IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 11 July 2023, accepted 1 September 2023, date of publication 14 September 2023, date of current version 20 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3315326

RESEARCH ARTICLE

Improved ORB-SLAM2 Mobile Robot Vision Algorithm Based on Multiple Feature Fusion

XIAOMEI HU[1], LUYING ZHU[1], PING WANG[1], HAILI YANG[2], AND XUAN LI[1]
[1]College of Intelligent Science and Engineering, Yantai Nanshan University, Yantai 265713, China
[2]School of Electric, Civil Engineering and Architecture, Shanxi University, Taiyuan 030000, China

Corresponding author: Xiaomei Hu (hxm2023@126.com)

ABSTRACT Traditional wheeled robot vision algorithms suffer from low texture tracking failures. Therefore, this study proposes a vision improvement algorithm for mobile robots in view of multi feature fusion; This algorithm introduces line surface features and Manhattan Frame on the basis of traditional algorithms, and proposes an improved algorithm in view of multi-sensor fusion to improve tracking accuracy. The experiment shows that the average Root-mean-square deviation of the position of the improved mobile robot vision algorithm in view of multi feature fusion is 0.02 in nine data packets of the Tum dataset; The average Root-mean-square deviation of the position of the data packet successfully tracked by the traditional wheeled robot vision algorithm is 0.016; It improved the average accuracy by 11.11%, which is 31.03% higher than the average accuracy of the Manhattan wheeled robot vision algorithm. Compared to the multi feature fusion based vision improvement algorithm for mobile robots and the closed-loop detection based multi-sensor improvement algorithm, the accuracy of the closed-loop detection based multi-sensor improvement algorithm has increased by 0.655% and 10.47%, respectively. The outcomes indicate that the improved algorithm can improve the accuracy of mobile robot tracking, thereby expanding its application range.

INDEX TERMS Multi feature fusion, mobile robots, visual algorithms, multi sensor fusion, encoder.

I. INTRODUCTION

With the development of intelligent robot technology, wheeled robots have gradually entered the public's field of vision [1]. Compared to other types of robots, wheeled robots have advantages such as strong flexibility, fast movement speed, and low maintenance costs, and are widely used in fields such as logistics, surveying, education, and healthcare [2]. In the autonomous movement of wheeled robots, the Simultaneous Location and Map Creation (SLAM) algorithm was first used for localization. Through SLAM technology, robots can explore and navigate independently in unknown environments [3]. Later, the camera based robot vision SLAM algorithm was gradually applied to wheeled robots. With the development of deep learning technology, visual SLAM algorithms can be combined with deep learning, but deep learning requires a large amount of data, which limits its application scenarios [4]. Traditional robot visual tracking algorithms may experience significant tracking errors and tracking failures, mainly due to hardware, environment, and control algorithms [5]. Among them, robot vision algorithms may experience saturation during the error integration process, resulting in inaccurate path tracking and planning. Therefore, the study proposes a vision improvement algorithm for mobile robots based on multi feature fusion, with the aim of improving the tracking performance of robots. The main content and innovation of the research are mainly divided into two aspects. The first aspect is to improve the ORB-SLAM2 algorithm by using point, surface, and surface feature fusion to minimize the objective function to estimate the current pose, thereby improving tracking performance. The second aspect is to improve the multi feature fusion algorithm by integrating monocular cameras, encoders, and inertial measurement units, utilizing back-end optimized objective

The associate editor coordinating the review of this manuscript and approving it for publication was Nikhil Padhi.

functions, etc., to improve positioning accuracy. The research is divided into four parts. The first part is a summary of existing robot vision algorithms, the second part is the design of improved algorithms for mobile robot vision, the third part is the verification of the effectiveness of the improved algorithms, and the fourth part is a summary of the research.

II. RELATED WORKS

In an unknown environment, the SLAM algorithm enables the robot's own sensors to simultaneously locate and establish an environmental map, which is used for positioning, planning, and control. According to the main sensor types, SLAM algorithms are divided into laser SLAM and visual SLAM. With the improvement of processor computing speed, the advantages of cameras such as low cost, rich information, and easy installation have gradually become prominent. Visual SLAM has gradually become one of the research hotspots and has broad application prospects in the field of robotics. Researchers have conducted many studies on the SLAM algorithm for robots, such as W. Zhang's scientific research team proposed a SLAM optimization algorithm for node coverage robot in view of improved Particle filter; This algorithm utilizes sensor node construction to improve the coverage of location information nodes and optimizes them in local states; The relevant outcomes illustrate that the positioning stability of this method is superior to traditional SLAM algorithm [6]. B. Fang's research group proposes a dynamic scene SLAM algorithm in view of boundary box and depth continuity; This algorithm uses a deep bounding box to fill in pixels for random search, and eliminates the influence of the target through dynamic feature filtering; The experimental results show that the positioning accuracy and real-time performance of this method in complex dynamic scenes meet the design expectations [7]. Hu et al. proposed an improved ORB-SLAM front-end tracking algorithm that utilizes a uniform velocity model to track effective frames and adjacent frames, and matches similar frames; Experimental data shows that this method can increase the number of effective tracking frames and reduce the computational complexity by two times [8]. Dong et al. proposed an improved RGB-D SLAM scheme that utilizes ORB for feature point extraction and descriptor calculation, and matches the current frame (CF) with the map; The research results show that this method reduces the Root-mean-square (RMS) deviation by 9% on average, and improves the indexing effect of point cloud images [9]. Su and Yu proposed a deep image SLAM reconstruction algorithm in view of multi-layer image invariant feature transformation; This algorithm utilizes the Convolutional neural network of multi-layer image invariant features to optimize the point feature extraction and reconstruction, and constructs a smooth and complete spatial model through 3D point cloud stitching; Comparative experiments have shown that this algorithm saves 0.093 seconds of time compared to ordinary extraction algorithms, and the reconstruction results are superior to algorithms such as Kintinous [10].

Robot vision algorithms refer to the visual technologies and algorithms applied in the process of robot perception, recognition, understanding, and operation. With the development of computer technology, the application range of machine vision detection technology is becoming increasingly widespread, which can be applied in fields such as agriculture and industry. Many researchers apply machine vision to specific scenarios and make improvements based on actual situations, such as B. Li's research group proposes a navigation path detection algorithm in view of improved random sampling consistency; This algorithm uses inverse perspective mapping to construct the original wheat image, and combines the acquisition boundary Corner detection to improve the recognition performance; This study indicates that robot visual navigation can help improve the effectiveness of crop processing [11]. S. Liang's scientific research team uses support vector machine to optimize the auto drive system of the vehicle, extracts and trains the features of the input image to get the classifier model, and then realizes the target detection and recognition of the robot; This study indicates that robot vision and cruising can improve the safety of vehicle autonomous driving [12]. Cui et al. utilized the edge detection principle of machine vision for trajectory planning, and combined it with the toilet seat trimming process to construct a new edge detection method that utilizes reflection areas to eliminate false edges in the image; This study indicates that this method can improve the accuracy of edge detection [13]. T. Liu's research group proposes a resolution enhancement algorithm with total change constraint for visual recognition of industrial materials; This algorithm utilizes spectral imaging systems and Fourier theory to calculate kernel functions; The experiment showcases that this method can markedly segment overlapping bands and improve the visual sensing accuracy of industrial robots [14]. E. The Cristofalo research group proposes an active perception controller; In this controller, the extended Kalman filter is utilized for estimating the position of the robot, and the derived controller is utilized for estimating the 3D position of the feature; The results indicate that this method is superior to traditional active sensing methods and helps to improve the estimation accuracy of aviation robots [15].

In summary, the relevant robot vision algorithms assume that wheeled robots perform planar motion and use encoder pre integration to calculate the encoder observation values between two or more frames of images, increasing encoder constraints and improving positioning accuracy. Due to the encoder's pre integration time reaching tens or even hundreds of milliseconds, when the ground is rough or there are small objects, the robot does not move in the plane, which can lead to significant observation errors, leading to a decrease in positioning accuracy and even tracking failure. Therefore, in response to the problems of accuracy degradation and tracking loss in low texture and robot turning scenarios caused by visual SLAM, a mobile robot vision improvement algorithm based on multi feature fusion is proposed to improve the tracking performance of visual robots.

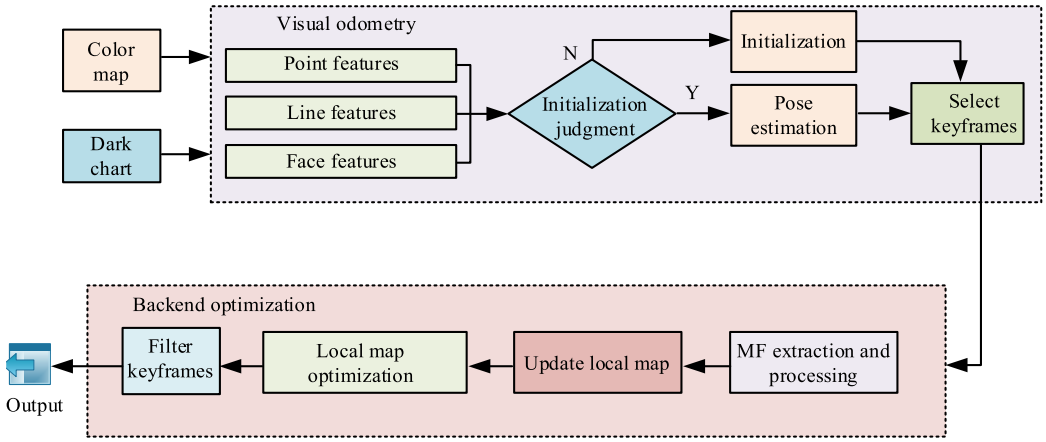


FIGURE 1. An improved ORB-SLAM2 algorithm framework in view ofMFF.

III. DESIGN OF IMPROVED ALGORITHM FOR MR VISION

This chapter is about the design of vision improvement algorithms for MR; The first section of this chapter introduces line and surface features into the depth camera version and proposes an improved MR vision algorithm in view of MFF. The tracking and local map threads of the algorithm correspond to visual odometry and backend optimization, respectively; The second section of this chapter is an improved algorithm in view of multi-sensor fusion; The algorithm includes sensor pre-processing, joint initialization of vision Inertial measurement unit encoder, sliding window optimization and closed-loop (CL) optimization. Encoder pre-processing and related constraints are introduced in combination with nonlinear optimization methods to improve the positioning accuracy.

A. IMPROVED ORB-SLAM2 ALGORITHM IN VIEW OF MULTIPLE FEATURE FUSION

At present, wheeled robots are widely used, with two and three wheeled MR being common [16]. This study is in view of the background of a two wheel differential MR and improves the visual algorithm (VA) of the MR. This study introduces line and surface features into the depth camera version and proposes an improved feature point detection and extraction - simultaneous localization and mapping 2 (Oriented FAST and Rotated BRIEF Simultaneous Looseness and Mapping 2) ORB-SLAM2 algorithm in view of MFF. The framework of the improved ORB-SLAM2 algorithm in view of MFF is shown in Figure 1.

The improved algorithm consists of two parts: visual odometer and backend optimization, corresponding to the tracking thread and local map thread of ORB-SLAM2, respectively. The visual odometer is responsible for real-time processing of color and depth maps, completing initialization and estimating the current frame pose, and determining whether to insert the current frame as a keyframe into the backend. Backend optimization is responsible for processing keyframes, reconstructing new map points and lines through triangulation, and estimating state variables such as keyframe pose, point, line, and surface positions in local maps using nonlinear optimization methods.The ORB-SLAM2 algorithm's tracking and local map threads correspond to visual odometry and backend optimization, respectively. The improvement lies in the introduction of line surface features and Manhattan Frame (FM). The real-time processing of color and depth maps is completed through a visual odometer to initialize and estimate the CF pose, and determine the insertion status of key frames. The specific processing process cannot be separated from the extraction and processing of point, line, and surface features. Key frames are processed through backend optimization, triangulation reconstruction of map points and lines, and estimation of key frame pose and position of points, lines, and surfaces using nonlinear optimization methods.

The representative regions in the image are the features of the image, which are divided into key regions and descriptors to establish image association relationships. The area where the features are located in the image is called the key area, which is divided into key points, lines, and surfaces. The description information of the features is called a descriptor to calculate the similarity and differentiation of the features. Point features are widely used in SLAM, and the improved algorithm selects ORB point features for analysis. The centroid calculation of image blocks is shown in equation (1).

$$\begin{cases} c = (\dfrac{m_{10}}{m_{00}}, \dfrac{m_{01}}{m_{00}}) \\ m_{bd} = \sum\limits_{x,y \in B} x^b y^d I(x, y) \end{cases} \tag{1}$$

In equation (1), the center of mass is represented by c; The pixel value of the pixel coordinate (x, y) in the image is represented by $I(x, y)$; The image area is represented by B; b and d represent different positions of 0 and 1, and the corresponding pixel values under different values are represented by m_{bd}. The key point direction is the reference direction of descriptor extraction. The feature similarity during point

feature matching is inversely proportional to Hamming distance. In a certain threshold range, two point features are considered to match. The threshold range in the improved algorithm is [50,100]. The next level feature of point features is line features, which describe the contour of the object. This study uses Fast Line Detector (FLD) to describe key lines and extracts binary Line Band Descriptor (LBD) descriptors. In the process of line feature matching, judge in view of Hamming distance. A higher level than points and lines is the surface feature, which describes the object surface. The research optimizes the Hierarchical clustering (AHC) algorithm to achieve the extraction of surface features from the depth map. The observed values of line features are shown in equation (2).

$$\tilde{l} = \Gamma(\tilde{p}_s) \times \Gamma(\tilde{p}_e)/(\|\Gamma(\tilde{p}_s)\| \cdot \|\Gamma(\tilde{p}_e)\|) \tag{2}$$

In equation (2), the observed values of line features are represented by $\tilde{l}$; The coordinates of the starting pixel and the ending pixel are represented by $\tilde{p}_s$ and $\tilde{p}_e$, respectively; Homogeneous coordinate operation is represented by $\Gamma(\cdot)$; The vector modulus operation is represented by $\|\cdot\|$. The optimized surface feature representation is shown in equation (3).

$$\Upsilon(\tilde{\pi}) = [\arctan(\tilde{n}_y/\tilde{n}_x), \arctan(\tilde{n}_z), d_\pi]^T \tag{3}$$

In equation (3), the surface without redundancy is represented by $\Upsilon(\tilde{\pi})$; Improve the representation of AHC surface features with $\tilde{\pi}$; The normal vector of the camera coordinate system (CS) for the CF is represented by $\tilde{n} = [\tilde{n}_x, \tilde{n}_y, \tilde{n}_z]$; The distance between the origin of the camera CS and the face in the CF is represented by d_π. Before the algorithm starts, an initial 3D map needs to be constructed, and the position calculation of the world CS is demonstrated in equation (4).

$$P_h^w = \tilde{d}_h \Pi^{-1}(\tilde{P}_h) \tag{4}$$

In equation (4), the pixel coordinates and depth of the h-th feature point are represented by $\tilde{P}_h$ and $\tilde{d}_h$, respectively; The world CS position of the h-th map point is represented by P_h^w; The backprojection function is represented by Π^{-1}. This study used the Levenberg Marquardt (LM) method for solving the least squares problem in VA. The process of solving the least squares problem under the LM method is shown in Figure 2.

When solving the least squares problem using the LM method, the increment is constantly searched for and updated based on the given initial value, causing the objective function to decrease. When the increment is small enough or reaches the iteration number, it ends and outputs the optimal solution.The key for solving the least squares issue is the calculation of the objective function and Jacobian matrix and determinant. The algorithm estimates the CF position through the optimal positioning under the LM method. The objective

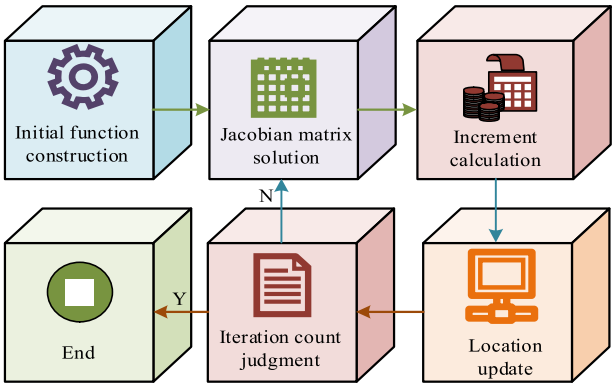


FIGURE 2. The least squares problem solving process under LM method.

function under the LM method is shown in equation (5).

$$F(R_w^{c_k}, p_w^{c_k}) = \sum_{h=0}^{H_1} \left\| e_{p,k-h} \right\|^2_{\Sigma_{p,k-h}} + \sum_{i=0}^{I_1} \left\| e_{l,k-i} \right\|^2_{\Sigma_{l,k-i}} + \sum_{j=0}^{J_1} \left\| e_{\pi,k-j} \right\|^2_{\Sigma_{\pi,k-j}} \tag{5}$$

In equation (5), the inverse of the world CS pose of the camera CS c_k is represented by $(R_w^{c_k}, p_w^{c_k})$; The Covariance matrix of point, line and plane measurement noise is expressed as $\Sigma_{p,k-h}$, $\Sigma_{l,k-i}$ and $\Sigma_{\pi,k-j}$ respectively; The total number of matching point sets is represented by H_1, and the number of matching line set buses is represented by I_1; The total number of matching faces in the set is represented by J_1; The measurement error terms of point, line, and surface are represented by $e_{p,k-h}$, $e_{l,k-i}$, and $e_{\pi,k-j}$. The relevant expression for point measurement error is shown in equation (6).

$$e_{p,k-h} = \tilde{p}_h^{c_k} - \Pi(R_w^{c_k} P_h^w + p_w^{c_k}) \tag{6}$$

In equation (6), the position of the h-th map point and its CF observation values are represented by P_h^w and $\tilde{p}_h^{c_k}$, respectively; The forward projection function is represented by Π. The relevant expression for line measurement error is shown in equation (7).

$$e_{l,k-i} = (\tilde{l}_i^{c_k})^T \Gamma(\Pi(R_w^{c_k} P_{s_i}^w + p_w^{c_k})) + (\tilde{l}_i^{c_k})^T \Gamma(\Pi(R_w^{c_k} P_{e_i}^w + p_w^{c_k})) \tag{7}$$

In equation (7), the starting and ending points of the i-th line in the world CS are represented by $P_{s_i}^w$ and $P_{e_i}^w$, respectively; Its CF observation is represented by $\tilde{l}_i^{c_k}$. The formula for calculating the surface measurement error term is shown in equation (8).

$$e_{\pi,k-j} = \Upsilon(\tilde{\pi}_j^{c_k}) - \Upsilon([(R_w^{c_k} n_j^w)^T, d_j^w - (n_j^w)(R_w^{c_k})^T p_w^{c_k}]^T) \tag{8}$$

In equation (8), the world CS surface of the j-th face is represented by $\pi_j^w = [(n_j^w)^T, (d_j^w)]^T$; The CF observation

value is represented by $\tilde{\pi}_j^{c_k}$. Under the call of g2o library, combined with the Jacobian matrix and determinant of point, line and plane error terms, the automatic calculation of all error terms Jacobian matrix and determinant can be realized. Combined with the objective function under LM method, the CF pose can be solved. If the new frame tracking fails, it needs to be repositioned and the CF pose estimated. When calculating the pose of the CF, it is affected by the incomplete perpendicularity of the line and surface direction vectors. The coordinate axis vector needs to be corrected by combining singular decomposition method. After singular decomposition, the pose of the CF is shown in equation (9).

$$\tilde{R}_{c_k}^{w} = R_{c_r}^{w} R_{MF}^{c_r} (R_{MF}^{c_k})^{-1} \tag{9}$$

In equation (9), the CF pose is represented by $\tilde{R}_{c_k}^{w}$; The reference frame posture is represented by $R_{c_r}^{w}$; The coordinate axis matrix is represented by $R_{MF}^{c_r}$; The reverse vector of the coordinate axis is represented by $R_{MF}^{c_k}$. In the improved algorithm, the calculation of the map position of the CF's 2D point line includes point line matching in the visual odometer. After successful matching, the 2D point of the CF is obtained, and its depth value and map point line position are calculated. The remaining point lines need to be matched with triangulation, as shown in Figure 3.

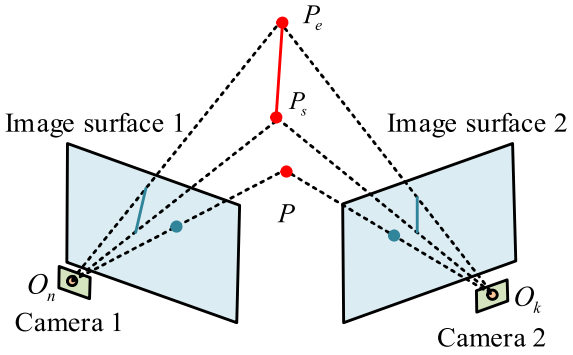


FIGURE 3. Triangulation of points and lines.

In point and line triangulation, the optical centers of frames n and k are represented by O_n and O_k, respectively; Map points are represented by P; The starting and ending points of map lines are represented by P_s and P_e, respectively. Local map optimization involves solving the pose of key frames and the position of points, lines, and planes, with the definition of state variables as shown in equation (10).

$$\begin{aligned}\chi = [&R_{c_0}^{w}, R_{c_1}^{w} \cdots, R_{c_{k'}}^{w}, p_{c_0}^{w}, p_{c_1}^{w}, \cdots, p_{c_{k'}}^{w}, P_0^{w}, P_1^{w}, \cdots, P_H^{w}, \\ &P_{s_0}^{w}, P_{s_1}^{w}, \cdots, P_{s_I}^{w}, P_{e_0}^{w}, P_{e_1}^{w}, \cdots, P_{e_I}^{w}, \pi_0^{w}, \pi_1^{w}, \cdots, \pi_J^{w}]\end{aligned} \tag{10}$$

In equation (10), the local map keyframes and the total number of points, lines, and surfaces are represented by K', H, I, and J, respectively. The objective function is shown in equation (11).

$$\begin{aligned}F(\chi) = \sum_{k=0}^{K'} (&\sum_{h=0}^{H_k} \left\| e_{p,k-h} \right\|_{\Sigma_{p,k-h}}^{2} + \sum_{i=0}^{I_k} \left\| e_{l,k-i} \right\|_{\Sigma_{l,k-i}}^{2} \\ &+ \sum_{j=0}^{J_k} \left\| e_{\pi,k-j} \right\|_{\Sigma_{\pi,k-j}}^{2} + \left\| e_{MF,k} \right\|_{\Sigma_{MF,k}}^{2})\end{aligned} \tag{11}$$

In equation (11), the number of point, line, and surface set elements matched in the k-th keyframe is represented by H_k, I_k, and J_k, respectively; The error term of MF measurement and the noise Covariance matrix are represented by $e_{MF,k}$ and $\Sigma_{MF,k}$ respectively; Combining the construction of Jacobian matrix and determinant and the call of g2o library, the key frame pose and point line plane position in the local map can be solved. In local maps, keyframes may increase rapidly over time, and excess keyframes need to be eliminated.

B. IMPROVED ALGORITHM IN VIEW OF MULTI-SENSOR FUSION

The improved algorithm for MFF may encounter issues of reduced accuracy and tracking loss when features are sparse [17]. Therefore, this study proposes an improved algorithm in view of MFF, which introduces the encoder error term into the objective function to improve scale observability and reduce cumulative positioning error. The framework of the improved algorithm in view of MFF is shown in Figure 4.

The improved algorithm includes sensor pre-processing, joint initialization of vision Inertial measurement unit encoder, sliding window optimization and CL optimization. Encoder pre-processing and related constraints are introduced, combined with nonlinear optimization methods, to improve the positioning accuracy. In CL testing, adaptive adjustment of the quantity of point features is used to improve the success rate of the CL. In Inertial measurement unit preprocessing, the measurement model in discrete time is shown in Formula (12).

$$\begin{cases} \tilde{a}_{k+l/n} = a_{k+l/n} + R_w^{b_{k+l/n}} g^w + b_{a_k} + n_a \\ \tilde{\omega}_{k+l/n} = \omega_{k+l/n} + b_{\omega_k} + n_\omega \end{cases} \tag{12}$$

In equation (12), the body acceleration and angular velocity are represented by $\tilde{a}_{k+l/n}$ and $\tilde{\omega}_{k+l/n}$, respectively; The true values of acceleration and angular velocity are represented by $a_{k+l/n}$ and $\omega_{k+l/n}$; The vector of gravity in the world coordinate is represented by g^w, and the Rotation matrix of the airframe CS at the time of Inertial measurement unit data in frame l of the world coordinate is represented by $R_w^{b_{k+l/n}}$; The Gaussian measurement noise of acceleration and angular velocity is represented by n_a and n_ω, respectively; The acceleration and angular velocity offsets between two frames are represented by b_{a_k} and b_{ω_k}, respectively. The average acceleration and average angular velocity of the body

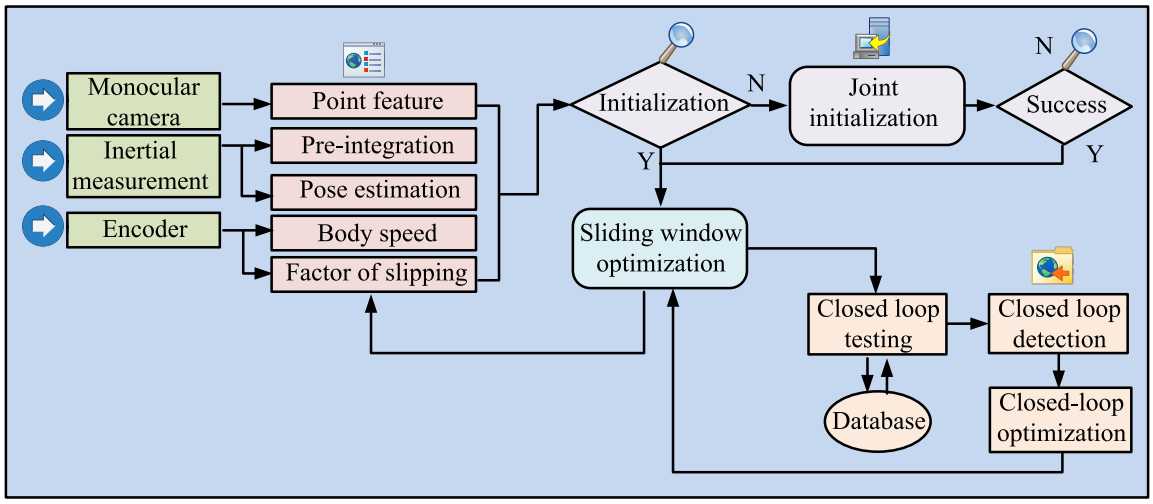


FIGURE 4. An improved algorithm framework in view of MFF.

of the gravity vector are calculated as shown in equation (13).

$$\begin{cases} \overline{a}^{b_k}_{k+l/n,k+(l+1)/n} \\ = 0.5[R^{b_k}_{b_{k+l/n}}(\tilde{a}_{k+l/n} - b_{a_k}) \\ \quad + R^{b_k}_{b_{k+(l+1)/n}}(\tilde{a}_{k+(l+1)/n} - b_{a_k})] \\ \overline{\omega}_{b_{k+l/n,k+(l+1)/n}} = 0.5[\tilde{\omega}_{k+l/n} + \tilde{\omega}_{k+(l+1)/n}] - b_{\omega_k} \end{cases} \tag{13}$$

In equation (13), the average acceleration and average angular velocity are represented by $\overline{a}^{b_k}_{k+l/n,k+(l+1)/n}$ and $\overline{\omega}_{b_{k+l/n,k+(l+1)/n}}$; The rotation of the body CS at frame l relative to frame 0 is represented by $R^{b_k}_{b_{k+l/n}}$. The settlement of Inertial measurement unit observation value between two images is shown in Formula (14).

$$\begin{cases} \tilde{a}^{b_k}_{b_{k+1}} = 0.5\overline{a}^{b_k}_{k,k+1/n}T_I^2 + \sum_{l=1}^{n-1}\sum_{i=0}^{l-1}\overline{a}^{b_k}_{k+i/n,k+(i+1)/n}T_I^2 \\ + \sum_{l=1}^{n-1}\sum_{i=0}^{l-1}\overline{a}^{b_k}_{k+l/n,k+(l+1)/n}T_I^2 \\ \overline{\beta}^{b_k}_{b_{k+1}} = \sum_{l=0}^{n-1}\overline{a}^{b_k}_{k+l/n,k+(l+1)/n}T_I \\ \overline{\gamma}^{b_k}_{b_{k+1}} = q^{b_k}_{b_{k+1}} \end{cases} \tag{14}$$

In equation (14), the velocity increment, relative displacement, and relative rotation between two frames are represented by $\tilde{a}^{b_k}_{b_{k+1}}$, $\overline{\beta}^{b_k}_{b_{k+1}}$, and $\overline{\gamma}^{b_k}_{b_{k+1}}$, respectively. The acceleration measurement value of Inertial measurement unit needs to remove the gravity vector to realize its transformation to the world CS. The first step of encoder preprocessing is to calculate the body speed observation value to determine the encoder error term in the joint initialization and optimization of vision Inertial measurement unit encoder, and determine the degree of wheel slip through the calculation of robot slip factor. The linear and angular velocities of the robot body in the encoder coordinates are calculated as shown in equation (15) [18].

$$\begin{cases} v_{e_k} = \dfrac{v_r + v_l}{2} \\ \omega_{e_k} = \dfrac{v_r - v_l}{D} \end{cases} \tag{15}$$

In equation (15), the left and right wheel speeds at frame k are represented by v_l and v_r respectively; The linear velocity and angular velocity in the encoder coordinates are represented by v_{e_k} and ω_{e_k}, respectively; The distance between two wheels is represented by D. At this point, the encoder speed at this moment can be represented by the matrix of linear speed and converted into the speed observation value of Inertial measurement unit. The speed measurement accuracy of the encoder's two wheels will be affected by wheel slip, thereby affecting the accuracy of linear and angular velocities; The angular velocity measurement of the Inertial measurement unit is accurate. The measured values of the three angular velocity meters can be used as a reference, and the slip factor can be solved by using the angular velocity difference between the two units. The value range of the slip factor is 0 to 1, and a smaller value indicates a more pronounced slip phenomenon [19]. When the algorithm starts, the initial value of the state variable required for the sliding window optimization is obtained through initialization calculation, which specifically includes the position, attitude and speed of the airframe, the location of the location map, and the angular velocity offset of the Inertial measurement unit. Compared to traditional multi-sensor fusion algorithms, the improved algorithm introduces the encoder error term into the objective function to achieve accuracy and improvement in the calculation of body speed, gravity vector, and scale factor, ensuring that initialization can proceed normally when the signal-to-noise ratio of the accelerometer is small. Early judgment

of slip factors helps to stabilize the initialization results; If the slip factor is too small, initialization needs to be exited; When the slip factor is appropriate, the fixed frame data preprocesses the image before initialization. The initialization process is shown in Figure 5.

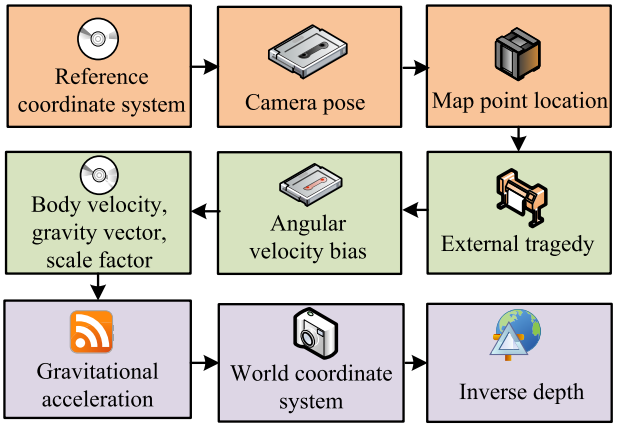


FIGURE 5. Initialization process.

The first step in the initialization step is to determine the reference CS. The first frame image CS can be utilized as a reference, and combined with the five point method, the camera pose of the frame image with the largest relative parallax in the first frame is calculated. Through triangle matching, two-dimensional point features are obtained, and the three-dimensional pose of the point in the reference coordinate is calculated [20]. Subsequently, the PnP method is utilized for tracking the 3D location map and calculate the pose of the remaining frames. After matching the features of the remaining 2D points, the reconstruction of the location map can be achieved. Finally, the camera pose and location of the location map in the sliding window without scale are optimized. The second step is to calculate the attitude at different image times and the external parameters of the Inertial measurement unit, and take the position and attitude of the reference CS at zero time of the airframe as the observation value. The optimal solution of the angular velocity offset of the Inertial measurement unit of all image frames is calculated by minimizing the objective function. The third step is to calculate the body velocity, gravity vector, and scale factor; The fourth step is to compare the gravity vector to obtain a more accurate gravity vector; The fifth step is to rotate the reference CS to the world CS, and obtain the pose, velocity, and position of the frame image in the world CS in view of variables such as camera pose. After initialization is completed, sliding window optimization processing is carried out before the data preprocessing results of the next frame are conveyed. The optimization of vision Inertial measurement unit encoder sliding window is shown in Figure 6.

Sliding window optimization includes the objective function solution of vision, Inertial measurement unit and encoder data to obtain the positioning results. This study introduces the encoder error term into the objective function for

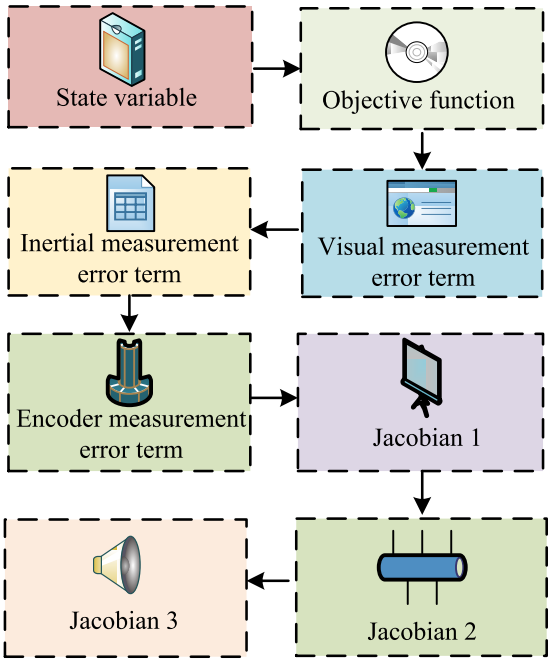


FIGURE 6. Vision inertial measurement unit encoder sliding window optimization.

improving the accuracy of robot positioning. The CL detection and optimization method of the improved algorithm is the same as the original algorithm to eliminate cumulative errors. Firstly, key frame features are extracted for CL detection and optimization [21]. This study adaptively adjusts the brightness difference threshold in view of the extraction of the number of ORB point features. The threshold needs to be lowered when there are fewer point features to improve the success rate of the CL. The foundation of CL detection is the word bag constructed by ORB point feature descriptors. The first step is the extraction of ORB features. By adjusting the brightness difference threshold, key points are extracted and their descriptors are obtained; The second step is to search for similar CL frames in view of descriptors and word bags; The third step is to combine the CF and descriptor for feature point matching, to obtain matching point pairs, and use Ransac algorithm to remove mismatched point pairs; The fourth step is to use the 3D coordinates of the matching points and the 2D pixel coordinates of the CL frame to calculate the machine posture in the world CS of the CL frame [22]. The CL error term and sequence error term form the objective function of CL optimization, which optimizes the position and heading angle of the 4-degree-of-freedom airframe in the CL. The CL optimized body position, original roll angle, and pitch angle form a 6-degree of freedom body posture, thereby forming the positioning results under the CL.

IV. EXPERIMENTAL ANALYSIS OF MR VISION ALGORITHM

This chapter conducts experimental verification and analysis on the algorithm proposed in Chapter 2. The first section

of this chapter is the experimental analysis of the improved ORB-SLAM2 algorithm in view of MFF. The Tum dataset is used to experiment on MR, and the traditional ORB-SLAM2 algorithm and Manhattan SLAM algorithm are compared. The second section of this chapter is the experimental analysis of the improved algorithm in view of multi-sensor fusion, and compares the improved ORB-SLAM2 algorithm in view of MFF.

A. EXPERIMENTAL ANALYSIS OF IMPROVED ORB-SLAM2 ALGORITHM IN VIEW OF MULTIPLE FEATURE FUSION

This study conducted experiments on MR using the Tum dataset for verifying the performance of the improved ORB-SLAM2 algorithm in view of MFF.And compare the traditional ORB-SLAM2 algorithm with Manhattan SLAM algorithm, as well as the commonly used point line feature fusion SLMA algorithm (labeled PL SLAM) and loop detection SLAM algorithm (labeled LD SLAM) from 2021 to 2022, and closes the closed loop optimization thread of the ORB-SLAM2 algorithm and the dense thread optimization of the Manhattan SLAM algorithm. This is to ensure the fairness of the comparative experiment and preserve the sparse mapping thread. The trajectory of the algorithm on the Tum dataset is illustrated in Figure 7.

Figure 7 shows that on low texture data packets, the traditional ORB-SLAM2 algorithm has less point feature extraction and may experience tracking failures. The commonly used PL-SLAM and PL-SLAM algorithms from 2021 to 2022 have added line features and improved tracking failure, but the accuracy of trajectory estimation is relatively low. Improved ORB-SLAM2 algorithm and Manhattan SLAM algorithm by adding line and surface features, achieving successful prediction of motion trajectories and improving the stability of the algorithm in low texture scenes; The trajectory of the improved ORB-SLAM2 algorithm is closer to the real trajectory, improving the authenticity of the algorithm's tracking. In the experiment, the location RMS deviation is used to quantify the error of the algorithm. The comparison between the location RMS deviation and MF quantity of the algorithm on the Tum dataset is shown in Table 1.

In Table 1, the location RMS deviation of the failed algorithm packet tracking is null, the average location RMS deviation of the successful ORB-SLAM2 algorithm packet tracking is 0.018, and the average location RMS deviation of the Manhattan SLAM algorithm is 0.029; Among the nine data packets in the Tum data set, the average value (AEV) of the RMS deviation of the location of the improved algorithm is 0.02, and the AEV of the RMS deviation of the location of the data packets successfully tracked by the ORB-SLAM2 algorithm is 0.016; It improved the average accuracy by 11.11%, which is 31.03% higher than the average accuracy of the Manhattan SLAM algorithm. Relative to the Manhattan SLAM algorithm, the improved algorithm has little difference in the number of MFs in the first four data packets. For the fifth, eighth, and ninth data packets, the improved algorithm has 273, 1116, and 18 more MFs than the Manhattan SLAM algorithm, respectively; This indicates that the improved algorithm has more MF for feature extraction and stronger tracking authenticity. The average time comparison

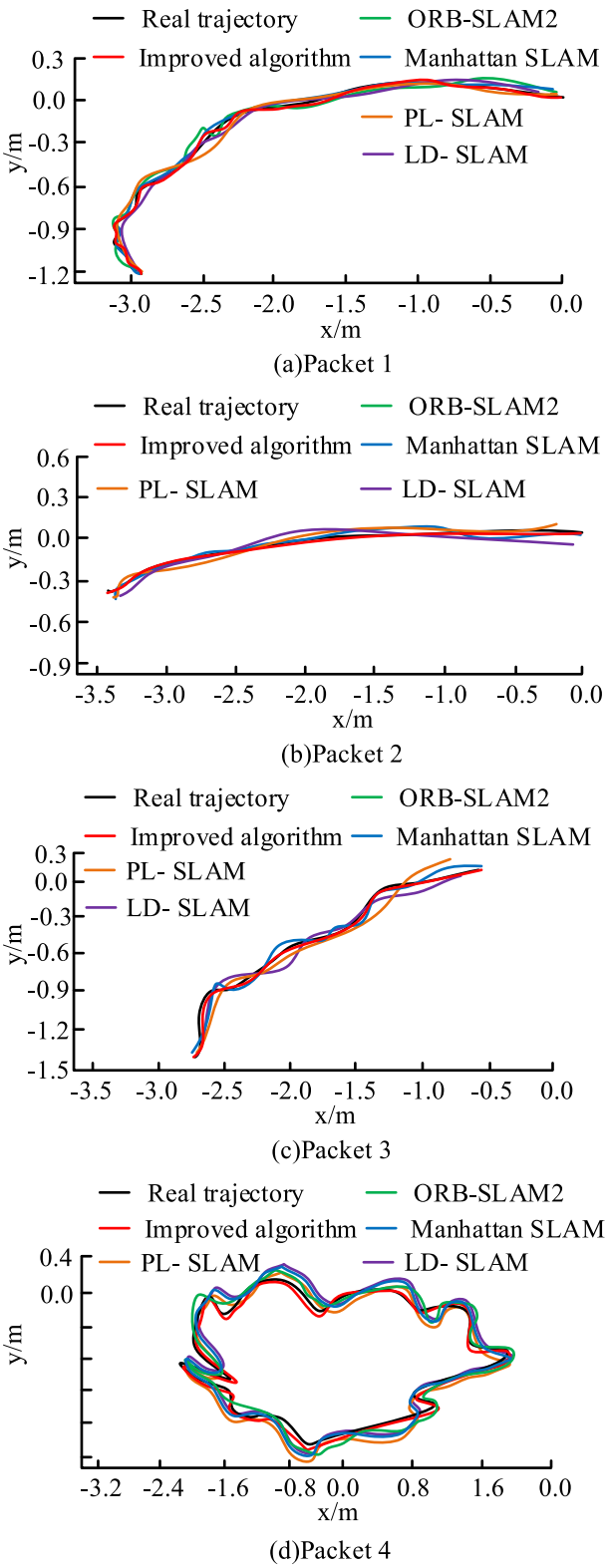


FIGURE 7. The trajectory of the algorithm on the Tum dataset.

TABLE 1. Comparison between the RMS deviation of the algorithm's position on the Tum dataset and the number of MF.

Data packet	Duration/s	Improvement algorithm RMSE/m	ORB-SLAM2 RMSE/m	Manhattan SLAM RMSE/m	Improvement algorithm MF	Manhattan SLAM MF
Packet 1	31.66	0.011	0.011	0.021	636	589
Packet 2	36.98	0.009	0.011	0.012	572	567
Packet 3	27.42	0.019	/	0.045	643	688
Packet 4	36.51	0.023	/	0.026	686	795
Packet 5	34.08	0.048	/	0.075	468	195
Packet 6	26.58	0.012	0.011	0.011	5	4
Packet 7	20.41	0.034	0.028	0.032	0	2
Packet 8	122.28	0.003	0.009	0.012	116	0
Packet 9	98.83	0.025	0.029	0.027	44	26

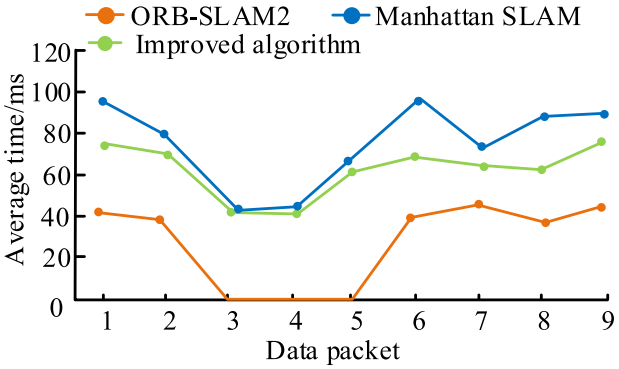


FIGURE 8. Comparison of average time for processing one frame of image using algorithms on the Tum dataset.

of algorithms processing one frame of image on the Tum dataset is shown in Figure 8.

In Figure 8, the average processing time of the ORB-SLAM2 algorithm for one frame of image is 41.1ms, the average processing time of the Manhattan SLAM algorithm for one frame of image is 75.3ms, and the average processing time of the improved algorithm for one frame of image is 62.1ms. The reason why the ORB-SLAM2 algorithm has less running time is that it does not extract and process line and surface features. The improved algorithm's one frame image processing time is 13.2ms less than the Manhattan SLAM algorithm, which improves the real-time performance of the algorithm by about 17.53%. For further validating the effectiveness of the algorithm, this study carried comparative experiments using a conventional indoor dataset. The comparison of robot trajectories under different algorithms in the conventional indoor dataset is shown in Figure 9.

Figure 9 demonstrates that the trajectory estimation of the improved ORB-SLAM2 algorithm is closer to reality, indicating that the algorithm has stronger tracking authenticity and improves the tracking ability of visual robots. The algorithm accuracy is 9.71% and 15.76% higher than the traditional ORB-SLAM2 algorithm and Manhattan SLAM algorithm, respectively.

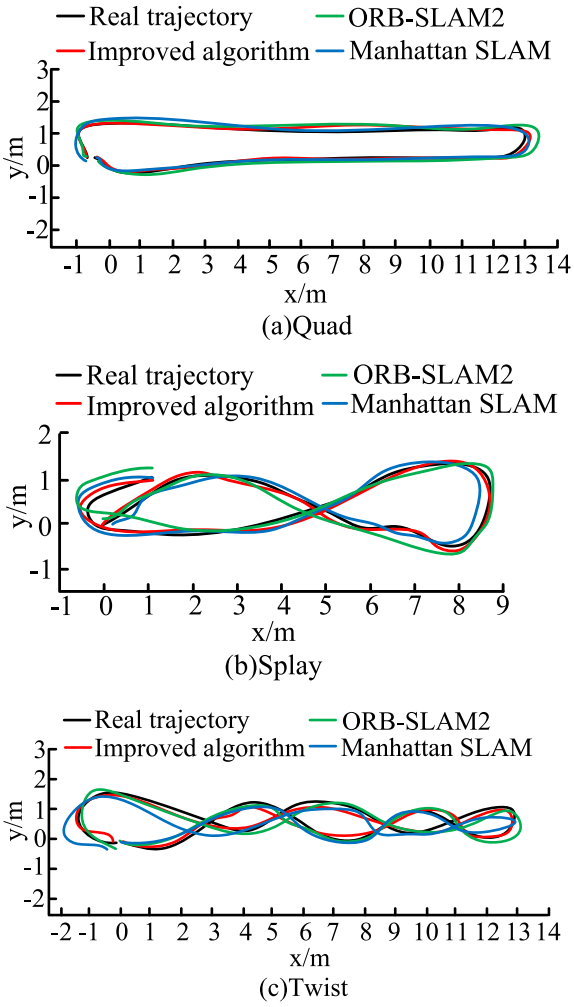


FIGURE 9. Comparison of robot trajectories under different algorithms in conventional indoor datasets.

B. EXPERIMENTAL ANALYSIS OF AN IMPROVED ALGORITHM IN VIEW OF MULTI-SENSOR FUSION

This study used slip experiments for verifying the slip detection effect of the improved algorithm in view of multi-sensor fusion (referred to as the multi-sensor improved algorithm). The trajectory and displacement comparison of the improved algorithm with and without slip detection are shown in Figure 10.

Figure 10 (a) shows a comparison of the trajectories of the improved multi-sensor algorithm with and without slip detection. It indicates that the trajectories of the improved multi-sensor algorithm with slip detection have a high degree of overlap with the real trajectories, while the latter half

TABLE 2. Comparison of position RMS deviation between two improved algorithms.

Data packet	Duration/s	Multi feature improvement algorithm RMSE/m	With closed-loop detection RMSE/m	No closed-loop detection RMSE/m
Packet 1	116.42	0.142	0.137	0.163
Packet 2	67.75	0.159	0.152	0.174
Packet 3	139.61	0.164	0.174	0.179
Packet 4	142.46	0.161	0.153	0.176
Packet 5	124.92	0.172	0.140	0.166
Packet 6	105.11	0.211	0.196	0.231

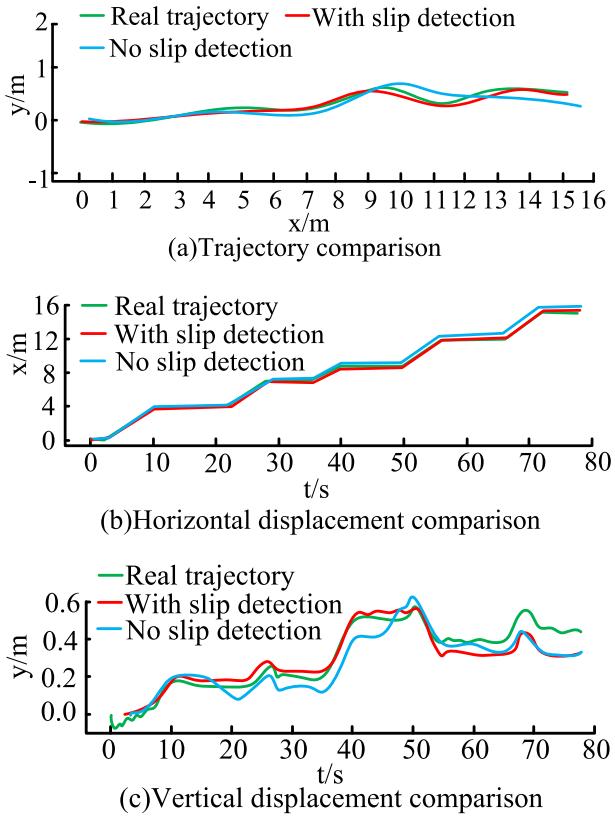


FIGURE 10. Comparison of trajectory and displacement of improved multi-sensor algorithms with and without slip detection.

of the trajectories of the improved multi-sensor algorithm without slip detection have poor authenticity. Figure 10 (a) shows the comparison of horizontal displacement between the improved multi-sensor algorithm with and without slip detection. It shows that the difference between the horizontal displacement of the improved multi-sensor algorithm with slip detection and the actual trajectory displacement is less than 0.1m, and the maximum difference between the horizontal displacement of the improved multi-sensor algorithm without slip detection and the actual trajectory is 0.5m. Figure 10 (a) shows a comparison of the vertical displacement of the improved multi-sensor algorithm with and without slip detection. It can be seen that the vertical displacement of the improved multi-sensor algorithm with slip detection has a higher overlap with the actual trajectory displacement; This indicates that improved algorithms with slip detection

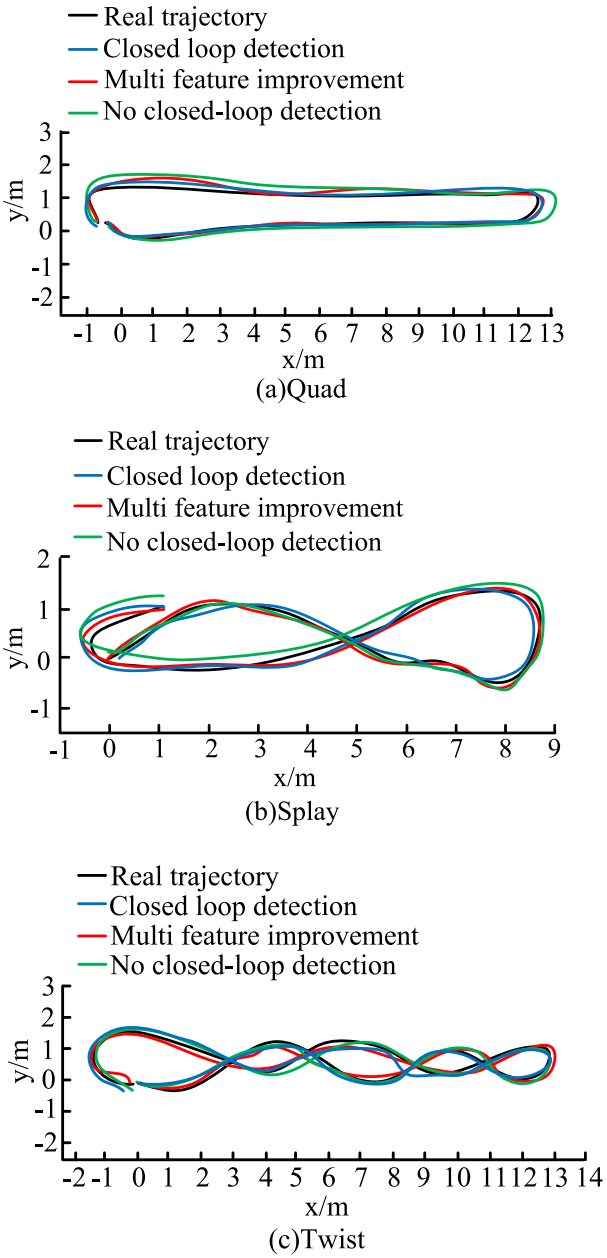


FIGURE 11. Comparison of trajectories between two improved algorithms on conventional indoor datasets.

can improve positioning accuracy by reducing wheel slip. To further validate the effectiveness of the improved multi-sensor algorithm, the experiment compared the improved

TABLE 3. Comparison of runtime between two improved algorithms.

Data packet	Multi feature improvement algorithm run time/ms	With closed-loop detection run time/ms	No closed-loop detection run time/ms
Packet 1	60.7	96.6	72.1
Packet 2	61.5	88.3	69.9
Packet 3	68.2	98.2	75.6
Packet 1-3 mean	63.5	94.4	72.5
Packet 4	23.6	35.5	33.2
Packet 5	25.2	37.3	34.9
Packet 6	22.5	31.8	33.4
Packet 4-6 mean	23.8	34.9	33.8

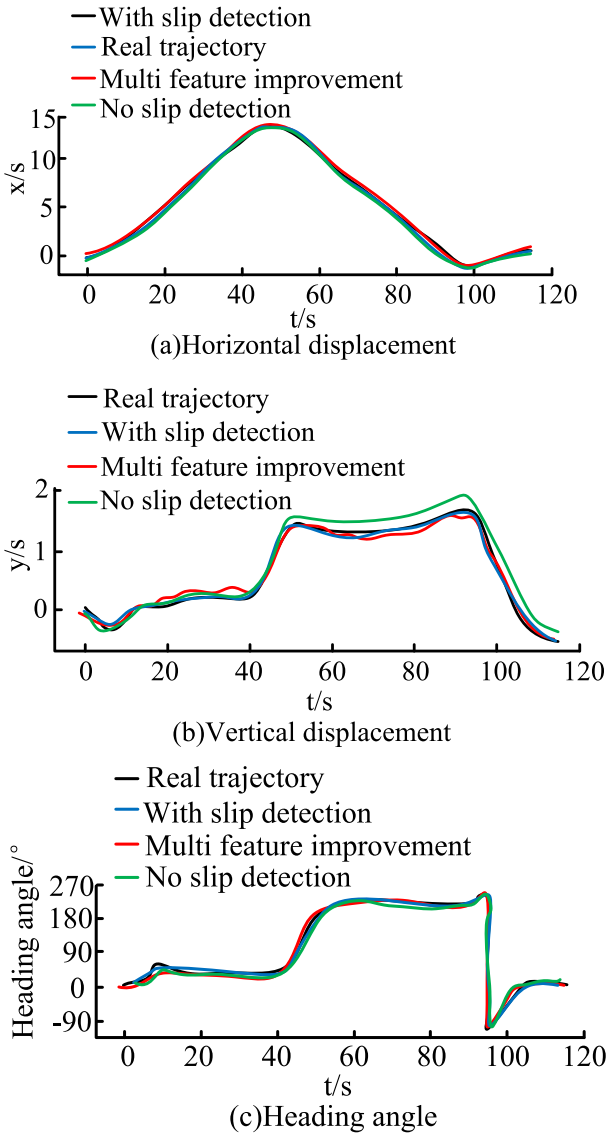


FIGURE 12. Comparison of displacement and heading angle between two improved algorithms in conventional dataset Quad packets.

ORB-SLAM2 algorithm in view of MFF (referred to as the multi feature improvement algorithm). The trajectory comparison of the two improved algorithms on a conventional indoor dataset is shown in Figure 11.

As can be seen in Figures 11 (a) and (b), under the Quad and Splay data packets, the trajectories of the multi feature improvement algorithm and the rated multi sensor improvement algorithm with CL detection are very close to the real trajectories; This indicates that the cumulative error of these two algorithms is small and can improve the accuracy of tracking. In Figure 11 (c), the multi-sensor improved algorithm did not detect a large CL and could not reduce cumulative errors. The multi feature improved algorithm is similar to the real trajectory; This indicates that the introduction of MF can reduce the cumulative error of multi feature improvement algorithms, thereby improving the accuracy of trajectory estimation. The comparison of displacement and heading angle between the two improved algorithms in the conventional dataset Quad data package is shown in Figure 12.

Figures 12 (a) and (b) show the comparison of horizontal and vertical displacement of the algorithm. It can be seen that after 40 seconds, the maximum vertical displacement error of the improved multi-sensor algorithm without CL detection is 0.05m, and CL optimization can eliminate this error. Compared to the multi sensor improvement algorithm and multi feature improvement algorithm without CL detection, the multi sensor improvement algorithm with CL detection has more accurate estimation of the heading angle, improving the accuracy of the robot's heading. The comparing of the RMS deviation of the two improved algorithms is shown in Table 2.

In Table 2, packets 1 to 3 are regular indoor dataset packets, while packets 4 to 5 are feature sparse indoor dataset packets. Under the data package of conventional indoor data set, the minimum AEV of the position RMS deviation of the improved multi-sensor algorithm with CL detection is 0.154m, and the AEV of the position RMS deviation of the improved multi-sensor algorithm with multiple features and without loop detection is 0.155m and 0.172m, respectively. Compared to the multi feature improvement algorithm and the multi sensor improvement algorithm for CL detection, the accuracy of the CL detection multi sensor improvement algorithm has increased by 0.655% and 10.47%, respectively. The comparing of the running time of the two improved algorithms is demonstrated in Table 3.

In Table 3, the average running times of the multi feature improvement algorithm, the multi sensor improvement algorithm with CL detection, and the multi sensor improvement algorithm without CL detection under conventional indoor dataset data packets are 63.5ms, 94.4ms, and 72.5ms, respectively. The average running time of the multi feature

improvement algorithm, the multi sensor improvement algorithm with CL detection, and the multi sensor improvement algorithm without CL detection under coefficient indoor dataset data packets is 23.8ms, 34.9ms, and 33.8ms, respectively. Compared with experimental data, it demonstrates that the multi feature improvement algorithm possesses strong real-time performance, and the multi sensor improvement algorithm with CL detection has a larger computational load and relatively longer running time.

V. CONCLUSION

With the development of technology and the popularization of robotics, people are gradually coming into contact with wheeled robots. In response to the issue of tracking loss in traditional algorithms, this study proposes an improved ORB-SLAM2 MR vision algorithm in view of MFF. It integrates point, line, and surface features on the basis of traditional algorithms, and proposes an improved algorithm in view of multi-sensor fusion to improve the accuracy of robot tracking. The experimental data shows that the AEV of the position RMS deviation of the improved ORB-SLAM2 MR vision algorithm in view of MFF is 0.02 in nine data packets of the Tum dataset; The AEV of the RMS deviation of the corresponding ORB-SLAM2 algorithm tracking successful packets is 0.016, which increases the average accuracy by 11.11% and 31.03% compared with the average accuracy of the Manhattan SLAM algorithm. The improved ORB-SLAM2 MR vision algorithm in view of MFF has a running time of 13.2ms less than the Manhattan SLAM algorithm in one frame image processing, which enhances the real-time performance of the algorithm by approximately 17.53%. The disparity in the horizontal displacement and the actual trajectory displacement of the improved multi-sensor algorithm with slip detection is less than 0.1m, while the maximum disparity in the horizontal displacement and the actual trajectory of the improved multi-sensor algorithm without slip detection is 0.5m; This indicates that improved algorithms with slip detection can improve positioning accuracy by reducing wheel slip. Compared to the improved ORB-SLAM2 MR vision algorithm in view of MFF and the improved multi-sensor algorithm for CL detection, the accuracy of the improved multi-sensor algorithm for CL detection has increased by 0.655% and 10.47%, respectively. The improved ORB-SLAM2 MR vision algorithm in view of MFF has strong real-time performance, and the improved multi-sensor algorithm with CL detection has a large computational load and relatively long running time. The introduction of line and surface features in this study will increase computational complexity, and future research can optimize them for further enhancing the real-time performance of the algorithm.The improved multi feature fusion algorithm proposed in the study can accurately estimate trajectories in conventional indoor scenes, with good reconstruction results. It is suitable for conventional indoor scenes, such as industrial production detection, warehousing robots, and other fields. However, in sparse features and outdoor scenes, there is a decrease in accuracy and tracking loss. The improved multi-sensor fusion algorithm has achieved high accuracy and robustness in both indoor and outdoor scenarios, and is suitable for both indoor and outdoor scenarios. It has potential and advantages in industrial automation applications.

REFERENCES

[1] A. Filatov and K. Krinkin, "A simplistic approach for lightweight multi-agent SLAM algorithm," *Int. J. Embedded Real-Time Commun. Syst.*, vol. 11, no. 3, pp. 67–83, Jul. 2020, doi: 10.4018/IJERTCS.2020070104.

[2] Z. Liu, X. Liu, Z. Cao, X. Gong, M. Tan, and J. Yu, "High precision calibration for three-dimensional vision-guided robot system," *IEEE Trans. Ind. Electron.*, vol. 70, no. 1, pp. 624–634, Jan. 2023, doi: 10.1109/TIE.2022.3152026.

[3] J. Goodwin and C. Saldaña, "Vision-based localization for cooperative robot-CNC hybrid manufacturing," *Int. J. Adv. Manuf. Technol.*, vol. 126, nos. 1–2, pp. 241–258, Feb. 2023, doi: 10.1007/s00170-023-11009-9.

[4] L.-H. Juang, "Humanoid robot fetching objects using monocular vision unit," *Multimedia Tools Appl.*, vol. 82, no. 5, pp. 6747–6767, Feb. 2023, doi: 10.1007/s11042-022-13602-8.

[5] P. Zhou, C. Zhang, G. Zhou, Z. He, X. Yan, S. Wang, M. Sun, and B. Hu, "Whole surface defect detection method for bearing rings based on machine vision," *Meas. Sci. Technol.*, vol. 34, no. 1, pp. 15017–15033, Jan. 2023, doi: 10.1088/1361-6501/ac9993.

[6] W. Zhang, Y. Du, and Q. Bai, "An optimized coverage robot SLAM algorithm based on improved particle filter for WSN nodes," *Int. J. Grid High Perform. Comput.*, vol. 12, no. 4, pp. 76–88, Oct. 2020, doi: 10.4018/IJGHPC.2020100106.

[7] B. Fang, X. Han, Z. Wang, and X. Yuan, "SLAM algorithm based on bounding box and deep continuity in dynamic scene," *Int. J. Wireless Mobile Comput.*, vol. 21, no. 4, pp. 349–364, Mar. 2021, doi: 10.1504/IJWMC.2021.121626.

[8] F. Hu, J. Cheng, Y. Bao, and Y. He, "Accuracy enhancement for the front-end tracking algorithm of RGB-D SLAM," *Intell. Service Robot.*, vol. 13, no. 2, pp. 207–218, Apr. 2020, doi: 10.1007/s11370-019-00299-2.

[9] J. Dong, Y. Jiang, and Z. Han, "Optimization algorithm of RGB-D SLAM visual odometry based on triangulation," *Int. J. Performability Eng.*, vol. 16, no. 3, pp. 438–445, Mar. 2020, doi: 10.23940/ijpe.20.03.p13.438445.

[10] Y. Su and L. Yu, "A dense RGB-D SLAM algorithm based on convolutional neural network of multi-layer image invariant feature," *Meas. Sci. Technol.*, vol. 33, no. 2, pp. 25402–25421, Dec. 2021, doi: 10.1088/1361-6501/ac38f1.

[11] B. Li, Y. Yang, C. Qin, X. Bai, and L. Wang, "Improved random sampling consensus algorithm for vision navigation of intelligent harvester robot," *Ind. Robot, Int. J. Robot. Res. Appl.*, vol. 47, no. 6, pp. 881–887, Aug. 2020, doi: 10.1108/IR-03-2020-0055.

[12] S. Liang, C. Chen, and G. Zou, "Intelligent driving system of robot based on computer vision and neural network algorithm," *J. Intell. Fuzzy Syst.*, vol. 38, no. 6, pp. 7279–7290, Jun. 2020, doi: 10.3233/JIFS-179803.

[13] B. Cui, P. Chen, and W. Zhang, "Machine vision based edge detection method for toilet seat," *Int. J. Model., Identificat. Control*, vol. 35, no. 4, pp. 338–345, May 2021, doi: 10.1504/IJMIC.2020.114790.

[14] T. Liu, H. Liu, Y.-F. Li, Z. Chen, Z. Zhang, and S. Liu, "Flexible FTIR spectral imaging enhancement for industrial robot infrared vision sensing," *IEEE Trans. Ind. Informat.*, vol. 16, no. 1, pp. 544–554, Jan. 2020, doi: 10.1109/TII.2019.2934728.

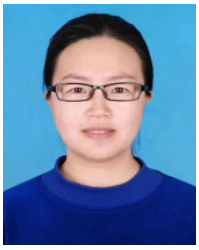
[15] E. Cristofalo, E. Montijano, and M. Schwager, "Vision-based control for fast 3-D reconstruction with an aerial robot," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 4, pp. 1189–1202, Jul. 2020, doi: 10.1109/TCST.2019.2905227.

[16] Y. Meng, Z. Wu, Y. Li, D. Chen, M. Tan, and J. Yu, "Vision-based underwater target following control of an agile robotic manta with flexible pectoral fins," *IEEE Robot. Autom. Lett.*, vol. 8, no. 4, pp. 2277–2284, Apr. 2023, doi: 10.1109/LRA.2023.3250004.

[17] L. Xing, "Research on the positioning and recognition of an intelligent inspection robot in substations," *Int. J. Autom. Technol.*, vol. 17, no. 1, pp. 65–70, Jan. 2023, doi: 10.20965/ijat.2023.p0065.

[18] D. K. Ko, K. W. Lee, D. H. Lee, and S. C. Lim, "Vision-based interaction force estimation for robot grip motion without tactile/force sensor," *Expert Syst. Appl.*, vol. 211, pp. 118441–118452, Jan. 2023, doi: 10.1016/j.eswa.2022.118441.

[19] L. Xu, H. Yin, T. Shi, D. Jiang, and B. Huang, "EPLF-VINS: Real-time monocular visual-inertial SLAM with efficient point-line flow features," *IEEE Robot. Autom. Lett.*, vol. 8, no. 2, pp. 752–759, Feb. 2023, doi: 10.1109/LRA.2022.3231983.
[20] S. Liang, W. Zhu, C. Chakraborty, J. Du, and K. Yu, "The IMU augment SLAM on unmanned vehicle for detection of protective measures in COVID-19," *IEEE Sensors J.*, vol. 23, no. 2, pp. 933–946, Jan. 2023, doi: 10.1109/JSEN.2022.3189033.
[21] Y. Gao, D. Shi, R. Li, Z. Liu, and W. Sun, "Gyro-Net: IMU gyroscopes random errors compensation method based on deep learning," *IEEE Robot. Autom. Lett.*, vol. 8, no. 3, pp. 1471–1478, Mar. 2023, doi: 10.1109/LRA.2022.3230594.
[22] J. Zan, "Research on robot path perception and optimization technology based on whale optimization algorithm," *J. Comput. Cogn. Eng.*, vol. 1, no. 4, pp. 201–208, Mar. 2022, doi: 10.47852/bonviewJCCE597820205514.



XIAOMEI HU was born in March 1985, in Qixia, Shandong, Han nationality. She received the bachelor's degree in automation from Qufu Normal University, in 2008. She is currently pursuing the master's degree in engineering management with the Liaoning University of Technology. She has been a Teacher with Yantai Nanshan University, since 2008. She holds two authorized invention patents. At present, it has accepted two invention patents and published three academic articles in CSCD core journals.

LUYING ZHU was born in October 1982, in Longkou, Shandong, Han nationality. She received the bachelor's degree in automation from Qingdao University, in 2005, and the M.E. degree in signal and information processing from Yantai University, in 2009. From July 2009 to September 2011, she was with Yantai Sanhuan Technology Company Ltd., engaged in product development. Since October 2011, she has been with the Teaching and Research Department, Yantai Nanshan University, as the Director. She has published six academic articles, accepted two CSCD core articles, participated in one authorized invention patent, and declared and accepted five invention patents.

PING WANG was born in January 1977, in Qingdao, Shandong, Han nationality. She received the master's degree in computer science from Qingdao University, in 2009. Since August 2000, she has been with Yantai Nanshan University. She has published more than 30 articles and four textbooks and monographs. She has presided over or participated in more than 20 teaching and scientific research projects and authorized five patents. She has won one third prize of the Shandong Province Teaching Achievement Award.

HAILI YANG was born in August 1980, in Taiyuan, Shanxi, Han nationality. She received the bachelor's degree in power engineering from Shanxi University, in 2005. Since 2008, she has been with Shandong Yili Electric Company Ltd. She has published seven papers and authorized one utility model patent.

XUAN LI was born in June 1985, in Jining, Shandong, Han nationality. He received the degree from Yantai Nanshan University, in 2008. Since 2008, he has been with KONE Auto Parts Company Ltd.