, achieving significant improvements in localization accuracy, robustness, and computational efficiency. By leveraging IMU heading constraints and historical pose priors, it effectively reduces yaw error and increases relocalization success rates, particularly in challenging scenarios with dynamic obstacles or poor lighting. Compared to AMCL and ORB-SLAM3, the method not only offers faster and more reliable relocalization but also operates with substantially lower CPU usage. These results validate the algorithm’s suitability for real-world mobile robot applications, especially in environments where traditional vision- or particle-based methods struggle. Nevertheless, the method faces challenges in environments with weak geometric features, repetitive structures, or severe sensor interference such as fog, dust, or reflections.

Future work will first address the limitations caused by sensor noise and failure. We plan to conduct a detailed sensitivity analysis on the impact of errors in historical poses and sensor measurements, and explore adaptive uncertainty modeling to enhance robustness under challenging conditions. In parallel, we will explore machine learning-based methods such as CNNs and Transformers to enhance LiDAR feature extraction in ambiguous environments. Once a more robust single-robot system is established, we will extend the method to multi-robot scenarios, enabling shared relocalization for better scalability and coordination.

ACKNOWLEDGMENTS

The authors would like to thank the technical staff at Professor Jia-Qiang Yang’s laboratory, located in Room 1803 of the Hanggang Metallurgical Technology Building, for their assistance in setting up the experimental environment and data collection.

FUNDING INFORMATION

Authors state there is no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Ye-Ming Shen	✓	✓	✓	✓	✓	✓		✓	✓	✓				✓
Min Kang						✓		✓	✓	✓	✓	✓		
Jia-Qiang Yang		✓	✓	✓		✓			✓		✓		✓	✓
Zhong-Hou Cai	✓				✓		✓			✓		✓		

C : **C**onceptualization

M : **M**ethodology

So : **S**oftware

Va : **V**alidation

Fo : **F**ormal analysis

I : **I**nvestigation

R : **R**esources

D : **D**ata Curation

O : Writing - **O**riginal Draft

E : Writing - Review & **E**ditng

Vi : **V**isualization

Su : **S**upervision

P : **P**roject administration

Fu : **F**unding acquisition

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

INFORMED CONSENT

We have obtained informed consent from all individuals included in this study.

ETHICAL APPROVAL

The research related to human use has been complied with all the relevant national regulations and institutional policies in accordance with the tenets of the Helsinki Declaration and has been approved by the authors' institutional review board or equivalent committee.





DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author, Ye-Ming Shen, upon reasonable request.





REFERENCES

- [1] G. Wang, B. Shi, L. Wang, D. Song, and X. Jia, "Robot relocalization combining residual network and iterative closest point," *Journal of Xi'an Polytechnic University*, vol. 38, no. 4, pp. 18–25, 2024, doi: 10.13338/j.issn.1674-649x.2024.04.003.
- [2] H. Ma, W. Yan, and L. Yang, "Research on depth vision based mobile robot autonomous navigation in underground coal mine," *Journal of China Coal Society*, vol. 45, no. 6, pp. 2193–2206, 2020, doi: 10.13225/j.cnki.jccs.ZN20.0214.
- [3] J. D. Peña-Narvaez, F. Martín, J. M. Guerrero, and R. Pérez-Rodríguez, "A visual questioning answering approach to enhance robot localization in indoor environments," *Frontiers in Neurobotics*, vol. 17, Nov. 2023, doi: 10.3389/fnbot.2023.1290584.
- [4] Z. Zheng, K. Su, S. Lin, Z. Fu, and C. Yang, "Development of vision-based SLAM: from traditional methods to multimodal fusion," *Robotic Intelligence and Automation*, vol. 44, no. 4, pp. 529–548, Jul. 2024, doi: 10.1108/RIA-10-2023-0142.
- [5] J. Bao, S. Yang, R. Zhu, H. Tang, and A. Song, "Indoor mobile robot relocalization method combining vision and laser," *China Meas. Test*, vol. 47, no. 11, pp. 1–7, 2021, doi: 10.11857/j.issn.1674-5124.2021060197.
- [6] G. Ge, Y. Zhang, W. Wang, Q. Jiang, L. Hu, and Y. Wang, "Text-MCL: Autonomous mobile robot localization in similar environment using text-level semantic information," *Machines*, vol. 10, no. 3, p. 169, Feb. 2022, doi: 10.3390/machines10030169.
- [7] C.-L. Zhou, J.-D. Chen, and F. Huang, "WiFi-PDR fusion indoor positioning technology based on unscented particle filter," *Computer Science*, vol. 49, no. 6A, pp. 606–611, 2022, doi: 10.11896/j.sj.kx.210700108.
- [8] L. Jiang, C. Xiang, J.-Y. Zhu, and Q. Liu, "Particle filter relocation with semantic likelihood estimation," *Acta Electronica Sinica*, vol. 49, no. 2, pp. 306–314, 2021, doi: 10.12263/DZXB.20200396.
- [9] Q. Ye, M. Zheng, and X. Qiu, "Implementation of an indoor autonomous navigation mobile robot system based on ROS," *Transducer Microsyst. Technol.*, vol. 41, no. 2, pp. 90–93, 2022, doi: 10.13873/J.1000-9787(2022)02-0090-04.
- [10] J. Shou, Z. Zhang, Y. Su, and Z. Zhong, "Design and implementation of an indoor mobile robot localization and navigation system based on ROS and LiDAR," *Machine and Electronics*, vol. 36, no. 11, pp. 76–80, 2018, doi: 10.3969/j.issn.1001-2257.2018.11.018.
- [11] Z. Hu, J. Liu, G. Huang, and Q. Tao, "Robot indoor localization fusing WiFi, LiDAR, and maps," *J. Electron. Inf. Technol.*, vol. 43, no. 8, pp. 2308–2316, 2021, doi: 10.11999/JEIT200671.
- [12] Z. Wang, B. Yan, M. Dong, J. Wang, and P. Sun, "Wall-climbing robot localization method based on LiDAR and improved AMCL," *Chin. J. Sci. Instrum.*, vol. 43, no. 12, pp. 220–227, 2022, doi: 10.19650/j.cnki.cjsi.J2210261.
- [13] F. Lan, R. Liu, L. Guo, T. Deng, Z. Deng, and Y. Xiao, "Mobile robot localization fusing UWB azimuth and distance," *J. Electron. Meas. Instrum.*, vol. 37, no. 8, pp. 155–163, 2023, doi: 10.13382/j.jemi.B2306261.
- [14] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, "On-manifold preintegration for real-time visual-inertial odometry," *IEEE Transactions on Robotics*, vol. 33, no. 1, pp. 1–21, Feb. 2017, doi: 10.1109/TRO.2016.2597321.
- [15] C. Chen and Y. Ding, "Laser radar image edge feature extraction method based on median filtering," *Laser J.*, vol. 44, no. 8, pp. 94–98, 2023, doi: 10.14016/j.cnki.jgzz.2023.08.094.
- [16] Z. Gao, S. Wang, T. Miao, and A. Song, "Indoor laser SLAM system based on ground constraints and principal component analysis feature extraction," *Manned Spacefl.*, vol. 30, no. 2, pp. 150–159, 2024, doi: 10.16329/j.cnki.zrht.2024.02.010.
- [17] X. Xu, R. Yang, and N. Wang, "A robust circle detector with regionalized radius aid," *Pattern Recognition*, vol. 149, p. 110256, May 2024, doi: 10.1016/j.patcog.2024.110256.
- [18] J. Xue, P. Wang, J. Zhou, and F. Cheng, "Accurate construction of orchard two-dimensional environmental map based on improved Gmapping algorithm," *Transactions of the Chinese Society for Agricultural Machinery*, vol. 54, no. 7, pp. 26–34, 2023, doi: 10.6041/j.issn.1000-1298.2023.07.003.
- [19] Y. Lyu, L. Hua, J. Wu, X. Liang, and C. Zhao, "Robust radar inertial odometry in dynamic 3D environments," *Drones*, vol. 8, no. 5, p. 197, May 2024, doi: 10.3390/drones8050197.
- [20] R. Zheng, G. Sun, and F. D. Li, "A fusion localization system for security robots based on millimeter wave radar and inertial sensors," *Sensors*, vol. 24, no. 23, p. 7551, Nov. 2024, doi: 10.3390/s24237551.
- [21] X. Chi *et al.*, "A laser data compensation algorithm based on indoor depth map enhancement," *Electronics*, vol. 12, no. 12, p. 2716, Jun. 2023, doi: 10.3390/electronics12122716.
- [22] N. Adurthi, "Scan matching-based particle filter for LIDAR-only localization," *Sensors*, vol. 23, no. 8, p. 4010, Apr. 2023, doi: 10.3390/s23084010.
- [23] H. Wang, Y. Yin, and Q. Jing, "Comparative analysis of 3D LiDAR scan-matching methods for state estimation of autonomous surface vessel," *Journal of Marine Science and Engineering*, vol. 11, no. 4, p. 840, Apr. 2023, doi: 10.3390/jmse11040840.
- [24] K. Rajathi, N. Gomathi, M. Mahdal, and R. Guras, "Path segmentation from point cloud data for autonomous navigation," *Applied Sciences*, vol. 13, no. 6, p. 3977, Mar. 2023, doi: 10.3390/app13063977.
- [25] Y. Li, Y. Wei, Y. Wang, Y. Lin, W. Shen, and W. Jiang, "False detections revising algorithm for millimeter wave radar SLAM in tunnel," *Remote Sensing*, vol. 15, no. 1, p. 277, Jan. 2023, doi: 10.3390/rs15010277.





BIOGRAPHIES OF AUTHORS

Ye-Ming Shen     is a master's student in Mechanical Engineering at Zhejiang University of Science and Technology. His research focuses on intelligent robots, specifically exploring aspects such as robotic motion control and sensing technologies. He is committed to enhancing the intelligence and autonomy of robots to better adapt to complex tasks and environments. He can be contacted at 222307855043@zust.edu.cn.







Min Kang     is a professor at Zhejiang University of Science and Technology. He holds a doctorate (post-doctorate) and is a postgraduate supervisor, recognized as a third-level talent in Zhejiang Province. He graduated from Zhejiang University's Electrical Engineering Department. His research focuses on permanent magnet motor control, multi-phase motors, and motor simulation analysis and optimization design. He can be contacted at kangmin@zust.edu.cn.



Jia-Qiang Yang     is a professor at Zhejiang University. He holds a doctorate and is a doctoral supervisor. He is engaged in research on electric vehicle motor drive systems, new-energy power generation and energy-storage technology, automotive intelligent electronic systems and control, robot vision sensing and unmanned driving control, and electric energy quality control. He can be contacted at yjq1998@163.com; yangjiaq@zju.edu.cn.



Zhong-Hou Cai     is a Ph.D. student in the College of Electrical Engineering at Zhejiang University. His research is dedicated to the field of intelligent robots, delving into areas like robot navigation, path planning, perception, and decision-making algorithms. He aims to enhance the performance and reliability of intelligent robots to effectively meet the challenges presented by various real-world scenarios. He can be contacted at 12310032@zju.edu.cn.