

# Indoor Robot SLAM with Multi-Sensor Fusion

Jionglin He

School of Computer Science & Engineering  
Xi'an Technological University  
Xi'an, China  
E-mail: 1013598511@qq.com

Jiaxiang Fang

School of Computer Science & Engineering  
Xi'an Technological University  
Xi'an, China  
E-mail: 2293950221@qq.com

Shuping Xu

School of Computer Science & Engineering  
Xi'an Technological University  
Xi'an, China  
E-mail: 563937848@qq.com

Dingzhe Yang

School of Computer Science & Engineering  
Xi'an Technological University  
Xi'an, China  
E-mail: 1820522365@qq.com

**Abstract**—In order to solve the problem of large positioning error and incomplete mapping of SLAM based on two-dimensional lidar in indoor environment, a multi-sensor fusion SLAM algorithm for indoor robots was proposed. Aiming at the mismatch problem of the traditional ICP algorithm in the front end of the lidar SLAM, the algorithm adopts the PL-ICP algorithm that is more suitable for the indoor environment, and uses the extended Kalman filter to fuse the wheel odometer and IMU to provide the initial motion estimation value. Then, during the mapping phase, the pseudo 2D laser data converted from the 3D point cloud data obtained by the depth camera is fused with the data obtained from the 2D lidar to compensate for the lack of vertical field of view in the 2D lidar mapping. The final experimental results show that the fusion odometer data has improved the positioning accuracy by at least 33% compared to a single wheeled odometer, providing a higher initial iteration value for the PL-ICP algorithm. At the same time, fusion mapping compensates for the shortcomings of a single two-dimensional lidar mapping, and constructs an environmental map with more complete environmental information.

**Keywords**—SLAM; Indoor Robot; Extended Kalman Filter; PL-ICP

## I. INTRODUCTION

With various types of indoor mobile robots being widely used in human life and production, robot Simultaneous Localization and Mapping (SLAM) technology, as the basis for robots to complete service work, has gradually become a

research hotspot [1]. Affected by the diversity and complexity of the indoor environment, the information that can be obtained by the simultaneous localization and mapping of the common single radar is limited, so it cannot construct complete three-dimensional map information [2]. Therefore, multi-sensor fusion has become a new method to solve the defects of indoor robot SLAM. At present, the mainstream sensors applied to solve SLAM problem are lidar and vision camera, each of which has different advantages [3]. Among them, two-dimensional lidar has the advantages of small error and high reliability, but it has no vertical direction information and is easy to cause waste of resources. The depth camera has the advantages of a wide field of view and can identify specific objects, but the accuracy and stability of mapping are poor, and the probability of deviation is high [4-5]. At the moment, many scholars have proposed many fusions SLAM schemes based on the complementarity of perception sensors.

In terms of the fusion of vision and pose data Measurement sensors, literature [6] uses the coupling method to fuse the camera with the Inertial Measurement Unit (IMU) and the odometer, so as to obtain more accurate pose data. However, because the use is not tightly coupled, the vision sensor is prone to photosensitivity and instability. Literature [7] transfers the data

collected by the IMU as the initial pose to the vision sensor, which reduces the data pressure of the depth camera in the mapping to a certain extent, but it is prone to data disorder. If the data in the IMU unit is wrong, it will lead to problems in the vision part. In terms of wheeled odometer and laser sensor, literature [8] uses the Extended Kalman Filter (EKF) algorithm to allow the odometer to provide pose while fusing the point cloud data of lidar to make the obtained data more accurate and reliable. In literature [9], key frame data are identified by lidar and weighted and fused with odometer data by Iterative Closest Point (ICP) algorithm, so as to obtain more accurate pose information. However, these two algorithms will be too traditional due to the implementation method. It is easy to have problems such as low efficiency and poor data fusion effect. In the fusion of laser and attitude measurement sensor. Literature [10] tightly coupled the lidar and IMU fusion, which provided a higher frequency of real-time pose data for the robot, but there was no way to eliminate the invalid data resulting in data redundancy. Literature [11] tries to introduce odometer data to assist IMU for fusion, but there are still some errors. In the fusion of vision and laser sensor, literature [12] fuses laser and camera to realize loop closure detection and complete global map construction, but the stability is not good. Literature [13] uses the fusion of vision and laser based on static scenes to improve the accuracy of mapping, but it cannot achieve dynamic real-time performance. In literature [14], the overall fusion data of odometry, IMU, visual camera and lidar were analyzed, and the feasibility of multi-sensor fusion SLAM was verified, but there were problems such as accuracy decline and partial map missing.

Based on the above literature, it can be seen that SLAM with multi-sensor fusion can provide more accurate pose data, and can improve the accuracy of mapping and navigation. However, there are also many urgent problems to be solved, such as odometer slippage and map fusion defects. In this paper, a new fusion method is proposed, which uses two-dimensional lidar, depth camera, wheeled odometer and IMU unit to carry out multi-sensor fusion, and uses the improved Point-

to-line ICP (PL-ICP) algorithm to intercept key frames. To make up for the shortcomings of the common ICP algorithm in data interception and fusion. EFK algorithm is directly applied to IMU to ensure the simplification and effectiveness of data acquisition. Finally, the Bayesian method is used to fuse the data of vision and laser sensors, and the fusion two-dimensional raster map is constructed to make up for the shortcomings of single sensor mapping. This paper aims to improve the accuracy of indoor robot pose information, establish a global raster map with more complete information elements and more complete data, and prove the effectiveness of the fusion in the actual environment.

## II. INTER-FRAME MATCHING OF LASER POINT CLOUD

The position and orientation information of the mobile robot can be calculated by wheeled odometry or inertial odometry, but the odometry information obtained by these two methods has large errors. In fact, the laser odometry calculation method uses multiple sub-images to complete the map. In fact, it is impossible to insert all the point clouds identified in each frame into the submap, so the laser odometry only selects the key frame data for insertion, and the unimportant frame data will be discarded. The data of the traditional wheel odometer is used as the initial iteration value for the laser odometer to solve the robot's position and orientation information.

In the laser inter-frame matching solution, the Iterative Closest Point (ICP) algorithm can obtain a good matching effect by iterating the initial value without point cloud segmentation and feature extraction, so ICP algorithm has become one of the most studied and most mature algorithms [15].

### A. ICP Algorithm

The basic principle of ICP algorithm is as follows:

Let the set of spatial coordinates of the two-point cloud frames of the laser be: Starting

frame  $X = \{x_1, x_2, \dots, x_i\}$ ,  $x_i = \begin{bmatrix} d_i \cos \theta_i \\ d_i \sin \theta_i \end{bmatrix}$ . Target

frame  $P = \{p_1, p_2, \dots, p_j\}$ ,  $p_j = \begin{bmatrix} d_j \cos \theta_j \\ d_j \sin \theta_j \end{bmatrix}$ . Where  $d_i$ ,  $\theta_i$  and  $d_j$ ,  $\theta_j$  are the distance and the corresponding angle of the environmental information acquired by the two frames of lidar, respectively. Then, by finding  $k$  groups of corresponding points, the rotation  $R$  and translation  $t$  of the two laser frames can be solved. Finally, the error function  $E(R, t)$  is constructed and iterated continuously to make the error function result meet the set threshold, and the optimal  $R$  and  $t$  can be obtained.

The error function is as follows.

$$E(R, t) = \frac{1}{k} \sum_{n=1}^k \|x_n - (Rp_n + t)\|^2 \quad (1)$$

Where  $x_n$  and  $p_n$  denote a certain set of corresponding points in  $k$  groups, and then the mean value of the point cloud of the two frames is denoted by  $u_x$  and  $u_p$  respectively:

$$E(R, t) = \frac{1}{k} \sum_{n=1}^k (\|x_n - u_x - R(p_n - u_p)\|^2 + \|u_x - Ru_p - t\|^2) \quad (2)$$

For any  $R$  in the right term above, a  $t$  can be found to make the right term overall 0, so the left term above can be transformed into the maximum

value of  $\text{Trace} \sum_{n=1}^k R p_n' x_n'^T$ , Where  $x_i' = x_i - u_x$  and  $p_i' = p_i - u_p$  are two point clouds respectively subtracting their respective point cloud geometric center to form a new point cloud. Decentralizing the point clouds is equivalent to performing a translation, which shortens the distance between two point clouds. The purpose is to approximately convert two point clouds that may be in different coordinate systems to the same coordinate system, and also to prevent local optima. The SVD decomposition yields the following.

$$W = \sum_{n=1}^k p_n' x_n'^T = U \Sigma V^T \quad (3)$$

When  $W$  has full rank:

$$R = UV^T \quad (4)$$

$$t = u_x - Ru_p \quad (5)$$

In summary, each iteration of ICP algorithm will traverse every point in the origin set until it finds the closest point to the target point [16]. Therefore, the basic process of ICP algorithm is as follows: firstly, for the two laser point clouds that need to be matched, the nearest associated point of the point cloud is found. Usually, the initial corresponding point is obtained by using the data of the wheeled mileage meter. Then based on the corresponding points,  $R$  and  $t$  are further solved. After that, the point cloud is transformed, the matching error is calculated and whether the error meets the set threshold is judged. If it does, the  $R$  and  $t$  solved are output, if not, the iteration continues until the error meets the set threshold.

### B. PL-ICP Algorithm

In the ICP algorithm in the previous section, in the process of finding corresponding points, the point with the closest Euclidean distance is considered to be the corresponding point of the point cloud. However, in the actual indoor environment, the distance between the laser point and the actual environmental surface is the best error scale, so the standard ICP algorithm will cause a certain number of wrong corresponding points [17]. To solve this problem, many improved versions of ICP algorithm have been derived. The iterative closest point from point to line improves the error equation of the standard ICP algorithm, and uses the distance between the laser point and the line of the nearest two points of the laser point cloud in the next frame to approximate the real scale relationship, which is more suitable for indoor scenes [18].

Therefore, the error equation of PL-ICP algorithm is as follows.

$$J(R_{k+1}, t_{k+1}) = \sum_i \left( n_i^T \left[ R_{k+1} p_i + t_{k+1} - p_{j_i} \right] \right)^2 \quad (6)$$

Where  $p_i$  represents the  $i$ -th sampling point,  $p_{j_i}$  is the nearest matching point of the sampling point under the target point cloud,  $n_i$  is the normal vector of the two nearest matching points,  $R_{k+1}$ ,  $t_{k+1}$  represent the transformation parameters, and the optimal transformation parameters can be obtained by minimizing the objective function  $J$ .

Therefore, the basic process of PL-ICP algorithm is as follows: firstly, the initial rotation matrix is obtained according to the wheel odometer data, and then the current frame laser data is converted to the reference coordinate system. Then for each point of the current laser frame, the two closest points in the reference frame are found and the error is calculated. The error equation is constructed by removing the point with too large error. Finally,  $R$  and  $t$  are solved and the error is judged. If the set threshold is not satisfied, the  $R$  and  $t$  are used to go back to Step1 and continue to iterate. If they are satisfied, the output  $R$  and  $t$  are used to obtain the pose of the robot.

PL-ICP is an algorithm [18] with high matching accuracy and robustness. Compared with ICP algorithm, PL-ICP algorithm has higher solution accuracy and is more suitable for indoor environment [20]. Therefore, the PL-ICP algorithm will also be used in the laser front-end of this paper to solve the inter-frame matching of the laser. However, it is also relatively more sensitive to the initial value and requires a higher accuracy initial value. Although the wheel odometer has the advantages of high frequency and low negative environmental impact, it also has the problems of low test accuracy and the wheel is prone to deformation and slip, which leads to large errors in the test. If only the data of the wheel odometer is used as the initial value of matching, the algorithm is easy to fall into a local cycle due to the large cumulative error of the wheel odometer.

### III. SLAM WITH MULTI-SENSOR FUSION

#### A. Extended Kalman Filter Fusing Wheeled Odometry and IMU

Since the PL-ICP algorithm has higher initial value requirements than the ICP algorithm, and the wheel odometer will cause more and more cumulative errors with the movement of the robot and tire slip, if only the data of the wheel odometer is used as the initial value of the PL-ICP algorithm iteration, the PL-ICP algorithm may fall into a local loop. Although IMU has the problem of integral pose divergence, its instantaneous pose is accurate. Therefore, extended Kalman filter can be used to fuse wheel odometry and IMU to improve the accuracy of pose, and the fused pose data can be used as the initial value of PL-ICP algorithm to avoid falling into local circulation [21].

The extended Kalman filter realizes the linearization of the nonlinear function by performing the first-order Taylor expansion of the nonlinear function. The basic process is as follows:

The state equation is as follows.

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (7)$$

The observation equation is:

$$z_k = h(x_k, v_k) \quad (8)$$

Where  $x_k$  and  $u_{k-1}$  are the state variable and the control of the system respectively,  $w_{k-1}$  and  $v_k$  are the process noise and measurement noise and meet normal distribution, and 1 in the subscript  $k$  and  $k-1$  is the time unit, indicating the sampling time. Because of the state equation and observation equation is nonlinear, so a posteriori state estimation on the system of  $\hat{x}_{k-1}$  place for the first order Taylor expansion for linearization, the results are as follows:

$$x_k = f(\hat{x}_{k-1}, u_{k-1}, w_{k-1}) + A(x_k - \hat{x}_{k-1}) + Ww_{k-1} \quad (9)$$

Because of  $w_{k-1}$  error cannot be computed, so assume it is 0, the  $\tilde{x}_k = f(\hat{x}_{k-1}, u_{k-1}, 0)$ , in which  $A$  is the Jacobian of the partial derivative of the function  $f$  with respect to  $x$ ,  $A = \frac{\partial f}{\partial x} \Big|_{\hat{x}_{k-1}, u_{k-1}}$ ,  $W$  is the Jacobian of the partial derivative of the function  $f$  with respect to  $w$ ,  $W = \frac{\partial f}{\partial w} \Big|_{\hat{x}_{k-1}, u_{k-1}}$ , in this way:

$$x_k = \tilde{x}_k + A(x_k - \hat{x}_{k-1}) + Ww_{k-1} \quad (10)$$

The observation equation is  $\tilde{x}_k$  linearized at, and the result is as follows:

$$z_k = h(\tilde{x}_k, v_k) + H(x_k - \tilde{x}_k) + Vv_k \quad (11)$$

Because  $v_k$  for error cannot be computed, assuming it is 0, let  $\tilde{z}_k = h(\tilde{x}_k, 0)$ .  $H$  is the Jacobian of the partial derivative of the function  $h$  with respect to  $x$ ,  $H = \frac{\partial h}{\partial x} \Big|_{\tilde{x}_k}$ ,  $V$  is the Jacobian of the partial derivative of the function  $h$  with respect to  $v$ ,  $V = \frac{\partial h}{\partial v} \Big|_{\tilde{x}_k}$ ,  $P(Vv_k) \sim N(0, VRV^T)$ .

$$z_k = \tilde{z}_k + H(x_k - \tilde{x}_k) + Vv_k \quad (12)$$

So the prediction stage of the extended Kalman filter is as follows:

1) The prior state estimate is:

$$\tilde{x}_k^- = f(\hat{x}_{k-1}, u_{k-1}, 0) \quad (13)$$

2) The covariance matrix of the error is:

$$P_k^- = AP_{k-1}A^T + WQW^T \quad (14)$$

correction stage is as follows:

1) Kalman gain:

$$K_k = \frac{P_k^- H^T}{HP_k^- H^T + VRV^T} \quad (15)$$

2) Posterior state estimation results:

$$\tilde{x}_k = \tilde{x}_k^- + K_k(z_k - h(\tilde{x}_k^-, 0)) \quad (16)$$

3) Update the error covariance matrix:

$$P_k = (I - K_k H) P_k^- \quad (17)$$

Therefore, the fusion process of wheel odometer and IMU using extended Kalman filter is as follows: When the observation equation of the first sensor is updated, the state quantity and the covariance matrix of the system are obtained as the system predicted state quantity and the system predicted covariance matrix of the second sensor correction process, and then the updated system state estimation result and the covariance matrix are used as the fused output. And they are used in the prediction process for iteration at the next moment [22]. In order to ensure that the subsequent fusion experiment in the real environment can be carried out smoothly, the algorithm is first tested in the simulation software to verify whether the algorithm can correctly subscribe the information of IMU and odometer node and output the fused odometer data. Fig. 1 shows the simulation fusion calculation diagram of the algorithm under ROS:



The above experiments are simulation tests of the fusion algorithm. In order to further verify the effectiveness of the extended Kalman filter for the fusion of IMU and wheeled odometer data, the positioning accuracy of the robot is tested in the real scene as shown in Fig. 4. The starting position in the figure is the position where the robot is turned on and the origin of the world coordinate system. The tile size of the test scene is 65cm×65cm. For the convenience of recording, let the robot move from the starting position in the following figure to the end position in the figure along the red rectangle trajectory.

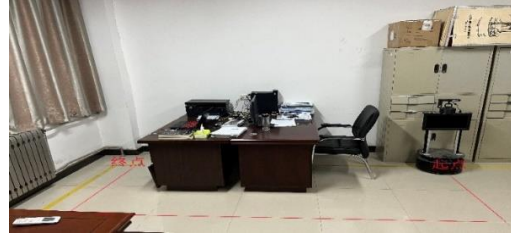


Figure 4. Experimental scenario

In order to facilitate the comparison of odometer data, the remaining three vertices in the rectangular trajectory except the starting point were selected as the target points during the test, and the odometer data was reserved for three decimal places, and the unit was m. The final test results are shown in Table 1 below:

TABLE I. EXPERIMENTAL RESULT

Starting point	Target point	Wheel odometry	Error	Fusion odometry	Error
(0,0)	(1.3,0)	(1.347,0.035)	(0.047,0.035)	(1.324,0.029)	(0.024, 0.029)
	(1.3,-3.25)	(1.391,-3.373)	(0.091,0.123)	(1.386,-3.322)	(0.086, 0.072)
	(0,-3.25)	(0.152,-3.387)	(0.152,0.137)	(0.085,-3.316)	(0.085, 0.066)
Mean error			(0.097,0.098)		(0.065, 0.056)

It can be seen from the test data in the table above that in the rectangular area trajectory of 1.3m×3.25m, the positioning accuracy of the robot is improved by at least 33% compared with the positioning accuracy of the wheeled odometer by fusing the data of the wheeled odometer and IMU. Since the PL-ICP algorithm requires higher initial values than ICP, poor initial values may cause the iterative solution process to fall into a local cycle, so this paper uses the extended Kalman fusion wheel odometer and IMU data, and takes the relatively more accurate data after fusion as the initial value of the PL-ICP algorithm. To a certain extent, the problem that PL-ICP may fall into a local cycle is avoided.

### B. Creation of Fused Two-Dimensional Raster Map

Because the simple two-dimensional laser SLAM mapping can only scan the information of installation height and lack the information of vertical direction, there may be missing information when scanning and mapping the desk, stool and other items in the indoor environment,

such as: When the height of the desktop is higher than the installation height of the lidar, the lidar can only scan the information of the table leg, and there is no obstacle in the middle of the table leg. If the actual height of the robot is higher than the height of the table, the subsequent navigation, because the laser radar can not scan the table, it is determined that there is no obstacle to pass through the middle of the table leg, this situation may lead to damage to the robot. At present, the visual SLAM algorithms based on feature points and direct methods are established in a static environment for measurement, but there are many variables in the real scene, which lead to a serious decline in the positioning accuracy and robustness of the SLAM system, and even lead to the failure of mapping. And pure visual SLAM, due to the limitation of the sensor itself, has low accuracy, is greatly affected by ambient light, and the error of the lack of texture environment is very large. The traditional visual SLAM algorithm does not have the ability to perceive the target in real time, so it is not suitable for simple visual mapping for navigation. However, visual SLAM can scan the vertical information, so the local grid map

established by vision and the local grid map established by laser can be fused. On the one hand, it can make up for the lack of vertical information in laser mapping, and on the other hand, it can also improve the low accuracy of visual mapping. The schematic diagram of the scanning range of laser and vision is shown in Fig. 5:

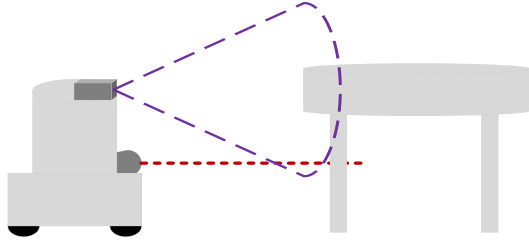


Figure 5. Laser and visual scanning range

The red dotted line in the figure above represents the scanning range of the two-dimensional laser sensor installation position, which can only scan objects in the installation height plane, and the purple dotted line of the cone represents the scanning range of the vision sensor. The Xtion Pro live depth camera used in this paper has a vertical field of view of  $45^\circ$  and a horizontal field of view of  $58^\circ$ .

Since the final established is a two-dimensional raster map, and the depth point cloud information of the depth camera is three-dimensional, it is necessary to project the three-dimensional depth information into two-dimensional pseudo-laser data, and then construct a two-dimensional raster map. Therefore, the final fusion process of the local raster map established by laser and vision is as follows: after the point cloud information is obtained by two-dimensional lidar, the obstacle information obtained by laser is transformed into the raster map coordinate system according to the solved robot pose, and the local two-dimensional raster map of the environment obstacles is formed. At the same time, the data of the depth camera is projected into a pseudo-two-dimensional laser data, which is also transformed into a raster map coordinate system to form a local two-dimensional raster map. Then, the local 2D raster maps of the two are fused to supplement the vertical environmental obstacle information that is not obtained by the 2D laser radar, and a global raster map is formed.

For the fusion of local raster maps, according to the basic principle of raster maps, the Bayesian method is continued to be used for the fusion of raster maps, and the fusion formula [23]:

$$P^0 = \frac{P_s^0 P_m^0}{P_s^0 P_m^0 + (1 - P_s^0)(1 - P_m^0)} \quad (18)$$

In the above equation,  $P_m^0$  and  $1 - P_m^0$  represent the prior probability of occupied and unoccupied grid before fusion respectively,  $P_s^0$  represents the conditional probability of grid state obtained by distance sensor, and  $P^0$  represents the estimated value updated by distance sensor according to the current measurement distance after the obstacle is measured. In the fusion process, the fusion is performed according to the coordinates of the raster according to the rules shown in Table 2 below.

TABLE II. LOCAL MAP FUSION RULES

2D excitation Optical radar	Depth camera		
	Occupy	empty	Uncertain
Occupied	Occupy	Occupy	Occupy
empty	Occupy	empty	empty
Uncertain	Occupy	empty	Uncertain

If the obtained probability of grid occupation  $P^0$  is greater than the initial threshold of the grid  $T_0$ , then the probability of the current grid occupation is set to 1, otherwise it is still  $P^0$ , where  $T_0$  is 0.5. Then the probability value of each grid after the distance sensor measurement is:

$$P_{n=1,2}^0 = \begin{cases} 1 & , P^0 > T_0 \\ P^0 & , P^0 \leq T_0 \end{cases} \quad (19)$$

In the equation,  $P_{n=1}^0 = P_{n=2}^0 = P_2^0$  is the grid probability value corresponding to different sensors, so the probability of grid being occupied after fusion can be obtained by using Bayesian estimation:



$$P_f^0 = \frac{P_1^0 P_2^0}{P_1^0 P_2^0 + (1 - P_1^0)(1 - P_2^0)} \quad (20)$$

#### IV. EXPERIMENTAL VERIFICATION AND ANALYSIS

##### A. Simulation Environment Analysis

The robot used in this paper is Handsfree, so firstly, Gazebo 3D simulation software is used to create a simulation robot model with the same sensors as Handsfree platform, as shown in Fig. 6 after the creation is completed:

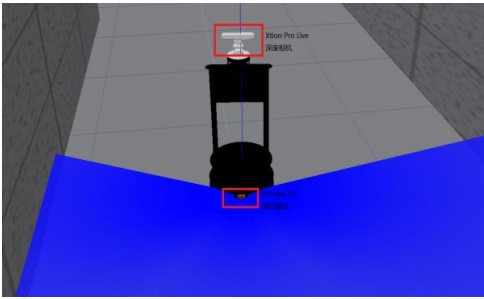


Figure 6. Robot model

The above figure is the overall model of the simulated robot, in which the red boxes are the simulated Xtion Pro Live depth camera and RPLidar A1 lidar model respectively. At the same time, in order to ensure better follow-up experiments, the same sensor parameters of each sensor of the simulated robot and the real robot are set respectively. Then, a simulation experiment environment is established for the experimental effects that need to be verified by the algorithm in this paper. Obstacles such as four-legged desks, T-shaped tables (the height of the desktop is higher than the installation height of the two-dimensional lidar), trucks (the height of the truck body is also higher than the installation height of the two-dimensional lidar) and solid spheres for convenient comparison are set in the simulation experiment environment. The setting of the truck model is to better show the fusion effect on the final created environment map, and the final created simulation experiment environment is shown in Fig. 7:

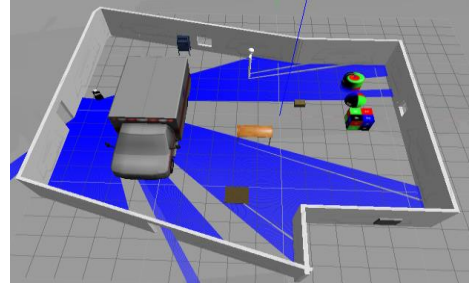
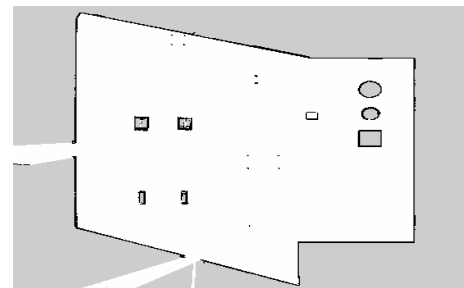
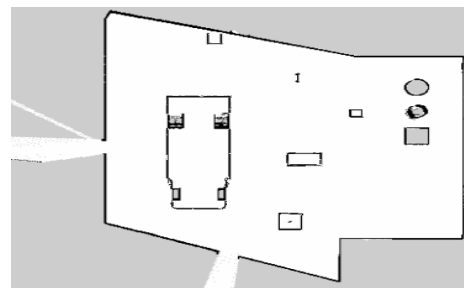


Figure 7. Simulation experimental environment

It can be seen from the above figure that the height of the table and the truck body are higher than the installation height of the 2D lidar, so the scanning point of the 2D lidar passes through the table and the truck. If it is a single 2D LIDAR SLAM, it will be considered that there are no obstacles here, which may lead to the collision between the robot and the table and the truck in the subsequent navigation. The effectiveness of the fusion is verified through the simulation experiment of the single two-dimensional laser sensor mapping and the multi-sensor fusion scheme in this paper, and the angular velocity and linear velocity of the robot motion are kept the same during the experiment. The results are shown in Fig. 8:



(a) Single 2D laser mapping



(b) Fusion mapping

Figure 8. Comparison of simulation experiments

Through the mapping results of the above figure combined with the simulation experiment environment, it can be seen that the two-dimensional lidar can only scan the object with the installation height, and cannot scan the main body of the truck and the desktop with the installation height higher than the lidar in the simulation environment. For example, the blue range in the simulation environment in Fig. 7 is the scanning point of the lidar. As there is no laser information of the main truck and the desktop in the simulation environment, it will be considered that there are no obstacles here. The final mapping result is shown in Fig. 8 (a), and the complete obstacle map information cannot be established for the table and the truck in the simulation environment. For the fusion mapping, since the depth camera can scan the information of the main body of the truck and the table, and project these point cloud information into a fake two-dimensional laser data for mapping, it is very good to establish the complete information of the table and the truck on the final grid map, as shown in Fig.8 (b) , so as to avoid the risk of collision between the robot and the obstacles in the subsequent navigation.

### B. Experimental Verification and Analysis in Real Environment

In order to further verify the effect of fusion, this section will test the algorithm in the real environment of about 70m<sup>2</sup> as shown in Fig. 9 below to verify the effectiveness of the algorithm in the actual environment.



(a) Overall environment



(b) Environmental front

Figure 9. Real experimental environment

FIG. 9 (a) shows the overall situation of the experimental environment, and (b) shows the front part of the experimental environment. In (b), there is a row of conference tables on the right, and the height of the gap left under the table top is higher than the lidar installation height of the mobile robot. This is shown in Fig. 10:



Figure 10. Installation height of LiDAR

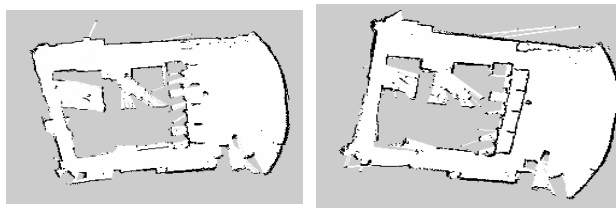
The red circle in the figure above is the comparison between the height of the lidar and the gap under the conference table. Since the gap height of the conference table is larger than the installation height of the lidar, the information above the conference table will not be built into the final grid map of the environment when the single 2D lidar is used for mapping.

During the experiment, keep the linear speed of the robot at 0.2m/s and the angular speed of the steering at 30° per second. Start at the position shown in Fig. 11, circle around the experimental environment clockwise and return to the origin.



Figure 11. Robot departure position

As a comparison, the Gmapping algorithm based on a single two-dimensional lidar and the fusion algorithm in this paper are used for mapping tests, and the motion speed of the robot is kept unchanged during the experiment, and the results are shown in Fig. 12:



(a) Single 2D laser mapping (b) Fusion mapping

Figure 12. Real environment mapping results

TABLE III. POSITIONING RESULTS

Actual location (m)	Actual pose (°)	Gmapping				Vision + Laser			
		Estimated position (m)	Estimated pose (°)	Root mean square error of position (cm)	Attitude error (°)	Estimated position (m)	Estimated pose (°)	Root Mean square error of position (cm)	Pose Error (°)
(6,0)	30	(6.085, 0.079)	33.149	11.604	3.149	(6.059, 0.037)	31.092	5.9	1.092
(6,-3)	90	(6.094, -3.081)	94.634	12.408	4.634	(6.064, -3.051)	92.043	8.184	2.043
(6,-6)	90	(6.103, -6.089)	95.005	13.612	5.005	(6.072, -6.063)	92.729	9.567	2.729
(0,-6)	180	(0.135, -6.112)	186.024	17.541	6.024	(0.089, -6.075)	184.007	11.639	4.007
(0,0)	0	(0.156, 0.127)	7.678	20.116	7.678	(0.091, 0.08)	4.96	12.117	4.96

According to the results in the above table combined with the mapping results, the Gmapping algorithm based on a single two-dimensional lidar relies heavily on the wheeled odometer, and the cumulative error of the wheeled odometer information increases with the movement of the robot, so when the robot finally moves back to the origin, the established environment map has been misaligned. The root mean square error of the position reaches about 20cm. The fusion algorithm

Fig 12 (a) shows the mapping results of Gmapping. It can be seen that part of the information of the conference table is missing in the final raster map, and only the leg information of the conference table is available. In (b), the complete information of the conference table is added to the final raster map, and this information is obtained by fusing the local raster map established by the projection of the depth camera into a pseudo-two-dimensional laser data, so as to verify the effectiveness of the fusion mapping.

In addition, in order to facilitate the recording of the robot's pose during operation, the timing control is used to release forward and turn commands to the mobile robot chassis to control the robot to reach the specified position. Five points at the same position in the two experiments were selected to analyze the accuracy of the pose, and the results are shown in Table 3 below.

in this paper uses the laser odometer, and uses the extended Kalman filter to fuse the data of the wheel odometer and IMU as the iterative initial value of the laser odometer to obtain a more accurate robot pose, so the overall pose error is significantly smaller than Gmapping, and the root mean square error of the robot's position is about 12cm when it returns to the origin. The positioning error of the robot is effectively reduced.

## V. CONCLUSIONS

In this paper, SLAM based on single 2D LiDAR is analyzed and optimized. Extended Kalman filter is used to fuse wheel odometer and IMU data to provide initial iteration values for the laser interframe matching algorithm PL-ICP, and at the same time, the fusion visual projection is built into a local 2D raster map of pseudo-2D laser data. The results show that compared with the single two-dimensional laser SLAM, the multi-sensor fusion SLAM scheme in the paper can improve the pose accuracy of the robot and build a more complete global map of the environment details.

## ACKNOWLEDGMENT

The authors wish to thank the cooperators. This research is partially funded by the Innovation and Entrepreneurship Project Fund for College Students (202310702040).

## REFERENCES

- [1] Randall Smith, Matthew Self. Estimating Uncertain Spatial Relationships in Robotics [J]. CoRR, 2013, abs/1304.3111.
- [2] HAO Rui, Li Rui, SHI Yingjing et al. System self-localization and map reconstruction based on multi-sensor fusion [J/OL]. Radio Engineering :1-9[2023-09-04].
- [3] ZHAO Shaoan. Simultaneous Localization and Mapping for Mobile Robot Navigation with 3D Laser Point Clouds [D]. University of Electronic Science and Technology of China, 2018. (in Chinese)
- [4] ZHAO Yancheng, WEI Tianxu, TONG Di et al. Research on Visual SLAM algorithm based on YOLOv5s in dynamic scenarios [J/OL]. Radio Engineering :1-10[2023-08-09].
- [5] DENG Zibin. Research on Dynamic target tracking algorithm based on Visual SLAM [D]. Southwest University of Science and Technology, 2023.
- [6] Quan M, Piao S, Tan M, et al. Tightly-Coupled Monocular Visual-Odometric SLAM Using Wheels and a MEMS Gyroscope [J]. IEEE Access, 2019, 7:97374-97389.
- [7] XU Tong. Simultaneous Localization and Mapping of Mobile Robot Based on Multi-Source Perception Information Fusion [D]. Dalian Maritime University, 2020. (in Chinese)
- [8] TANG Yuanwen, LIU Zuoshi. Research on multi-sensor Fusion SLAM of mobile robot in Complex Environment [J]. Manufacturing Automation, 2023, 45(08):108-112+166. (in Chinese)
- [9] Zhu J B, Zhao J H, Cui C, et al. Robustness Design of UWB/IMU integrated Navigation System based on EKF [J]. Computer Simulation, 20, 37(12):47-52, 57.
- [10] Ye H, Chen Y, Liu M. Tightly Coupled 3D Lidar Inertial Odometry and Mapping [J]. International Conference on Robotics and Automation (ICRA), 2019, 3144-3150.
- [11] YUE Shengjie, WANG Hongqi, LIU Qunpo, et al. Robot pose adaptive estimation based on Extended Kalman filter and point-line nearest point iterative scan matching algorithm [J]. Surveying and mapping bulletin, 2022 (7) : 49-53.
- [12] Labbé M, François, Michaud O. RTAB - Map as an Open - Source Lidar and Visual Simultaneous Localization and Mapping Library for Large - Scale and Long - Term Online Operation [J]. Journal of Field Robotics, 2019:416-446.
- [13] ZHANG Wen. Research on Autonomous Navigation Method for Indoor Robots Based on Multisensor Fusion [D]. University of Science and Technology of China, 2017. (in Chinese)
- [14] WANG yang. Research on SLAM Mapping Algorithm of Mobile robot based on image optimization [D]. Zhejiang University, 2022. (in Chinese)
- [15] Censi A. An ICP variant using a point-to-line metric [C]// IEEE International Conference on Robotics & Automation. IEEE, 2008.
- [16] LIU Hewei, QING Zhaobo, HUANG Jiajun. Research on interframe matching algorithm based on improved ICP [J]. Modern Electronic Technique, 2023, 46(08):73-78. (in Chinese)
- [17] REN Xisnghua. Lidar Indoor SLAM Method [D]. Harbin Engineering University, 2018. (in Chinese)
- [18] SUN Jian, LIU Longhui, LI Zhi, et al. Mobile Robot Sensor Data Fusion Algorithm Based on RGB-D Camera and Lidar [J]. Journal of Hunan Institute of Engineering(Natural Science Edition), 2022, 32(01):18-24. (in Chinese)
- [19] JIANG Zuopeng, MEI Tiancan. A single-line laser odometer based on PL-ICP and NDT point cloud matching [J]. Laser Journal, 2020,41(03):21-24.
- [20] WANG Zirun, YAN Bixi, DONG Mingli et al. Positioning method of wall-climbing Robot based on LiDAR and improved AMCL [J]. Chinese Journal of Scientific Instrumentation, 202, 43(12):220-227. (in Chinese)
- [21] FENG Bin. Research on Simultaneous Location and Mapping Algorithm Based on Bayesian Filter [D]. Chang'an University, 2020. (in Chinese)
- [22] ZHANG Ke. Research on Mapping and Navigation Algorithm of Indoor Robot Based on Depth Camera[D]. Harbin Institute of Technology, 2020. (in Chinese)
- [23] ZHANG Yi, DU Fanyu, LUO Yuan, et al. Map-building Approach Based on Laser and Depth Visual Sensor Fusion SLAM [J]. Application Research of Computers, 2016, 33(10):2970-2972+3006. (in Chinese)