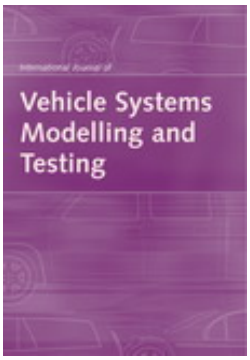


International Journal of Vehicle Systems Modelling and Testing

ISSN online: 1745-6444 - ISSN print: 1745-6436
https://www.inderscience.com/ijvsmt

Enhanced SLAM based on 2D LiDAR and RGB-D camera fusion for mobile robots navigation

Jie Yu, Peng-Hui Fu, Qing-Yong Zhang, Xiao-Lei Yan, Sheng Ye

DOI: 10.1504/IJVSMT.2025.10069362

Article History:

Received:	24 October 2024
Last revised:	16 December 2024
Accepted:	25 December 2024
Published online:	04 April 2025

Enhanced SLAM based on 2D LiDAR and RGB-D camera fusion for mobile robots navigation

Jie Yu, Peng-Hui Fu, Qing-Yong Zhang, Xiao-Lei Yan* and Sheng Ye

Fujian Key Laboratory of Automotive Electronics and Electric Drive,
Fujian University of Technology,
Fuzhou, 350118, China
Email: yujie@fjut.edu.cn
Email: 1016162140@qq.com
Email: 24486416@qq.com
Email: yanxiaolei@fjut.edu.cn
Email: 9021147@qq.com
*Corresponding author

Abstract: Real-time mapping and dynamic navigation for mobile robots present significant challenges, particularly due to the limitations of 2D LiDAR in environmental representation and the constraints of using a single RGB-D camera. This paper introduces a novel mapping method that enhances the traditional ORB-SLAM2 system by integrating 2D LiDAR and RGB-D camera data using Bayesian estimation. This approach enables the construction of dense maps, OctoMaps, and grid maps, improving the completeness and practicality of the mapping process. Additionally, Cartographer-SLAM is incorporated into the enhanced ORB-SLAM2 framework to further refine mapping capabilities. Comparative tests using the publicly available TUM dataset show that the proposed method reduces absolute pose error by 51.16%, with mapping trajectories closely aligning with ground truth values. The camera tracking trajectory improves by 16.2%. Experimental results demonstrate that the novel algorithm provides clearer environmental representations, increased accuracy, and higher mapping success rates.

Keywords: SLAM; simultaneous localisation and mapping; RGB-D camera; information fusion; Bayesian estimation.

Reference to this paper should be made as follows: Yu, J., Fu, P-H., Zhang, Q-Y., Yan, X-L. and Ye, S. (2025) 'Enhanced SLAM based on 2D LiDAR and RGB-D camera fusion for mobile robots navigation', *Int. J. Vehicle Systems Modelling and Testing*, Vol. 19, No. 1, pp.28–46.

Biographical notes: Jie Yu received his PhD degree in Automotive Engineering from the Fuzhou University, China, in 2020. He is currently an Associate Professor at the School of Mechanical and Automotive Engineering, Fujian University of Technology, China. His research interests include new energy and SLAM technology.

Peng-Hui Fu is a graduate student at Fujian University of Technology.

Qing-Yong Zhang is currently a Researcher at the Fujian Key Laboratory of Automotive Electronics and Electric Drive, China.

Xiao-Lei Yan is currently a Professor at the School of Mechanical and Automotive Engineering, Fujian University of Technology, China. His research interests are automotive optimisation design and intelligent vehicle navigation.

Sheng Ye is currently a Researcher at the Fujian Key Laboratory of Automotive Electronics and Electric Drive, China.

1 Introduction

Simultaneous localisation and mapping (SLAM) is the process by which a mobile robot utilises onboard sensors to gather environmental data in an unknown environment for real-time localisation and mapping (Cadena et al., 2016; Tsintotas et al., 2022). With the rapid advancement of SLAM technology (Arshad and Kim, 2021), it has found extensive applications in mobile robotics (Nguyen et al., 2022), virtual/augmented reality (VR/AR) (Detmer et al., 2017), drones, and autonomous driving (Zhang et al., 2021; Dao et al., 2024). In recent years, sensors such as LiDAR and cameras have become widely used in SLAM systems, with LiDAR-based SLAM achieving increasing maturity (Liu et al., 2023). In 2D LiDAR SLAM research (Yu et al., 2022), the laser SLAM was evaluated and tested using five single-line LiDARs (Grisetti et al., 2007), with results showing that Gmapping-SLAM (Ratul et al., 2021) and Karto-SLAM (Xu et al., 2022) were more efficient in positioning and mapping. Google later open-sourced Cartographer (Dwijotomo, 2020), which uses mainstream graph optimisation in its laser SLAM framework and introduced a branch-and-bound approach to address submap construction and global map matching challenges (Nguyen et al., 2022), further maturing the technology (Castellanos et al., 2018). Despite its high positioning accuracy and strong anti-interference capabilities, LiDAR has limitations, such as point cloud distortion and misalignment during operation (Aryan et al., 2021). Moreover, 2D LiDAR only scans a specific plane, making it unable to detect obstacles above or below that plane, resulting in unclear feature descriptions (Cheng et al., 2022).

Cameras and LiDAR sensors have been extensively applied in motion estimation and environment mapping (Kim and Park, 2019). Among these, 2D LiDAR provides advantages such as high positioning accuracy and strong anti-interference capabilities (Toschi et al., 2018). However, during operation, challenges such as point cloud distortion and mismatching arise. Additionally, 2D LiDAR scans a single plane, making it unable to detect obstacles above or below that plane, leading to incomplete feature descriptions. To address these limitations, the use of 3D LiDAR or combining 2D LiDAR with cameras has become increasingly popular in SLAM research.

Among various types of visual sensors, monocular cameras and RGB-D cameras are commonly used in mobile robotics (Marinov et al., 2023; Forster et al., 2016). Visual SLAM can provide rich environmental information (Taketomi et al., 2017), and several classic visual SLAM algorithms have been developed, including Mono SLAM (Klein and Murray, 2007), a kind of Mono SLAM based on EKF method, which plays an important role in the initial stage of visual SLAM, although it has some problems such as poor robustness and large computing load. PTAM (Engel et al., 2014) improves the problem that Mono SLAM cannot work stably in large-scale scenarios for a long time and adopts

nonlinear optimisation method in the back end for the first time. Raul Mur-Artal and his team added a loopback detection module on the basis of PTAM's dual-threading, and used ORB descriptors to detect feature points in images for positioning and mapping, and successively proposed ORB-SLAM (Murgartal et al., 2015) and ORB-SLAM2 (Mur-Artal and Tardós, 2017a). Campos came up with ORB-SLAM3 (Campos et al., 2021), a visual SLAM algorithm that supports a variety of cameras, and he optimised all stages to achieve excellent performance in terms of operational efficiency and composition accuracy. In addition, some of the other classic visual SLAMs include LSD-SLAM (Engel et al., 2014) and VI-ORB-SLAM2 (Mur-Artal and Tardós, 2017b). VI-ORB-SLAM2 is a noteworthy tightly-coupled visual-inertial (VI) system, developed based on ORB-SLAM2, DynaVIS (Song et al., 2022), and VINS-mono (Rosinol et al., 2020).

However, the ORB-SLAM series algorithms especially rely on environmental features, which is also a common problem of many feature point SLAM methods. It is difficult to detect enough feature points for extraction in low-texture scenes (corridors, etc.), thus reducing the robustness and accuracy of the system.

Given the limitations of single-sensor systems, multi-sensor fusion has emerged as a prominent research area (Liu et al., 2023). Existing multi-sensor fusion methods primarily include Kalman filtering (Li et al., 2023) and Bayesian estimation (Nam et al., 2024). Kalman filtering is mainly used to fuse low-level real-time dynamic multi-sensor redundant data. In the actual operation of the algorithm, when a single Kalman filter performs data statistics on a multi-sensor combined system, there is still a high failure rate of a single sensor, which reduces the reliability of the system. Bayesian estimation is used to describe the relationship between two conditional probabilities and provides a means for data fusion. It is a common method for fusing high-level information of multiple sensors in a static environment. The advantages of convolutional neural networks in feature extraction and classification can improve the above problems and bring more research space to closed-loop detection methods based on deep learning.

This paper proposes a loosely-coupled fusion strategy that integrates 2D LiDAR and RGB-D camera data to achieve robust motion estimation and construct 2D grid maps. This approach balances cost, efficiency, and mapping quality by utilising visual information to enhance LiDAR scan matching and incorporating LiDAR data to refine visual feature extraction. The main contributions of this work are summarised as follows:

- A loosely-coupled fusion strategy integrating 2D LiDAR and RGB-D camera measurements is proposed to enhance SLAM with greater accuracy and robust motion estimation.
- A multi-stage closed-loop detection strategy is introduced, which integrates visual and LiDAR data to accurately identify closed-loop candidates.
- An improved ORB-SLAM2 system is proposed, enabling the generation of dense maps, OctoMaps, and raster maps.

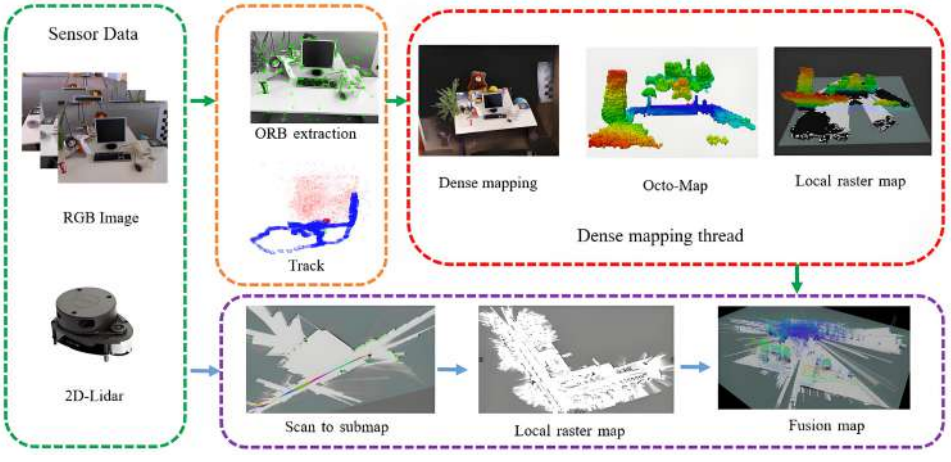
The structure of this paper is as follows: Section 2 describes the framework of Fusion-SLAM. Section 3 introduces the Fusion-SLAM mapping. Section 4 presents the experimental results and analysis. Finally, Section 5 concludes the paper by summarising the findings and outlining future research directions.

2 System overview and approach

2.1 Framework of Fusion-SLAM

Under these research backgrounds, this paper proposes a new SLAM method based on Bayesian estimation, which integrates the improved ORB-SLAM2 with the Cartographer-SLAM (Yu et al., 2022), this paper will name it Fusion-SLAM, as shown in Figure 1.

Figure 1 Fusion-SLAM based on the fusion of 2D LiDAR and RGB-D camera (see online version for colours)



The novel method effectively utilises the respective advantages of RGB-D camera and 2D LiDAR to accurately estimate transformations and construct environmental maps. This fusion strategy can limit motion in large and complex scenes, ensuring the robustness of realistic scale visual tracking. In addition, the integrated map produced by the novel algorithm can be more suitable for navigation of mobile robots.

2.2 Improved ORB-SLAM2 framework

ORB-SLAM2 is a feature-based visual SLAM algorithm. The traditional ORB-SLAM2 framework comprises three key components: the tracking thread, local mapping thread, and loop closing thread. This structure produces a sparse point cloud map, which is insufficient for direct use in robot navigation and path planning. To address this limitation, this paper introduces three additional threads – dense mapping, OctoMap, and grid mapping – into the original ORB-SLAM2 framework, as illustrated in Figure 2. These enhancements allow for the creation of more detailed maps, improving the system's suitability for navigation and path planning.

The improved ORB-SLAM2 framework is composed of the following components:

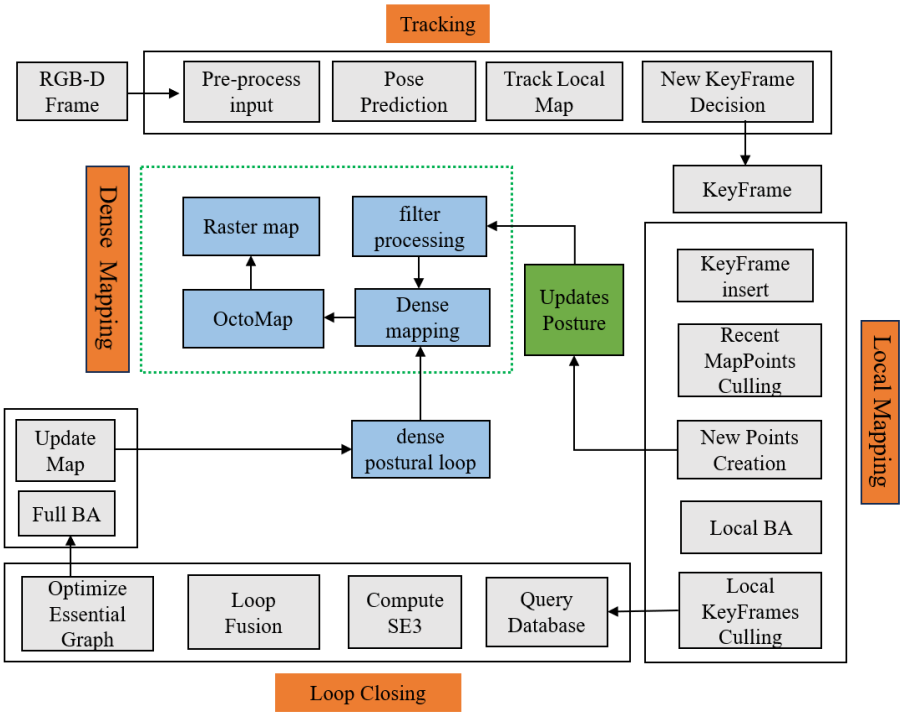
1 *Tracking*: The tracking thread, also known as visual odometry, aims to estimate the pose of the RGB-D camera and determine when to insert new keyframes. This process involves extracting features from the current image, matching them with nearby images, and calculating the camera pose based on these matching results.

2 *Local mapping*: The local mapping thread processes keyframes generated during tracking, inserts them into the map, and establishes relationships between the local map and point cloud matching. By filtering out point clouds with invalid judgements and low observation rates, the quality of successful matching points is improved. Local bundle adjustment (BA) is then used to optimise keyframes and remove redundant ones.

3 *Loop closing*: In this thread, keyframes are compared with neighbouring images, and an image recognition database is queried to eliminate low-score keyframes and remove connected ones. Afterward, the pose of the current frame is optimised based on similar transformations, followed by a bundle adjustment to refine the pose of all keyframes.

4 *Dense mapping*: This thread constructs a 3D dense point cloud from the RGB-D camera's keyframes, which were processed by the local mapping thread. This generates a dense environmental representation to enhance the detail and usability of the map.

Figure 2 Improved ORB-SLAM2 framework (see online version for colours)



Any point in space in the world coordinate system is defined as $P_w = [x_w, y_w, z_w]$. By traversing the RGB-D images and depth images corresponding to keyframes and extracting pixel coordinates (*u, v*) and spatial depth value *d*, the three-dimensional spatial point information in the world coordinate system can be obtained as follows:

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K\left(RP_w + t\right) \tag{1}$$

where, K denotes the camera internal reference, obtained from the camera calibration; R and t denote the rotation and translation matrices of the camera; and λ denotes the scale factor of the spatial value and the actual spatial distance.

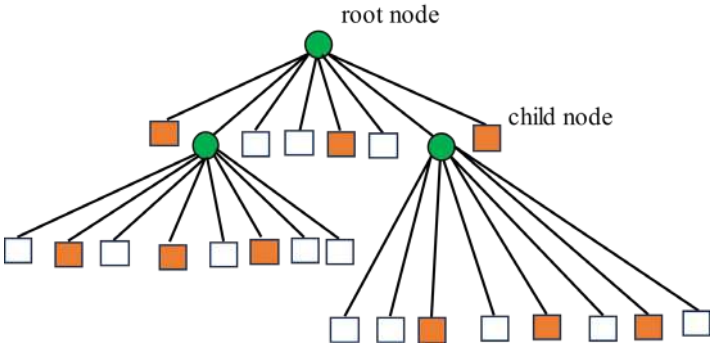
The origin of the world coordinate system coincides with the origin of the camera coordinate system, that is, there is no translation and rotation. At this moment, the 3D spatial position coordinates of the point cloud are obtained through equation (1):

$$\begin{cases} x_{\mathrm{w}} = \dfrac{\lambda\left(u - q_x\right)}{f_x} \\ y_{\mathrm{w}} = \dfrac{\lambda\left(u - q_y\right)}{f_y} \\ \lambda_{\mathrm{w}} = \lambda \end{cases} \tag{2}$$

where f_x, f_y, q_x, q_y represents the built-in parameters of the camera, f_x and f_y are the focal lengths in the horizontal and vertical directions, and q_x and q_{xy} are the pixel coordinates of the principal point position. Through equations (1) and (2), the depth information of the image is calculated to realise the construction of the 3D dense mapping.

5 *OctoMap*: The aim of this thread is to transform a dense point cloud map into an OctoMap (Li et al., 2020), as depicted in Figure 3. An octree map can be presented as a large cube evenly divided into eight blocks, having 8 child nodes beneath its root node. White indicates unoccupied space, purple indicates occupied and indivisible space; each node is analogous to a voxel. Unoccupied nodes keep expanding into eight child nodes, while occupied nodes cease to expand. When information is added to the map, as blank spaces are connected, the majority of octree nodes do not need to expand to the leaf level, thereby saving a considerable amount of storage space compared to the point cloud map.

Figure 3 Principle of octree diagram (see online version for colours)



Let $y \in R$, the probability that x is between 0 and 1, then the probability that a node is occupied is described by the logit transform:

$$y = logit(x) = \ln\left(\frac{x}{1-x}\right) \tag{3}$$

Let a node be n. The probability value of a node from the starting moment to moment t is $(n|M_{1:t})$, then the value at moment t is:

$$L(n|M_{1:t}) = L(n|M_{1:t-1}) + L(n|M_t) \tag{4}$$

where $L(n|M_{1:t})$ is the logarithmic value of the probability of being occupied at moment t; $L(n|M_{1:t-1})$ is the logarithmic value of the probability of being occupied at moment t–1; and M is the observed data.

The probability value of the node at a given moment is calculated through equation (4). $\varepsilon_{\text{occupy}}$ is a predefined parameter to calculate the probability value of the node at a given moment.

$$L(n|M_{1:t+1}) = \begin{cases} \varepsilon_{\text{occupy}} \\ 0 \end{cases} \tag{5}$$

The probability values are obtained through equation (5) and combined with the depth information from the RGB-D camera to obtain the octree map.

6 *Raster map*: The function of this thread is to convert the octree map obtained above into a grid map in real-time. Since the mobile robot navigates and plans its paths in a plane, it requires certain two-dimensional information. By projecting the octree map in the Z-axis direction, a grid map can be obtained for further research on the navigation and path planning of mobile robots.

3 Fusion-SLAM mapping

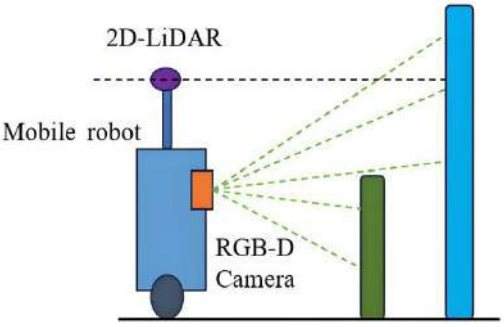
This section outlines the fusion rules and the framework for fusion mapping, detailing how the 2D LiDAR and RGB-D camera data are integrated to enhance Fusion-SLAM performance.

3.1 Fusion rules

During the mapping process in SLAM, the scanning range of the 2D LiDAR and the field of view of the RGB-D camera play a crucial role in determining mapping accuracy (Kolhatkar and Wagle, 2021). The 2D LiDAR offers a full 360 scanning range on a single plane, while the RGB-D camera has a more limited, acute-angled perception range. Figure 4 illustrates the differing perception ranges between the 2D LiDAR and the RGB-D camera. To maximise mapping accuracy, the fusion of data from these two sensors must consider their complementary capabilities. The 2D LiDAR provides

extensive horizontal coverage, capturing detailed structural information, while the RGB-D camera offers richer visual data, albeit with a narrower field of view. By integrating the two, the system can produce a more comprehensive and precise environmental map.

Figure 4 Schematic diagram of perception range of LiDAR and RGB-D camera (see online version for colours)



As shown in Figure 4, the 2D LiDAR collects environmental data within a single plane, but it cannot adequately capture obstacles above or below that plane, leading to gaps in environmental representation. On the other hand, the RGB-D camera provides rich external information, though its limited perspective can prevent it from accurately representing the full environment. In the process of local raster map construction, the local map data generated by 2D LiDAR and RGB-D camera are fused to generate high-precision local maps and realise real-time storage and update, and the fusion rules are shown in Table 1.

Table 1 Fusion rules

	2D LiDAR		
RGB-D Camera	*occupancy*	*unoccupied*	*unknown*
occupancy	occupancy	occupancy	occupancy
unoccupied	occupancy	unoccupied	unoccupied
unknown	occupancy	unoccupied	unknown

A loosely coupled approach is adopted in this study, where Bayesian estimation is used to fuse mapping. The Bayesian formula calculates the distribution function of conditional probabilities or posterior probabilities using prior distribution and likelihood function (Xu et al., 2022). Assuming the probability at time k in the same state is $P(x_k)$, and the measured values at this time are $T^k = (T_1, T_2, \ldots, T_k)$, the simplified prior distribution can be expressed as:

$$P(x_k|T^k) = \frac{P(T_k|x_k)\,P(x_k|T^{k-1})}{P(T^k|T^{k-1})} \tag{6}$$

where $P(T_k|x_k)$ denotes the likelihood function of the measurement model; $P(x_k|T^{k-1})$ denotes the prior distribution function at the (k –1) moment; and $P(T^k|T^{k-1})$ denotes the probability of the (k–1) moment estimate at the k moment.

By combining the observed data of the sensors with the Bayesian formula, the raster cell occupancy probability value of the sensor can be obtained as follows:

$$P^0 = \frac{P_s P_m}{P_s P_m + (1-P_s)(1-P_m)} \tag{7}$$

where P^0 denotes the updated probability value after data fusion; P_s denotes the conditional probability of occupying a raster cell after the point cloud is returned; P_m denotes the a priori probability of the raster being occupied; $(1-P_s)$ denotes the probability of the 2D LiDAR raster being empty; and $(1-P_m)$ denotes the probability of the RGB-D camera raster being empty.

The threshold is set in advance, when the grid cell occupancy probability P^0 is greater than the pre-designed threshold, the grid is in occupied state; when the grid cell occupancy probability P^0 is less than the pre-designed threshold, the grid is in idle state; then the grid occupancy probability after sensor fusion is shown as:

$$P_f^0 = \frac{P_1^0 P_2^0}{P_1^0 P_2^0 + (1-P_1^0)(1-P_2^0)} \tag{8}$$

where P_f^0 denotes the probability of occupying the raster after fusion; $(1-P_1^0)$ denotes the probability of not being occupied in the 2D LiDAR raster map; and $(1-P_2^0)$ denotes the probability of not being occupied in the RGB-D camera map.

3.2 Framework of fusion-SLAM mapping

Figure 5 illustrates the computational framework for the fusion mapping approach, which is divided into two main stages. In the first stage, Cartographer-SLAM is applied to the point cloud data obtained from the 2D LiDAR to construct a 2D grid map. Concurrently, the improved ORB-SLAM2 generates dense point cloud maps and octree maps from the data captured by the RGB-D camera, resulting in an initial grid map output. In the second stage, Bayesian estimation is employed to integrate the two constructed grid maps, producing a comprehensive grid map optimised for mobile robot navigation.

4 Experimentation and analysis

4.1 Experiment environment

The overall framework of the SLAM system includes a PC, WiFi router, Cartographer-SLAM, enhanced ORB-SLAM2, 2D LiDAR, RGB-D camera, and a mobile robot. The PC used for the experiment is equipped with an Intel Core i9-12900H processor, running on Ubuntu 20.04 with ROS Noetic. The sensors for the mobile robot consist of the RpLiDAR A1 LiDAR and the Orbbec Astra Mini RGB-D camera. The principal specifications of the RpLiDAR A1 are as follows: measurement radius of 16 m, sampling

frequency of 8 Hz, scanning frequency of 15 Hz, angular resolution of 0.9°, supply voltage of 5V, ranging resolution of ≤1% of the actual distance (for ranges ≤12 m), and range accuracy of 1% of the actual distance (for ranges ≤3 m). Key specifications of the Orbbec Astra Mini RGB-D camera include: depth resolution of 1280 × 1024 at 7 FPS, colour resolution of 1280 × 720 at 30 FPS, measuring range of 0.6–8 m, and field of view angles of H58.4° and V45.5°.

Figure 5 The computational framework for fusion-SLAM mapping method (see online version for colours)

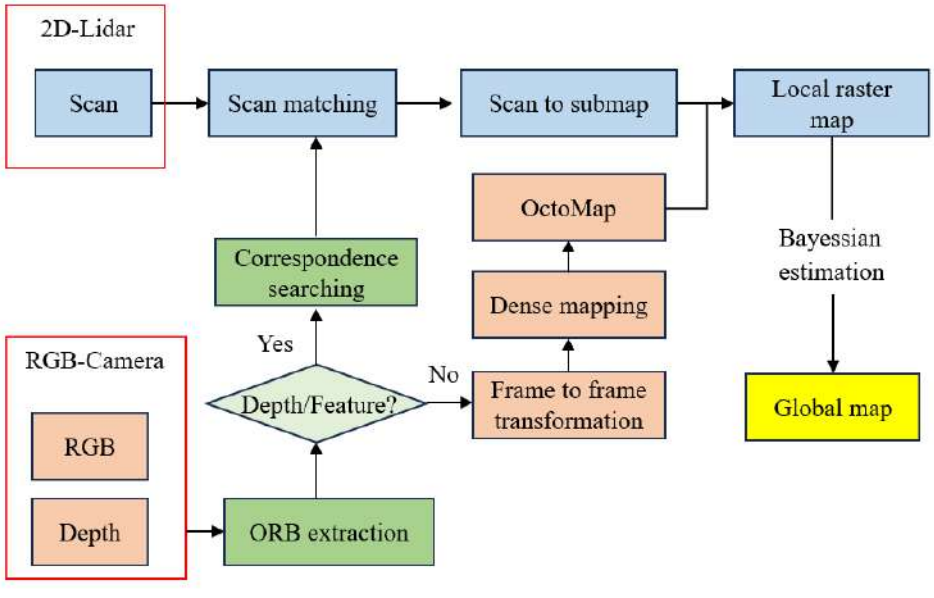
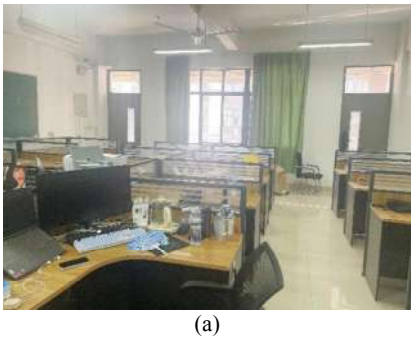
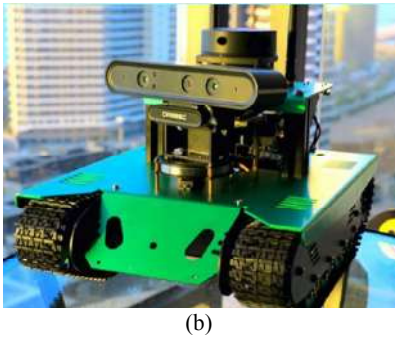


Figure 6 Mobile robot and experimental scene: (a) experimental scenario and (b) mobile robot (see online version for colours)



(a)



(b)

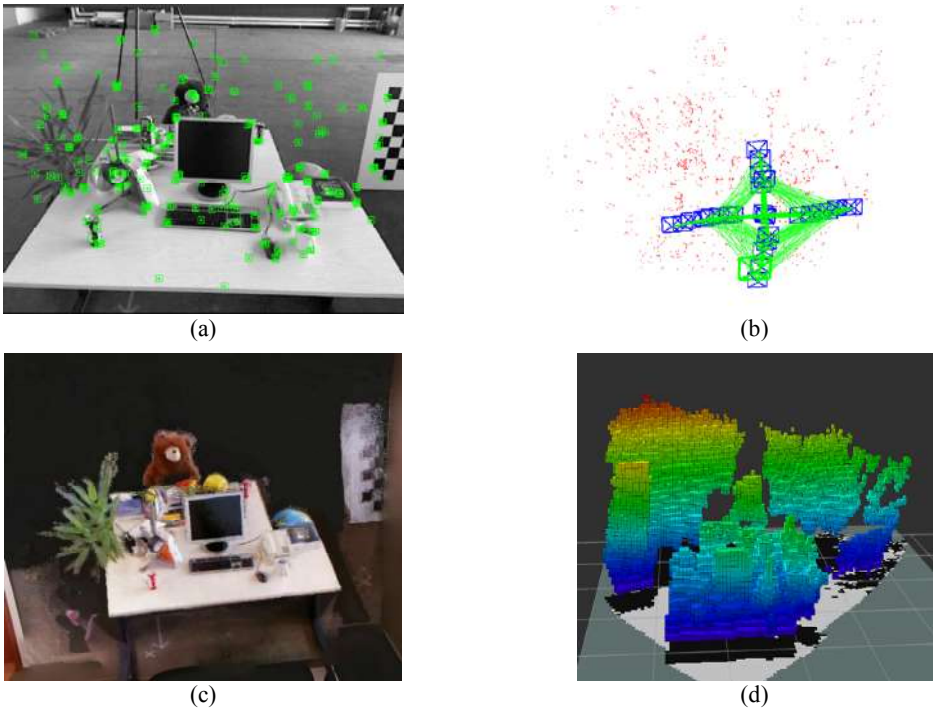
The experimental scene selected for this test is a medium-sized office, which includes a desk, chairs, and various other items, as shown in Figure 6(a). The mobile robot utilised in the experiment is depicted in Figure 6(b).

4.2 Mapping experiment for improved ORB-SLAM2

4.2.1 Experimental datasets and the EVO evaluation

To validate the improvements made to ORB-SLAM2 for mapping, this study employs the publicly available fr2_xyz dataset (Sturm et al., 2012) to assess the effectiveness of constructing dense maps and octree maps. The experimental results are illustrated in Figure 7.

Figure 7 The experimental results using fr2_xyz datasets: (a) feature extraction; (b) sparse mapping; (c) dense mapping and (d) OctoMap (see online version for colours)



(a) (b)

(c) (d)

To evaluate the performance of the improved ORB-SLAM2, the EVO evaluation tool is utilised to test both the absolute pose error (APE) and relative pose error (RPE) across various TUM datasets. APE measures the absolute pose difference between the ground truth trajectory and the estimated trajectory, providing an indication of the global consistency of the computed path (Kolhatkar and Wagle, 2021). In contrast, RPE assesses the local accuracy of the trajectory over specific time intervals, quantifying the difference between estimated and actual motion. This evaluation is crucial for assessing the performance of visual odometry systems, as shown in Tables 2 and 3.

Based on the above evaluation, the trajectory error of TUM datasets is evaluated, and the evaluation results are shown in Figure 8.

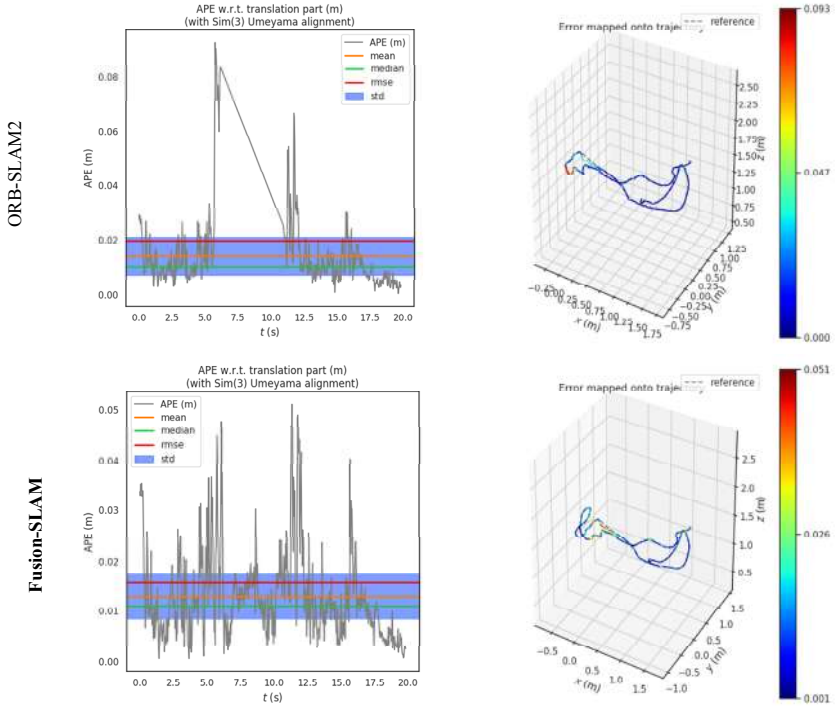
Table 2 Results of metrics absolute pose error (APE)

Sequences	*ORB-SLAM2*		*Fusion-SLAM*		*Improvements*	
	RMSE	*STD*	*RMSE*	*STD*	***RMSE***	***STD***
fr1_xyz	0.0898	0.0546	0.0106	0.0035	88.19%	93.58%
fr1_floor	0.1642	0.0736	0.0250	0.0102	84.77%	86.14%
fr1_room	0.0675	0.0315	0.0785	0.0389	–16.29%	–23.49%
fr2_desk	0.1025	0.0458	0.0256	0.0058	75.02%	65.5%
fr2_rpy	0.0252	0.0425	0.0158	0.0358	37.3%	18.7%

Table 3 Results of metrics relative pose error (RPE)

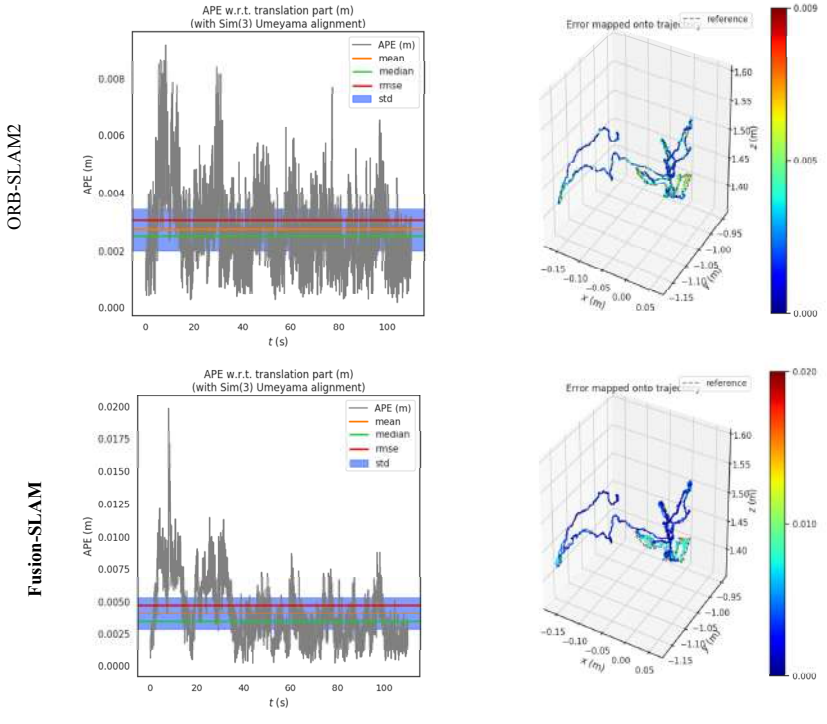
Sequences	*ORB-SLAM2*		*Fusion-SLAM*		*Improvements*	
	RMSE	*STD*	*RMSE*	*STD*	***RMSE***	***STD***
fr1_xyz	0.0608	0.0347	0.0061	0.0034	89.96%	90.20%
fr1_floor	0.0437	0.0286	0.0181	0.0177	58.58%	38.11%
fr1_room	0.1196	0.0856	0.1363	0.0952	–13.88%	–11.2%
fr2_desk	0.0458	0.0256	0.0158	0.0185	65.55%	38.37%
fr2_rpy	0.0635	0.0958	0.0725	0.1085	48.81%	49.37%

Figure 8 Absolute pose error plots for different datasets: (a) *fr1_desk* and (b) *fr1_rpy* (see online version for colours)



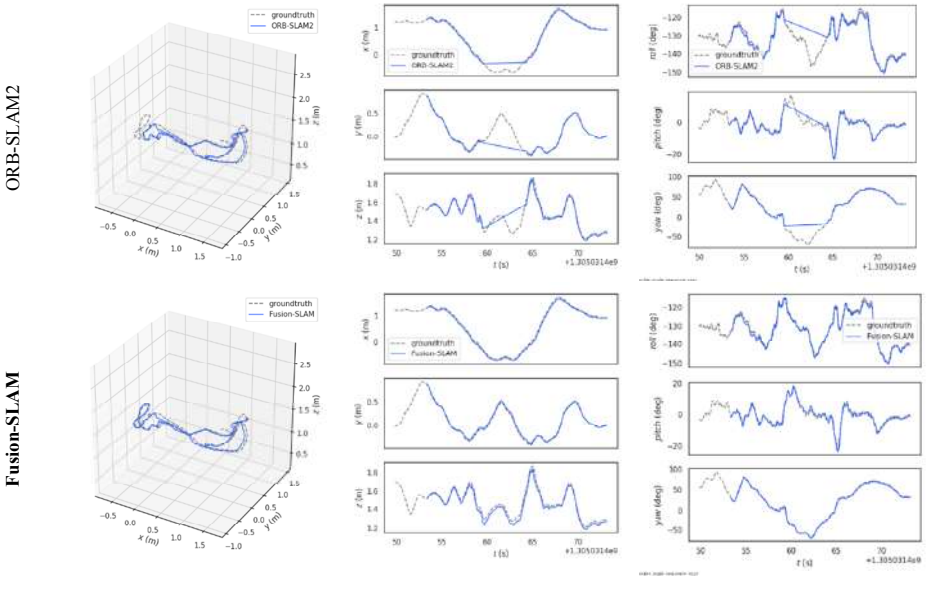
(a)

Figure 8 Absolute pose error plots for different datasets: (a) *fr1_desk* and (b) *fr1_rpy* (see online version for colours) (continued)



(b)

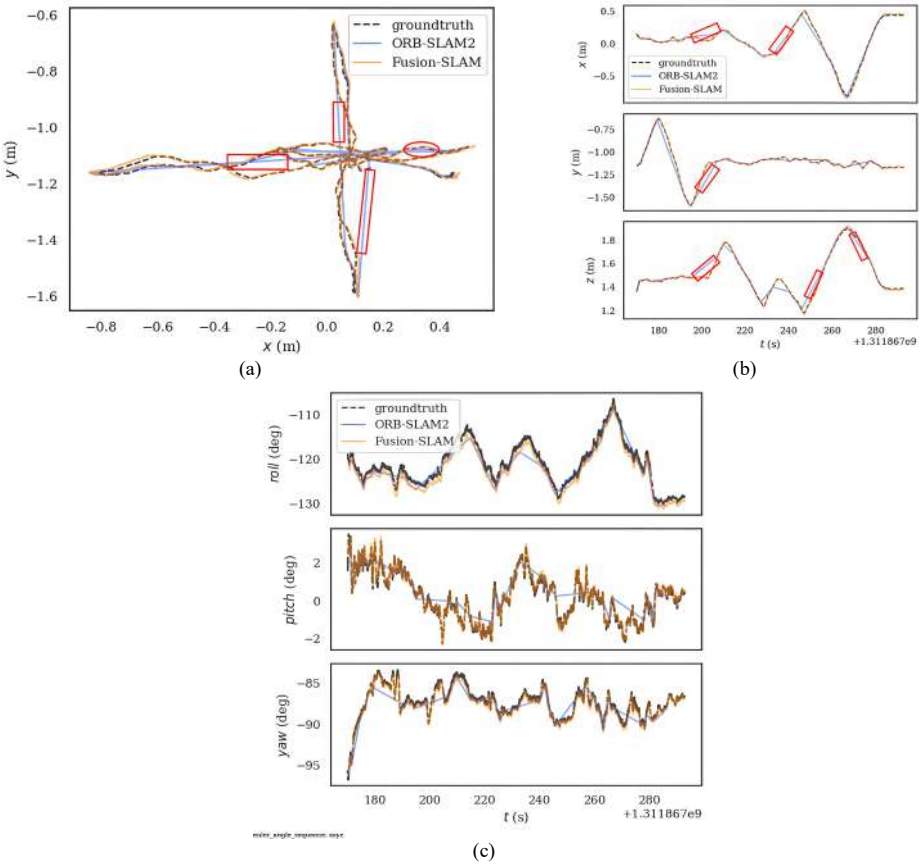
Figure 9 Trajectory comparison diagram (see online version for colours)



In Figure 8, the dashed portion represents the true trajectory value, the coloured section constitutes the estimated trajectory of the system, and the rightmost ribbon indicates the absolute position error value. It can be observed that the absolute position error of ORB-SLAM2 amounts to 0.073 m, while that of the improved ORB-SLAM2 is 0.032 m, achieving an improvement of 51.16% in the absolute position error. Moreover, the improved algorithm presents a lower trajectory error value and higher map accuracy. The comparison of trajectories of the two methods is shown in Figure 9.

To convey the accuracy comparison of the two methods more intuitively, a bar chart is employed to represent the fr1_desk dataset, as depicted in Figure 10.

Figure 10 Comparison of track error data: (a) trajectory comparison; (b) *X*, *Y*, *Z* plane trajectories and (c) comparison of errors (see online version for colours)



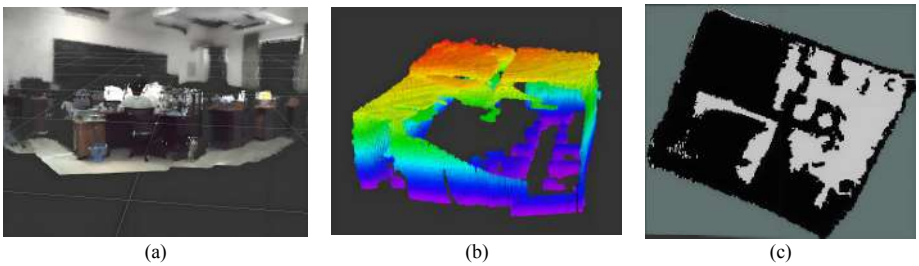
In Figure 10(a), the dashed part indicates the real environment values. The blue line represents the trajectory of the improved ORB-SLAM2 proposed in this paper, and the yellow line represents the trajectory of the original ORB-SLAM2. This clearly demonstrates that the method presented in this paper is closer to the real trajectory and has higher global consistency. Figure 10(b) presents the trajectory map in the *X*, *Y*, and *Z* planes. Figure 10(c) depicts the comparison of the median error, median error, standard deviation, maximum and minimum values of the algorithms in the two cases.

The number of feature points has a significant influence on the construction of the map. Therefore, the Linux system is employed to test the length of the camera trajectory of the two algorithms. According to the terminal display, the camera trajectory length of ORB-SLAM2 is 3.065m, and the camera trajectory length of the improved ORB-SLAM2 is 3.656m. The camera tracking trajectory increases by 16.2%, which indicates that the improved ORB-SLAM2 can perform better tracking in the case of fewer feature points.

4.2.2 Real environmental assessment

To verify the map building effect of the improved algorithm in the real scene, the experimental scene in Figure 6(a) is selected for the construction of dense point cloud maps to improve ORB-SLAM2 and realise real-time transformation of the octree maps, and ultimately obtain the raster maps that can be used for the navigation of mobile robots. The experimental results are shown in Figure 11.

Figure 11 Improved physical construction of ORB-SLAM2: (a) dense mapping; (b) OctoMap and (c) grid map (see online version for colours)



(a) (b) (c)

In most practical applications of visual SLAM, such as autonomous robots, driverless cars, and augmented reality, real-time performance is an important metric for evaluating SLAM systems. Therefore, fr1_xyz of the TUM dataset are used to test the trace time, as depicted in Table 4.

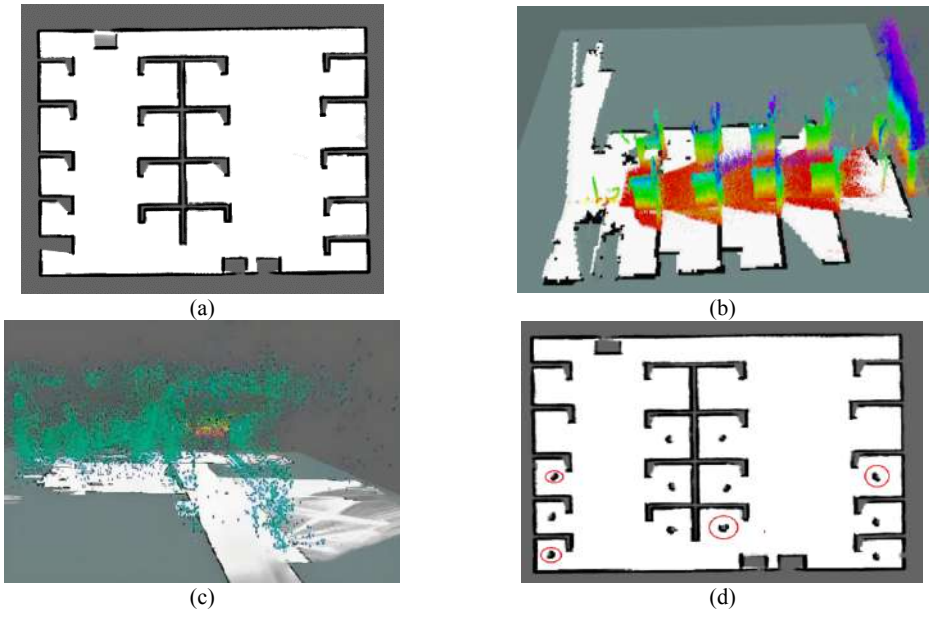
Table 4 Comparison of tracking time between the two methods

Time	*ORB-SLAM2*	*Fusion-SLAM*
Mean/(ms.frame^{-1})	65.69	66.38
Median/(ms.frame^{-1})	66.78	67.25

4.3 Mapping experiment for fusion-SLAM method

To validate the feasibility of the fusion of 2D LiDAR and RGB-D camera for map construction, the experimental scene is depicted in Figure 6(a), where the mobile robot is equipped with a RpLiDAR A1 single-line LiDAR and the Cartographer-SLAM is employed to construct the raster map. The result of the mapping based on fusion approach is presented in Figure 12.

Figure 12 The results of fusion-SLAM mapping process: (a) Cartographer-SLAM; (b) mapping effect; (c) angle mapping effect and (d) fusion grid map (see online version for colours)



As can be observed from Figure 12(a), the 2D LiDAR is incapable of identifying obstacles that are not in the same plane, such as chairs, etc. Hence, it is necessary to supplement the perspective of the RGB-D camera to acquire more complete environmental information. Therefore, this paper employs the fusion mapping method of Bayesian fusion to achieve the complementary advantages of the two. To better exert the advantages of RGB-D cameras, the indoor environment is selected for fusion mapping to avoid its influence on the mapping effect due to lighting reasons. The mapping effect is shown in Figure 12(b) and (c), and the fused grid map shown in Figure 12(d) is ultimately obtained.

The experimental results demonstrate that the detection of obstacles in the fused raster map is more comprehensive, the representation of obstacles lower than the mobile robot is more sufficient, and the detection at the edge points is more precise, which enhances the accuracy of map building for the mobile robot.

The Cartographer-SLAM and ORB-SLAM2 are tested in the experimental scenario in Figure 5(a) for several times of map building respectively, and the building time and the success rate of map building are recorded respectively, and the experimental results are compared with the fusion building maps, and the data table of map building as shown in Table 5 is obtained.

Table 5 Comparison of the different methods

Method	*Average running time* (s)	*Successful rate* (%)
Cartographer-SLAM	269	86
ORB-SLAM2	334	78
Proposed method	315	94

As can be seen from Table 5, in this paper, due to the fusion of two kinds of point cloud data from LiDAR and RGB-D camera, the construction time is longer than the single LiDAR construction time, but the success rate of the construction has been greatly improved, and the accuracy of the construction has been improved. For the method proposed in this paper, there is still some room for improvement, i.e., although the improved ORB-SLAM2 has some improvement over the original algorithm, there are still some problems in the process of feature extraction and matching, and the algorithm's accuracy can be improved by optimising the feature matching subsequently.

5 Conclusion

In this paper, aiming at the shortcomings of a single sensor, the traditional ORB-SLAM2 algorithm is enhanced to make it have the functions of dense map and octree map, and the improved method is integrated with the Bayesian estimation method of 2D LiDAR to realise the complementary advantages of the two, and the constructed raster map can be directly used in the navigation of mobile robots, improving the application range.

Evaluation on a public dataset demonstrated that the improved ORB-SLAM2 achieved a notable increase in accuracy, with a 51.16% reduction in absolute pose error and a 16.2% increase in camera tracking trajectory. Furthermore, the integration of the improved ORB-SLAM2 with Cartographer-SLAM resulted in a clearer representation of the environment and a higher success rate in map construction.

However, several challenges remain to be addressed in future work. The process of constructing dense maps requires significant memory resources, and there are certain inaccuracies associated with octree map projection. Additionally, further enhancements to the accuracy of the fused map are anticipated. Future research will focus on developing dynamic object detection capabilities and applying these techniques to robot navigation tasks, enabling more advanced functionalities in complex dynamic environments. This direction aims to enhance the intelligent navigation of mobile robots, allowing them to effectively navigate diverse dynamic scenarios.

Acknowledgements

This work was supported by the Research Foundation of Fujian University of Technology (GY-Z21025).

References

Arshad, S. and Kim, G-W. (2021) 'Role of deep learning in loop closure detection for visual and LiDAR SLAM: a survey', *Sensors*, Vol. 21, No. 4, p.1243.

Aryan, A., Bosché, F. and Tang, P. (2021) 'Planning for terrestrial laser scanning in construction: a review', *Automation in Construction*, Vol. 125, p.103551.

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I. and Leonard, J. (2016) 'Past, present, and future of simultaneous localization and mapping: toward the robust-perception age', *IEEE Transactions on Robotics*, Vol. 3, No. 6, pp.1309–1333.

Campos, C., Elvira, R., Rodríguez, J.J.G., Montiel, J.M.M. and Tardós, J.D. (2021) 'ORB-SLAM 3: an accurate open-source library for visual, visual–inertial, and multimap slam', *IEEE Transactions on Robotics*, Vol. 37, No. 6, pp.1874–1890.

Castellanos, J.A., Neira, J. and Tardós, J.D. (2018) 'Map building and SLAM algorithms', *Autonomous Mobile Robots*, CRC Press, pp.335–372.

Cheng, J., Zhang, L., Chen, Q., Hu, X. and Cai, J. (2022) 'A review of visual SLAM methods for autonomous driving vehicles', *Engineering Applications of Artificial Intelligence*, Vol. 114, p.104992.

Dao, T-K., Ngo, T-G., Pan, J-S., Nguyen, T-T-T. and Nguyen, T-T. (2024) 'Enhancing path planning capabilities of automated guided vehicles in dynamic environments: multi-objective PSO and dynamic-window approach', *Biomimetics*, Vol. 9, No. 1, doi: 10.3390/biomimetics 9010035.

Detmer, F.J., Hettig, J., Schindele, D., Schostak, M. and Hansen, C. (2017) 'Virtual and augmented reality systems for renal interventions: a systematic review', *IEEE Reviews in Biomedical Engineering*, Vol. 10, pp.78–94.

Engel, J., Schöps, T. and Cremers, D. (2014) 'LSD-SLAM: large-scale direct monocular SLAM', *European Conference on Computer Vision*, Springer, Cham, pp.834–849.

Forster, C., Zhang, Z., Gassner, M., Werlberger, M. and Scaramuzza, D. (2016) 'SVO: semidirect visual odometry for monocular and multicamera systems', *IEEE Transactions on Robotics*, Vol. 33, No. 2, pp.249–265.

Grisetti, G., Stachniss, C. and Burgard, W. (2007) 'Improved techniques for grid mapping with Rao-Blackwellized particle filters', *IEEE Transactions on Robotics*, Vol. 23, No. 1, pp.34–46.

Kim, E. and Park, S-Y. (2019) 'Extrinsic calibration between camera and LiDAR sensors by matching multiple 3D planes', *Sensors*, Vol. 20, No. 1, p.52.

Klein, G. and Murray, D. (2007) 'Parallel tracking and mapping for small AR workspaces', *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, Nara, Japan, pp.225–234.

Kolhatkar, C. and Wagle, K. (2021) 'Review of SLAM algorithms for indoor mobile robot with LIDAR and RGB-D camera technology', *Innovations in Electrical and Electronic Engineering: Proceedings of ICEEE 2020*, Springer, Singapore, pp.397–409.

Li, J., Zhang, X., Li, J., Liu, Y. and Wang, J. (2020) 'Building and optimization of 3D semantic map based on LiDAR and camera fusion', *Neurocomputing*, Vol. 409, pp.394–407.

Li, N., Zhou, F., Yao, K., Hu, X. and Wang, R (2023) 'Multisensor fusion SLAM research based on improved RBPF-SLAM algorithm', *Journal of Sensors*, doi: 10.1155/2023/3100646.

Liu, C., Liu, C., Zhang, G., Rong, Y., Shao, W., Meng, J., Li, G. and Huang Y. (2023) 'Hybrid metric-feature mapping based on camera and LiDAR sensor fusion', *Measurement*, Vol. 207, p.112411.

Liu, Q., Yang, H., Liu, T., Wu, T. and Lu, C. (2023) 'SLAM algorithm based on fusion of LiDAR and depth camera', *Transactions of the Chinese Society for Agricultural Machinery*, Vol. 54, No. 11, pp.29–38.

Marinov, E., Martins, R.J., Ben Youssef, M.A., Kyrou, C., Coulon, P-M. and Genevet, P. (2023) 'Overcoming the limitations of 3D sensors with wide field of view metasurface-enhanced scanning LiDAR', *Advanced Photonics*, Vol. 5, No. 4, p.460053.

Mur-Artal, R. and Tardós, J.D. (2017a) 'ORB-SLAM 2: an open-source SLAM system for monocular, stereo, and RGB-D cameras', *IEEE Transactions on Robotics*, Vol. 33, No. 5, pp.1255–1262.

Mur-Artal, R. and Tardós, J.D. (2017b) 'Visual-inertial monocular SLAM with map reuse', *IEEE Robot. Autom. Lett*, Vol. 2, No. 2, pp.796–803.

Murgartal, R., Montieljm, M. and Tardósjd, O.S. (2015) 'Aversatile and accurate monocular SLAM system', *IEEE Transactions on Robotics*, Vol. 31, No. 5, p.1147G1163.

Nam, D.V., Danh, P.T., Park, C.H. and Kim, G.W. (2024) 'Fusion consistency for industrial robot navigation: an integrated SLAM framework with multiple 2D LiDAR-visual-inertial sensors', *Computers and Electrical Engineering*, Vol. 120, p.109607.

Nguyen, T-T., Nguyen, T-D., Chu, S-C. and Dao, T-K. (2022) 'An improved ants colony optimization for mobile robot path planning', *Advances in Intelligent Information Hiding and Multimedia Signal Processing*, pp.239–249.

Nguyen, T-T., Nguyen, T-X-H., Do, V-C. and Ngo, T-G. (2022) 'An optimal path planning issue parameters with potential feasible area using evolutions Algorithm BT', *Intelligent Systems and Networks*, pp.394–401.

Ratul, M.D.T.A., Mahmud, M.S.A., Abidin, M.S.Z. and Ayop, R. (2021) 'Design and development of GMapping based SLAM algorithm in virtual agricultural environment', *2021 11th IEEE International Conference on Control System*, *Computing and Engineering* (*ICCSCE*), Penang, Malaysia, pp.109–113.

Rosinol, A., Abate, M., Chang, Y. and Carlone, L. (2020) 'Kimera: An open-source library for real-time metric-semantic localization and mapping', *2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, pp.1689–1696.

Song, S., Lim, H., Lee, A.J. and Myung, H. (2022) 'DynaVINS: a visual-inertial SLAM for dynamic environments', *IEEE Robotics and Automation Letters*, Vol. 7, No. 4, pp.11523–11530.

Sturm, J., Engelhard, N., Endres, F., Burgard, W. and Cremers, D. (2012) 'A benchmark for the evaluation of RGB-D SLAM systems', *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, pp.573–580.

Taketomi, T., Uchiyama, H. and Ikeda, S. (2017) 'Visual SLAM algorithms: a survey from 2010 to 2016', *IPSJ Transactions on Computer Vision and Applications*, Vol. 9, No. 1, pp.1–11.

Toschi, I., Remondino, F., Rothe, R. and Klimek, K. (2018) 'Combining airborne oblique camera and LiDAR sensors: investigation and new perspectives', *The International Archives of the Photogrammetry*, *Remote Sensing and Spatial Information Sciences*, Vol. 42, pp.437–444.

Tsintotas, K.A., Bampis, L. and Gasteratos, A. (2022) 'The revisiting problem in simultaneous localization and mapping: a survey on visual loop closure detection', *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 11, pp.19929–19953.

Xu, X., Zhang, L., Yang, J., Cao, C., Wang, W., Ran, Y., Tan, Z. and Luo, M. (2022) 'A review of multi-sensor fusion slam systems based on 3D LIDAR', *Remote Sensing*, Vol. 14, No. 12, p.2835.

Yu, J., Zhang, A., Zhong, Y., Nguyen, T-T. and Nguyen, T-D. (2022) 'An indoor mobile robot 2D LiDAR mapping based on cartographer-slam algorithm', *Journal of Network Intelligence*, Vol. 7, No. 3, pp.795–804.

Zhang, S., Zheng, L. and Tao, W. (2021) 'Survey and evaluation of RGB-D SLAM', *IEEE Access*, Vol. 9, pp.21367–21387.