ISSN: 2664-9640

Research on Monocular SLAM Algorithm Based on Feature Method and Direct Method

Qian Sun [1], Xiaohong Ren [1], Yingao Yue [2]

[1]College of Automation and Information Engineering, Sichuan University of Science & Engineering, Zigong, 644000, China;

[2]School of Computer Engineering, Hubei University of Arts and Science, Xiangyang, 441053, China

Abstract

Direct Sparse Odometry (DSO) is a visual odometry method that can directly collect pixels from an image environment with intensity gradients, because it does not rely on key point detectors or descriptors, making it suitable for low texture The image has the advantages of high robustness and fast speed. However, for indoor environments, DSO is susceptible to changes in illumination because it is based on the premise that the gray level is unchanged. At the same time, the camera moves too fast, which causes the pixel acquisition failure, resulting in poor stability of the algorithm. In order to improve the applicability and robustness of the direct sparse odometer, this paper proposes a visual odometer method using the feature point method and the direct method fusion. This method uses the direct method for local mapping to track the camera pose and combines the marginal data information with the feature point method. The information extracted by the feature point method is combined with the camera pose information to optimize the key frames of the local map. At the same time, the feature point information is used for Loop detection and build a global map. Verification by EuRoC data set: The method proposed in this paper has improved stability and accuracy.

Keywords

Visual positioning; direct method; feature point method; algorithm fusion.

1. Introduction

With the rapid development of information technology, simultaneous localization and mapping (SLAM) has become a key technology for applications such as robots, drones, and autonomous driving [1]. SLAM technology is to perceive the surrounding environment through the equipped sensors, such as lidar, camera, depth camera, etc., use the corresponding SLAM algorithm to calculate and track its position and posture in the environment in real time, and at the same time construct environmental map information. Among them, SLAM (Vision-based SLAM, VSLAM) based on vision sensors has become a research hotspot of scholars due to its advantages in capturing environmental information and strong real-time performance [2].

The feature point method has long occupied the mainstream position in visual odometry. It solves the camera's pose and scene information by matching features and performing geometric bundle adjustment (BA) that minimizes the reprojection error (Reprojection error). Davison A J et al. [3] first proposed a real-time monocular vision system MonoSLAM in 2007. Its front end extracts a small number of feature points, and the back end uses EKF filtering to optimize, thereby reducing the amount of calculation. It uses the map point and the pose as the state of the filter at the same time, and improves the robustness of the system by updating the covariance. Klein G et al. [4] proposed one of the most representative systems in VSLAM, PTAM, in 2007. He initially proposed dividing tracking and mapping into two parallel threads.

ISSN: 2664-9640

Different from the previous filtering method, PTAM uses beam adjustment (BA) at the back end of SLAM to globally optimize map points and multi-frame cameras, which further improves the accuracy of pose estimation. Mur-Artal R et al. [5], following PTAM in 2015, proposed ORB-SLAM, one of the most advanced and robust VSLAM systems so far. It uses three threads in parallel for the first time: real-time tracking threads and local optimization threads. And global loopback detection and optimization thread. This multi-threaded parallel mode not only greatly improves the accuracy of the system, but also ensures the real-time performance of the system. However, the VSLAM system based on feature points has obvious shortcomings [6]. For example, the extraction and matching of feature point information requires a lot of calculation, which is time-consuming and serious, and it leads to tracking in the case of low texture or too few feature points. failure. In contrast, the direct method calculates the camera movement based on the gradient intensity changes of the image pixels in the environment. VSLAM based on the direct method started late. Newcombe, Richard A et al. [7] first proposed the VSLAM system DTAM based on the direct method in 2011. Based on the assumption of constant luminosity, all pixels of the image are directly matched and optimized, and The inverse depth filtering method is used to construct dense maps, but due to the large amount of calculation required, GPUs are generally only used for calculation. Engel J et al. [8] in 2014, in order to achieve the purpose of applying the direct method to a wider range of cameras, they proposed the LSD-SLAM for semi-dense map construction, which only selects pixels with sharp gradient changes in the image for multi-pyramid Direct matching and tracking of the shape, this method reduces the amount of calculation and improves its positioning accuracy. Subsequently, Engel J et al. [9] proposed a more robust, more accurate, and faster sparse direct method DSO in 2017. Compared with the previous LSD-SLAM, the backend uses a sliding window and marginalization. The strategy greatly improves the real-time and accuracy of the direct method. Since the direct method is sensitive to ambient light, it is necessary to maintain a low-speed uniform motion as much as possible. Therefore, many scholars have proposed semi-direct methods, such as the SVO proposed by Forster C et al. [10] that uses tightly coupled fusion based on the direct method and feature point method. This method uses gray value matching to directly obtain the camera pose and adopts Geometric BA refines the posture and structure, which improves the accuracy and robustness to a certain extent.

This paper finds that the direct method has obvious advantages over the feature point method in low texture and speed, but it is more sensitive to light changes, and it is easy to affect its stability due to light. Moreover, the key points in the direct method have no description information and cannot be used for long distances. Tracking and positioning. Therefore, this paper proposes to merge the direct method with the feature point method, and the main contributions are as follows:

1) The direct method is used to perform local mapping and tracking the camera pose, and the collected marginal data is combined with the feature point method to make full use of the local robustness and high accuracy of the direct method to improve the accuracy of global mapping.

2) Use the feature point method to optimize the pose information of the key frame, and use the key point information provided by the feature point method to match to obtain the environment map data, combined with loop detection to construct a global map, which effectively solves the loss of the direct method due to the camera's fast motion tracking The problem.

2. Algorithm theory and implementation

2.1. Algorithm framework

The monocular SALM algorithm system framework based on the direct method and the feature point method fusion proposed in this paper is shown in Figure 1. The new image frame is

ISSN: 2664-9640

quickly tracked by the direct method, the photometric error between the image frames is calculated and the depth map is constructed, and adopted The marginalization method based on Shuer's complement removes the key frames and image points that have little contribution, builds a local map and provides initial information for the feature method to build a global map: key frame image information and pose information; key frame information provided by the direct method The feature points are extracted and matched, the initial pose estimation and correction are combined with the camera pose information, and the loop detection and beam adjustment (BA) are used for optimization, and a high-precision global map is calculated.

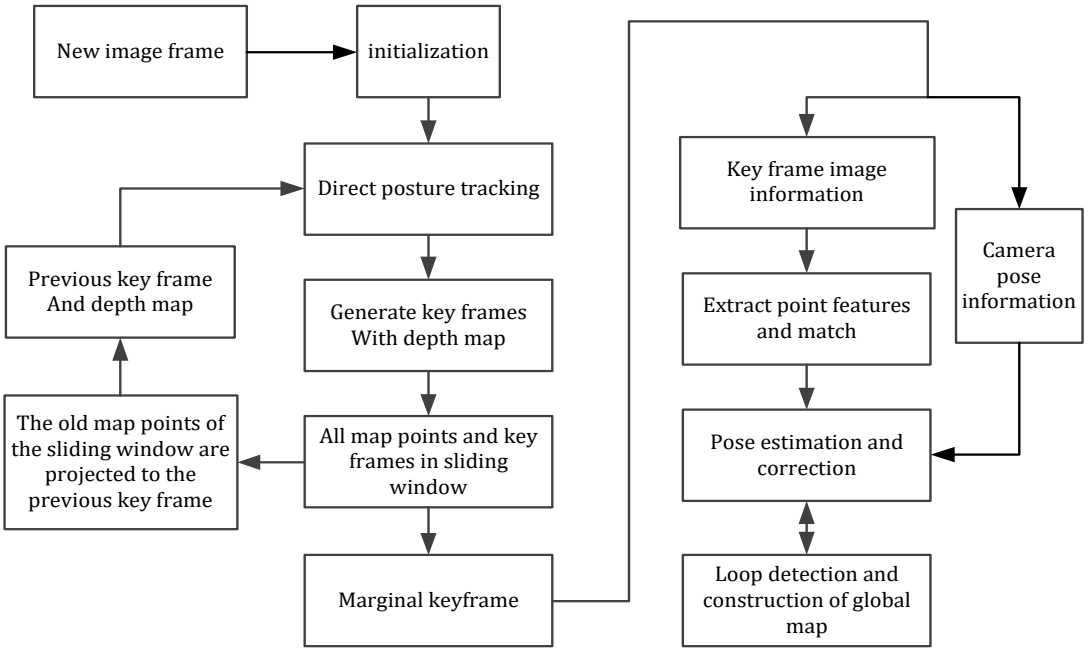


Figure 1: Algorithm theory framework

2.2. Local map construction based on direct method

The local mapping part based on the direct method in this paper mainly refers to the DSO proposed by Engel J et al. [8], which can be divided into three parts: camera luminosity calibration, sliding window optimization and marginalization.

2.2.1. Camera photometric calibration

For robots that rely on vision algorithms to build maps, the camera's photometric calibration is particularly important. In this paper, the camera's photometric calibration uses the method in the paper [11], and its expression is:

$$I_i(x) = G(t_i V(x) B_i(x)) \tag{1}$$

$$I_i'(x) := t_i B_i(x) = \frac{G^{-1}(I_i(x))}{V(x)} \tag{2}$$

Where G is the nonlinear response function $G: R \to [0,255]$, V is the normalized vignetting function, B_i is the irradiance observed in frame i, I_i is the image obtained in frame i of the camera, and t_i is the exposure time. For the rest, I_i is the image I_i' after photometric calibration.

2.2.2. Sliding window optimization

When the point p in the reference frame I_i is recognized in the next key frame I_j, the position and posture estimation is calculated and adjusted on the pixel N_p of the 8-point area according

ISSN: 2664-9640

to the corresponding luminosity error of the point, so the camera posture estimation in the two frames before and after The error function is:

$$E_{ij}^{p} := \sum_{p \in N_p} \omega_p \left\| \left(I_j[p'] - b_j\right) - \frac{t_j e^{\alpha_j}}{t_i e^{\alpha_i}} \left(I_i[p] - b_i\right) \right\|_{\gamma} \tag{3}$$

Where ω_p is to use a constant c to reduce the weight of pixels with high pixel gradient in the total error function, the expression is:

$$\omega_p := \frac{c^2}{c^2 + \left\|\nabla I_i(\tilde{p})\right\|_2^2} \tag{4}$$

Where p' is the projection point of point p, and its pose is the rotation matrix R and the translation vector t describing the rotation of the camera according to the point p and the inverse depth value d_p, and then calculated according to formula (5), the expression is as follows:

$$p' = \prod_c \left(R \prod_C^{-1}\left(P, d_p\right) + t\right) \tag{5}$$

$$\begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} := T_i^{-1} T_j \tag{6}$$

In the formula, t_i and t_j are the exposure time of images I_i and I_j, and $\|\cdot\|_{\gamma}$ is the Huber norm.

After calculating the photometric error between the two images, the complete photometric error function expression of all frames and points is:

$$E_{photot} := \sum_{i \in F} \sum_{p \in F} \sum_{j \in obs(p)} E_{ij}^{p} + \sum_{i \in F} \left(\lambda_a a_i^2 + \lambda_b b_i^2\right) \tag{7}$$

In the formula, a and b are brightness parameters, and F is the collection of images in the sliding window. Among them, when the exposure time is known, λ_a and λ_b will be set to fixed values, and when the exposure time is unknown, set $\lambda_a = \lambda_b = 0$ and $t_i = t_j = 1$. Using Levenberg's method to solve equation (7) iteratively, the update equation is:

$$\delta\xi = -\left(J^T W J\right)^{-1} J^T W_r$$
$$\text{and} \quad \xi^{new} \leftarrow \delta\xi \boxplus \xi \tag{8}$$

In the formula, r is the vector residual, J is the corresponding Jacobian matrix, and W is the weight matrix. The state variable ξ includes all variables in the sliding window, such as camera internal parameters, photometric calibration parameters, inverse depth value and camera pose.

2.2.3. Marginalization

In this paper, a marginalization method based on Shuer's complement [9] is used to remove map points and key frames that contribute little. The purpose of the marginalized key frame is to remove the old state quantity from the optimization window to ensure operating efficiency; at the same time, the key frame information of the removed state quantity is retained and provided to the feature point method for extracting position information, which not only avoids To avoid information loss and improve the accuracy of pose estimation, the specific usage rules are as follows:

1) Keep the latest two key frames;

2) Image frames with less than 5% of the common viewpoint with the latest key frame will be eliminated;

ISSN: 2664-9640

3) When the number of key frames in the sliding window is more than the set threshold, the frames with the larger "distance" value will be eliminated.

3. Global map construction based on feature point method

The global mapping part based on the feature point method in this paper mainly refers to the ORB-SLAM proposed by Mur-Artal R et al. [5], which can be divided into three parts: initial pose estimation, pose estimation and correction, and loop detection.

3.1. Initial pose estimation

The scale factor of the initial pose calculated by the direct method is inconsistent with the scale factor of the estimated pose based on the feature. The mapping scale will gradually deviate over time and will not converge to the same value. Therefore, we use the most recent 30 key frames in the feature point method and the corresponding key frames in the direct method image sequence to calculate the relative scale factor through $Sim(3)$ alignment [11]. According to the relative scale factor s, the relative ratio of the pose estimation between the corresponding key frames in the direct method calculation is appropriately scaled and used for the pose calculation based on the feature point method. Let i and j be the previous and current key frames, the calculation expression is:

$$T_{j\omega|F} = \begin{bmatrix} R_{ji|F} & t_{ji|F} \\ 0_{1\times 3} & 1 \end{bmatrix} T_{i\omega|F} \tag{9}$$

$$R_{ji|F} = R_{ji|D} = R_{j\omega|D}\left(R_{i\omega|D}\right)^{T} \tag{10}$$

$$t_{ji|F} = st_{ji|D} = s\left(-R_{ji|D}t_{i\omega|D} + t_{j\omega|D}\right) \tag{11}$$

In the formula, if the relative ratio s is obtained, the increment in the direct method can be scaled and used for the initial pose guessing in the feature point method. D and F are the pose estimation based on the direct method and the feature point method respectively.

3.2. Pose estimation and correction

Based on the direct method, the initial pose estimation value of the new key frame is provided to the feature point method, combined with the geometric BA to optimize the local feature map, the expression of the total function is:

$$E_{reproj} = \sum_{i\in F_{local}} \sum_{x\in p_i} \sum_{j\in obs(x)} \left\| \frac{P_{j,x} - \prod_c \left(T_{i\omega} X_{\omega}\right)}{\sigma_x^2} \right\|_{\gamma} \tag{12}$$

among them

$$\sigma_x^2 := \left(\lambda_{pyr}\right)^{2L_{pyr,x}} \tag{13}$$

This function is composed of the normalized reprojection error of the variance of the local map points, where F_{local} represents the set of all local key frames, $P_{j,x}$ represents the match with the key point x in the key frame j, and σ_x^2 represents the variance of the feature position in the frame i, λ_{pyr} represents the constant scale factor of the image pyramid, which is always greater than 1, and $L_{pyr,x}$ represents the pyramid level of the detected key points.

3.3. Loop detection

In the loop detection, we use the DBoW2 visual bag of words construction to traverse the key frame database to detect larger closed loops. Once the loop is detected, the key frames and map points are aligned and merged. In order to correct the elegant scale, the pose map of the trajectory is optimized, and its expression is:

ISSN: 2664-9640

$$E_{graph} = \sum_{(i,j)\in\varepsilon_{edge}} \left\| \log_{Sim(3)}\left(S_{ij,0} S_{j\omega} S_{i\omega}^{-1}\right) \right\|_2^2 \tag{14}$$

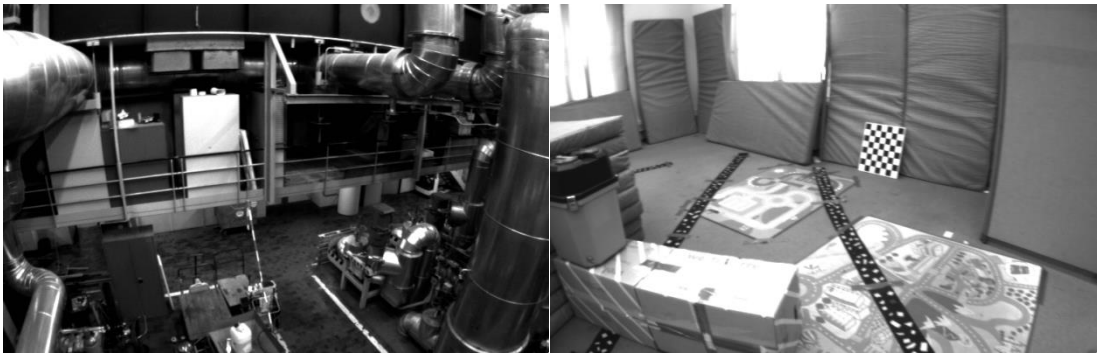
ε_{edge} represents a set of edges of the original basic graph, and $S_{ij,0}$ represents the similarity conversion between the i -th frame and the j -th frame before the optimization of the pose graph. After the closed loop is detected, the global BA optimization is performed immediately.

4. Experimental design and result analysis

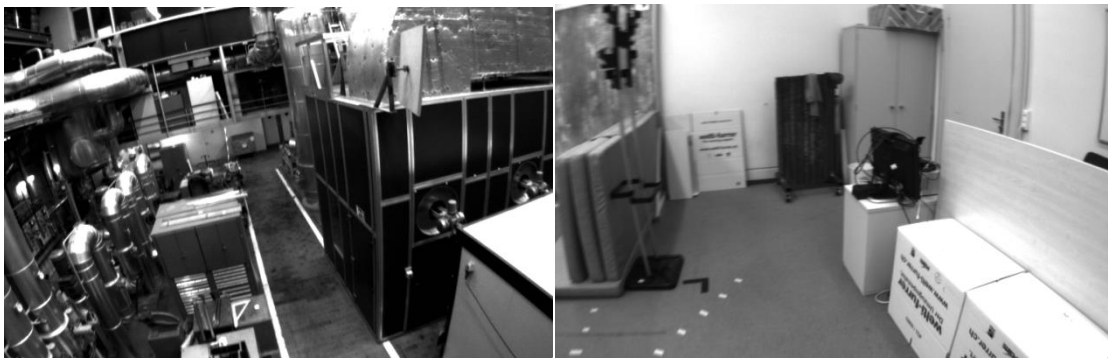
The algorithm running platform in this paper is configured with an Inter i5-7300HQ CPU, a main frequency of 2.50GHz, a memory of 8G, no GPU acceleration, the system is Ubuntu16.04, and the Ros version is Kinect.

4.1. Public data set selection

This article uses the public EuRoC data set for simulation experiments. It uses Skybotix VI sensors to provide a series of monocular and binocular camera image sequences. The true path of the data set is obtained by Leica MS50 lidar. The data set contains two scenes, an industrial plant and a general indoor scene, as shown in Figure 2. It is divided into three categories: easy, meidum, and difficult according to the environmental feature texture, light, movement speed and image quality. The left is the industrial plant. , The right is a normal indoor scene. This article is tested on 3 data sets of different difficulty.

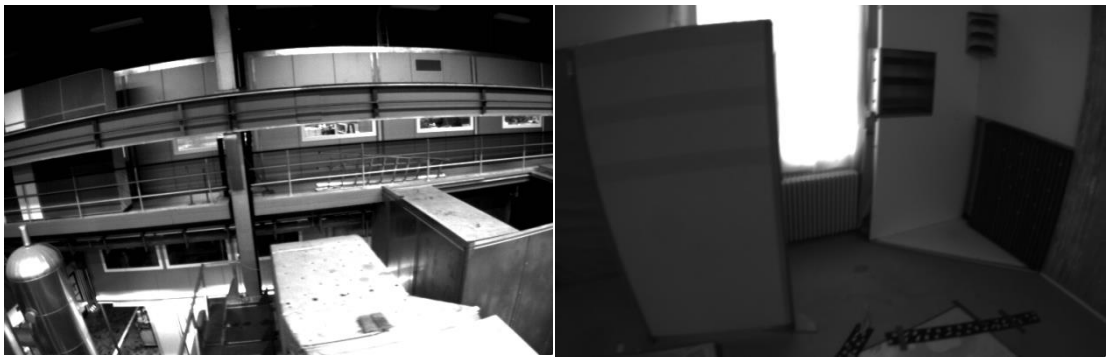


(a)easy



(b) medium

ISSN: 2664-9640



(c) difficult

Figure 2: Partial scene diagrams of different difficulties of the data set

4.2. Test results and evaluation

This paper selects 6 sets of data (MH_01, MH_03, MH_04, V1_01, V1_02, V1_03) for testing. The result shown in Figure 3 is the running effect of the algorithm in this paper on the MH_01 sequence. The top is the visual image of the trajectory of the algorithm in this paper. , The first view in the lower left corner is the visual screen of direct local mapping, the middle is the real tracking data but the screen, and the lower right is the visual image of global mapping by feature points. Fig. 4 is a 3D trajectory diagram after the running of Fig. 3 and its absolute error with the real trajectory.

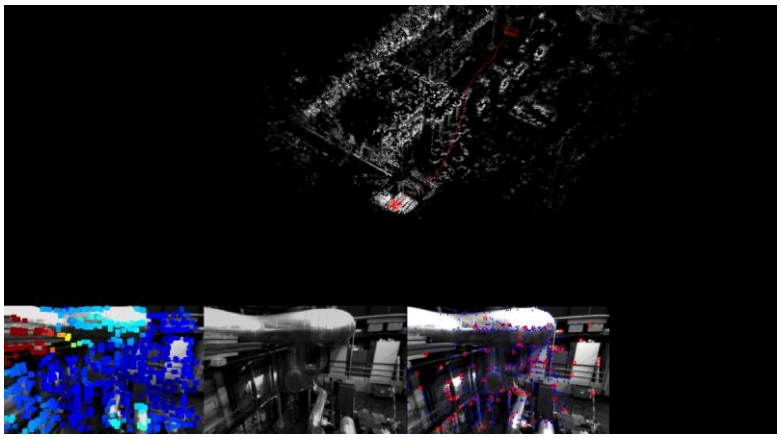


Figure 3: Schematic diagram of the operation effect of the algorithm in this paper

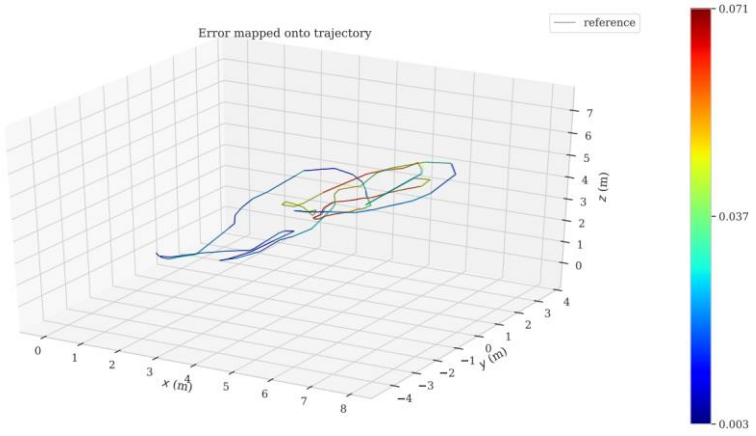
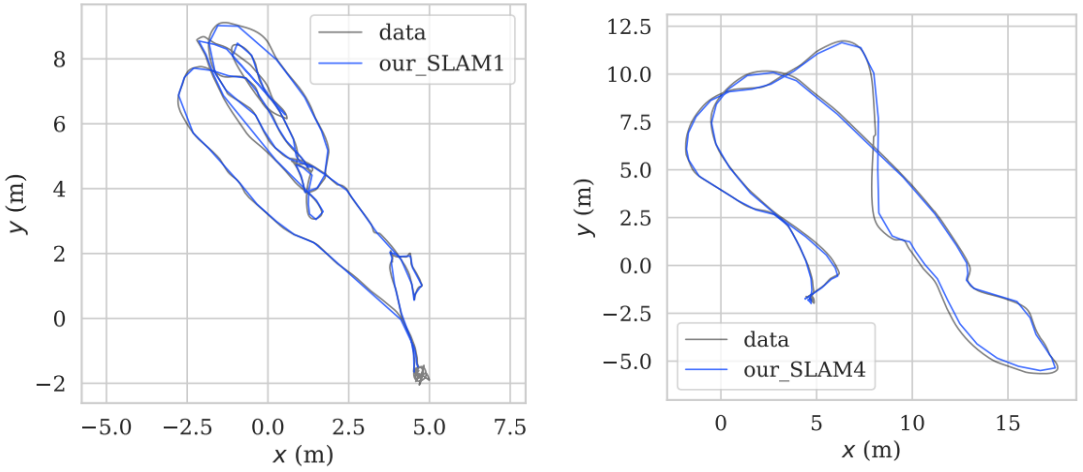


Figure 4: 3D trajectory diagram and absolute error of MH_01

Figure 5 shows the odometer experimental results of the algorithm on the MH_01 and MH_04 data sets. The gray part in the figure is the true value of the trajectory, and the blue line is the experimental trajectory result of this article. It can be seen that the algorithm in this paper performs well, and the actual trajectory basically coincides with the true value.

ISSN: 2664-9640



(a) MH_01 (b) MH_04

Figure 5: Odometer positioning effect on different difficulty data sets

In order to further verify the effect of the algorithm, this paper compares the posture/location accuracy of the proposed fusion algorithm based on direct method and feature point method, DSO based on direct method and ORB-SLAM based on feature point method. The algorithm was tested 5 times on the corresponding data set, and the average value of its root mean squared error (RMSE) was selected as the evaluation index. As shown in Table 1, the algorithm in this paper is better than the DSO algorithm in accuracy. Among them, the DSO algorithm basically failed 5 times on the V1_03 data set. The reason was that the camera moved too fast, which caused its pixel collection to fail; compared with the ORB-SLAM algorithm, it was tested on 6 data sets, 4 of which The accuracy of the test is better than ORB-SLAM.

Table 1: RMSE comparison of various algorithms

	本文算法	DSO	ORB_SLAM
MH_01_easy	0.038	0.053	0.045
MH_03_medium	0.039	0.156	0.037
MH_04_difficult	0.063	0.173	0.065
V1_01_easy	0.093	0.106	0.096
V1_02_medium	0.109	0.503	0.103
V1_03_difficult	0.431	×	0.452

5. In conclusion

This paper proposes a monocular vision algorithm based on the fusion of direct method and feature point method, which can effectively improve the positioning accuracy of monocular visual odometer. In this paper, the local map is constructed by the direct method, and the pose information of the marginal key frame is provided to optimize the feature point method, and the global map is constructed to achieve more accurate pose estimation. Compared with the traditional direct method, the algorithm in this paper has higher accuracy and stability. Because the feature point method consumes a certain amount of time to calculate the feature descriptor and matching, the algorithm in this paper is longer than the traditional direct method. We consider further optimizing the algorithm speed in this paper to reduce the running time.

ISSN: 2664-9640

Acknowledgements

Project support by Outstanding Young Science and Technology Talent Project of Sichuan Provincial Department of Science and Technology(Grant No.2020JDJQ0075)and Zigong Science and Technology Bureau Project(Grant No.2016DZ07) .

References

[1] Li Jianing. Research on Key Technologies of RGB-D Camera based Augmented Reality System[D]. Zhejiang University, 2017.(In Chinese)

[2] Wu Fan,Zong Yantao,Tang Xiaqing.Research status and prospects of visual SLAM[J].Application Research of Computers,2020,37(08):2248-2254.(In Chinese)

[3] Davison A J, Reid I D, Molton N D, et al. MonoSLAM: Real-time single camera SLAM[J]. IEEE transactions on pattern analysis and machine intelligence, 2007, 29(6): 1052-1067.

[4] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces[C]//Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on. IEEE, 2007: 225-234.

[5] Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: A versatile and accurate monocular SLAM system[J]. IEEE Transactions on Robotics, 2017, 31(5): 1147-1163.

[6] Zhang G L, Lin Z L, Yao E L, et al. Stereo visual odometry with multi-pose estimation constraints[J]. Control and Decision, 2018, 33(6): 1008-1016.(In Chinese)

[7] Richard A Newcombe,Steven J Lovegrove,Andrew J Davison.Dtam：Dense tracking and mapping in real-time[C]//Computer Vision(ICCV),2011 IEEE Internation Conference on.IEEE,2011:2320-2327.

[8] Engel J, Schöps T, Cremers D. LSD-SLAM: Large-scale direct monocular SLAM[C]//European conference on computer vision. Springer, Cham, 2014: 834-849.

[9] Engel J, Koltun V, Cremers D. Direct sparse odometry[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(3): 611-625.

[10] Forster C, Pizzoli M, Scaramuzza D. SVO: Fast semi-direct monocular visual odometry[C]//2014 IEEE international conference on robotics and automation (ICRA). IEEE, 2014: 15-22.

[11] J. Engel, V. Usenko, and D. Cremers, “A photometrically calibrated benchmark for monocular visual odometry,” arXiv:1607.02555, 2016.

[12] Gao X, Wang R, Demmel N, et al. LDSO: Direct sparse odometry with loop closure[C]//2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018: 2198-2204.