Research and Optimization of Real-time Simultaneous Localization and Mapping of Indoor Robot Based on Binocular Vision

Qiwei Zhang¹, Guanglu Zhou¹, Pengpeng Wang¹² and Fengguang Wu²
¹ School of Computer Science and Technology, Harbin Institute of Technology, 2 Wenhuaxi Road, 264209 Weihai, China
² Luoyang Electronic Equipment Test Center of China, 104 Tianjin Road, 471003 Luoyang, China

149858668@qq.com

Abstract. For the problem of inaccuracy and cumulative error of visual odometer, the research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision are studied. Based on ORB-SLAM2, key-frame map is created. First, the ORB feature is extracted from each frame of the input image and matched by fast approximation nearest neighbour (FLANN). Then, perform the preliminary pose estimation using EPNP, and optimize it with bundle adjustment and key-frame maps. When the tracking fails, apply key-frame maps and bag of words model to relocate. Finally, for the input binocular image, the SGBM is used to solve the parallax and then the depth, which will be converted to radar format data to create a map. In the research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision, propose a method of assisted positioning with key frame map, and a method of feature matching optimization and relocation, which combines various pose optimization to achieve the accuracy of the robot indoors positioning and map construction.

1. Introduction

With the deepening of intelligence, unmanned driving, virtual reality, augmented reality, and three-dimensional reconstruction have received extensive attention in both academic and general usage. However, the products in these areas are not yet mature. One of the key issues is the accuracy and efficiency of three-dimensional positioning. Also, the Simultaneous Localization and Mapping (SLAM) of visual robots is another core issue in three-dimensional positioning. The visual robot SLAM is divided into visual odometer and mapping. The development of current odometers mainly includes the method of the multi-sensor fusion, the method of deep learning, the method of optical flowing (OF) and the method of feature point. In the method of the multi-sensor fusion, the estimation the accuracy of the pose is directly related to the sensor used. In general, sensors with higher accuracy are also more expensive. Therefore, cost is the main reason that restricts the development of this method. The method of deep learning is a new method proposed in recent years. The main problem with this method is that in known environments, the estimation has high accuracy, while in unknown ones, the estimation accuracy of the pose is poor. The OF method, which is more severely affected by illumination, can only be used in specific environments. With the increase in computing speed and the
maturity of visual algorithms, the feature point method has become the most widely used method. This method refers to feature extraction of an image taken by a camera to estimate the pose of the camera. It has the advantages of rich information, low cost and good real-time performance. At present, the mainstream visual odometer system based on feature point method is also relatively mature. Representative methods such as SVO[1], PTAM[2], ORB-SLAM, LSD-SLAM[3,4], RTAB-MAP[5] have achieved good results in certain scenarios. The reconstruction mainly refers to three-dimensional reconstruction and two-dimensional reconstruction. Accurate maps ensure the accuracy of robot location navigation. This study is based on the ORB-SLAM2[6] framework and GMapping framework to create the key frame map and optimize the relocation method to solve the pose. Using EPnP[7] and BA to solve the robot pose, the precise positioning and map construction of the robot in the room is realized.

2. Basic knowledge

2.1. Binocular camera

A binocular camera is a device that interacts directly with the external environment and is a visually positioned eye. The physical model of the binocular camera is shown in Figure 1(a), and the geometric model is shown in Figure 1(b). The binocular camera is divided into a left eye camera and a right eye camera. In Figure 1(b), the optical centers of the left eye camera and the right eye camera are respectively recorded as $o_L$ and $u_R$, and are connected by a baseline $b$. The pixel points of the point $P$ in the left and right imaging planes are denoted as $P_L$ and $P_R$, respectively, $u_L$ and $u_R$ are the abscissas of the point $P$ on the left and right imaging planes. Because the $\Delta PP_LP_R$ is similar to the $\Delta PO_LO_R$, (2.1) is obtained. In (2.1), $d$ is recorded as the parallax of the spatial point in the left and right imaging planes. The depth of the spatial point can be obtained after the parallax is obtained.

$$z = \frac{fb}{d}, d = u_L - u_R$$

![Figure 1. Binocular vision camera model.](image)

2.2. Use the bag of word to match images

The bag of words, as dictionary data in relocation, generates a database of feature vectors for the images. First, collect get enough images, perform ORB[8] feature extraction on the images, and describe the feature with descriptors. Then, in [9], the dictionary is expressed by the K-tree in Figure 2. Finally, an unsupervised clustering algorithm is applied to form a word that represents a feature (or a collection of similar features)
3. Research and design of simultaneous localization and mapping of indoor robot based on binocular vision

The research and optimization of Simultaneous Localization and Mapping of indoor robot based on binocular vision is divided into three aspects: creating key frame map, visual odometer and mapping. First, save key-frames, landmarks, and related information acquired through the ORB-SLAM2 framework as key-frame maps. Secondly, load the key frame map in the visual odometer and extract the ORB feature for each frame of the input images and matched using fast approximate nearest neighbour (FLANN) [10]. Perform preliminary pose estimation using EPnP and optimize using bundle adjustment (BA) [11] and key-frame maps. When the tracking fails, an accurate estimation of the camera position is made using the key-frame map. Finally, a grid map is created based on the estimated camera pose and depth information of the image.

3.1. Create a key-frame map

Use a binocular camera to capture images from the environment, enter the obtained images into the ORB-SLAM2 frame to create a key-frame map and save it. When the first frame is input, the current frame is initialized. The camera coordinate system of the first frame is set as the origin coordinate of the world coordinate system. EPnP is applied to solve the pose of the camera for each frame of image input later. Determining key-frames and inserting them into the key-frame queue for optimization, finally, use bundle adjustment and closed loop detection to optimize all key-frames and save key-frame maps.

3.2. Computing visual odometer

Load the key-frame map and perform ORB feature extraction on the input image. Initial pose is estimated by EPnP and optimized with bundle adjustment. EPnP calculates the camera pose to represent the 3D coordinates of the world coordinate system as a weighted sum of a set of virtual control points. In general, the number of control points in the EPnP algorithm is four, and the four control points cannot be coplanar. Because the external parameters of the camera are unknown, the coordinates of the four control points in the camera reference coordinate system are unknown. The coordinates of the four control points in the camera reference coordinate system are solved, and the pose of the camera can be calculated. EPnP can be used to refer to the reference [7].

In the bundle adjustment optimization process, the coordinates of the spatial point are \( P_i = [X_i, Y_i, Z_i] \), and the coordinates of the pixel point are \( \eta_i = [x_i, y_i] \), and \( \xi \) indicates that the initial value is Lie algebra form of camera pose, which is solved by EPnP. \( s_i \) is the scaling factor. The relationship between pixel coordinates and spatial coordinates is shown in equation (3.1).The text should be set to single line spacing.
\[
\begin{bmatrix}
\varepsilon_i \\
\varepsilon_j \\
1
\end{bmatrix} = \text{K exp}(\xi^\wedge) \begin{bmatrix}
X_i \\
Y_i \\
Z_i \\
1
\end{bmatrix}
\]  

(3.1)

Write (3.1) as a matrix form as \( s_i^T \mu = \text{K exp}(\xi^\wedge)P_i \). To sum the errors, construct a least squares problem, and its reprojection error is shown in Figure 3. A preliminary pose estimation of P1 and P3 is performed on the matching points using EPnP.

**Figure 3.** Reprojection error diagram.

In the next frame image, the distance between P2 to P3 minimized by adjusting the camera pose to optimize the camera pose to optimize the camera pose shown in Equation (3.2).

\[
\xi^* = \arg \min_{\xi} \frac{1}{2} \sum_{i=1}^{n} \left\| u_i - \frac{1}{s_i} \text{K exp}(\xi^\wedge)P_i \right\|^2
\]  

(3.2)

Calculate the derivative of each error term with respect to the optimization variable, where \( e(x + \Delta x) \approx e(x) + J \Delta x \) is a Taylor expansion of the error term. \( e \) is the pixel coordinate error. \( x \) is the camera pose. \( J \) is a matrix of two rows and six columns. \( P \) is the coordinates of the space point in the camera coordinate system. The formula \( s_i^T \mu = KP \) can be obtained. For the \( \xi^\wedge \) left multiplication disturbance \( \partial \xi^\wedge \), calculate the derivative of the change amount of \( e \) with respect to the disturbance amount, and see equation (3.3) using law of the chain rule. (3.4) and (3.5) are the results of (3.3)

\[
\frac{\partial e}{\partial \xi^\wedge} = \lim_{\xi^\wedge \to 0} \frac{e(\delta \xi^\wedge \oplus \xi^\wedge) - e(\xi^\wedge)}{\partial \xi^\wedge} = \frac{\partial e}{\partial P} \frac{\partial P}{\partial \xi^\wedge}
\]  

(3.3)

\[
\frac{\partial e}{\partial P} = \begin{bmatrix}
\frac{\partial u}{\partial X} & \frac{\partial u}{\partial Y} & \frac{\partial u}{\partial Z} \\
\frac{\partial v}{\partial X} & \frac{\partial v}{\partial Y} & \frac{\partial v}{\partial Z}
\end{bmatrix} = \begin{bmatrix}
f_x & 0 & -f_x X/Z^2 \\
0 & f_y/Z & -f_y Y/Z^2
\end{bmatrix}
\]  

(3.4)

\[
\frac{\partial e}{\partial \xi^\wedge} = \begin{bmatrix}
\frac{f_x}{Z} & 0 & -f_x X/Z^2 & -f_y Y/Z^2 \\
0 & f_y/Z & f_x X/Z^2 & f_y Y/Z^2
\end{bmatrix}
\]  

(3.5)

This Jacobian matrix \( \frac{\partial e}{\partial \xi^\wedge} \) describes the first derivative of the reprojection error with respect to the Lie algebra of camera pose. In addition to optimizing the pose, the research and optimization of SLAM of indoor robots based on binocular vision also optimizes the spatial position of feature points. Therefore, it is necessary to calculate the derivative of \( e \) with respect to the spatial point \( P \). The calculation results are shown in formula (3.6) (3.7) (3.8).

\[
\frac{\partial e}{\partial \xi^\wedge} = \lim_{\xi^\wedge \to 0} \frac{e(\delta \xi^\wedge \oplus \xi^\wedge) - e(\xi^\wedge)}{\delta \xi^\wedge} = \frac{\partial e}{\partial P} \frac{\partial P}{\partial \xi^\wedge}
\]

\[
P = \text{exp}(\xi^\wedge)P = KP + t
\]

(3.6)

(3.7)
\[
\frac{\partial e}{\partial P} = \begin{bmatrix}
\frac{f_x}{Z} & 0 & \frac{-f_x X}{Z^2} \\
0 & \frac{f_y}{Z} & \frac{-f_y Y}{Z^2}
\end{bmatrix}
\]  

(3.8)

So far, a derivative matrix of pixel coordinate errors with respect to camera pose and feature point position is obtained. SLAM’s workers recognize the sparsity of the \( H \) matrix by the cumulative pixel coordinate error with respect to camera pose and feature point pose, which can be represented by graph optimization, which allows the use of this matrix in combination with Gauss-Newton method to real time optimize camera position and feature point positions are possible.

After obtaining the current frame pose by the Gauss-Newton method, the rotation matrix \( R \) and the translation vector \( t \) are obtained from the pose matrix. Use Rodriguez formulas (3.9) and (3.10). Since the camera rotates around the \( z \) axis, taking \( n = [0 \ 0 \ 1] \). At the same time, the rotation matrix is converted into Euler angles and the value of \( \theta \) is determined by combining the correspondence between \( \sin \theta \) and the rotation matrix \( R \). The \( \theta \) is summed to obtain the rotation angle \( \theta_{\text{sum}} \) of the current camera, and the calculation is given by Equation (3.11).

\[
R = \cos \theta I + (1 - \cos \theta)nn^T + \sin \theta n\wedge
\]

(3.9)

\[
\theta = \arccos \left( \frac{\text{tr}(R) - 1}{2} \right)
\]

(3.10)

\[
\theta_{\text{sum}} = \sum_{i=0}^{\infty} \theta
\]

(3.11)

When tracking the previous frame failure, use the created bag of word to determine the word bag vector of the current frame, and compare the similarity between the word bag vector of the current frame and the word bag vector of the key frame by using formula (3.12), and select similarity. The key frame, which is with the highest degree and is greater than the set threshold is used as the candidate key frame of the current frame. After determining the key frame of the current frame, feature extraction is performed on the current frame and the candidate key frame respectively, and the transformation matrix of the pose of the current frame relative to the pose of the key frame is calculated by matching the feature points, and the current frame posture is determined by transforming the matrix.

\[
s(a,b) = 1 - \frac{1}{w} \| a - b \|
\]

(3.12)

\( a \) is the word bag vector of the current frame, \( b \) is the word bag vector of the candidate key frame, and \( w \) is the proportion of each word in the word bag dictionary, and different weights are assigned according to the influence of different words on the result.

3.3. Create radar format data and build grid map
Converting deep data to radar format data is better compatible with current mainstream mapping frameworks and navigation frameworks. At the same time, since the general navigation positioning uses a two-dimensional grid map, the conversion of depth data is also an important part of the research and optimization of indoor robot SLAM based on binocular vision. According to different application scenarios, the binocular image is first converted into a depth map and projected onto a two-dimensional plane to generate a two-dimensional image, and then the two-dimensional image is converted into radar format data. Finally, radar format data is released to create grid maps and navigation against obstacles.

4. Research result
The research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision are divided into three aspects: creating key frame map, locating with key frame map and creating grid map.
4.1. Create a key-frame map

Create a key-frame map with the ORB-SLAM2 framework. Taking an indoor scene as the background, the binocular camera captures the image of the room, and the image is input into the ORB-SLAM2 frame to process and save the key frame map. Key-frame maps using key-frames, landmarks, and related information are basically consistent with indoor scenes. The results of creating a key-frame map are shown in Figure 4(b). A key-frame map is created. The blue box in Figure 4(a) is the extracted feature point.

![Figure 4. Create a key-frame map.](image)

4.2. Location use key-frame maps

When the current frame is tracked successfully, the pose of the current frame is obtained by the research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision; Figure 5(a) indicates that the feature points are correctly tracked, and the pose of the current frame is displayed in Figure 5(b). When a frame fails on frame tracking, the key frame map is used for relocation. Figure 5(c) indicates that the tracking failed, but after a short time, this success.

![Figure 5. Relocation when tracking fails.](image)

4.3. Create grid map

Firstly, the camera pose is obtained by using the method of solving the odometer in the research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision. Then, the binocular camera is used to solve the depth map and the 3D depth map is projected into the 2D plane to create the grid map. The final study results for the visual odometer are shown in Figure 6(a) visual odometer. The grid created map is shown in Figure 6(b).

![Figure 6. Creating a map.](image)
5. Conclusion
Research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision with an indoor scene as background, using a binocular camera sensor, a key-frame map is created based on the ORB-SLAM2 framework. In the visual odometer section, EPnP and BA are used to solve the camera pose. Use key-frame maps to assist in positioning at the same time. When the tracking fails, the real-time and accuracy of relocation is ensured by adding key frame maps and word bag dictionaries. Finally, the acquired binocular image is converted into a depth image and then converted into radar format data to construct a grid map. Research and optimization of real-time Simultaneous Localization and Mapping of indoor robot based on binocular vision integrates mainstream framework and stability algorithm of visual robot, In the computational visual odometer section, a key-frame map is used to assist in solving the odometer, optimize the relocation pose. It can effectively solve the problem of location and mapping of mobile robots.

References
[1] C. Forster, M. Pizzoli, and D. Scaramuzza, “Svo: Fast semi-direct monocular visual odometry,” in Robotics and Automation (ICRA), 2014 IEEE International Conference on (rs, ed.), pp. 15–22, IEEE, 2014.
[2] G. Klein and D. Murray, “Parallel tracking and mapping for small ar workspaces,” in Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on, pp. 225–234, IEEE, 2007.
[3] J. Engel, T. Schöps, and D. Cremers, “Lsd-slam: Large-scale direct monocular slam,” in Computer Vision–ECCV 2014, pp. 834–849, Springer, 2014.
[4] J. Engel, J. Sturm, and D. Cremers, “Semi-dense visual odometry for a monocular camera,” in Proceedings of the IEEE International Conference on Computer Vision, pp. 1449–1456, 2013..
[5] M. Labbé and F. Michaud, “Online global loop closure detection for large-scale multi-session graph-based slam,” in 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2661–2666, IEEE, 2014.
[6] Mur-Artal R, Tardos J D. ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras [J]. IEEE Transactions on Robotics, 2017:1-8.
[7] Lepetit V, Moreno-Noguer F, Fua P. EPnP: An AccurateO(n) Solution to the PnP Problem[J]. International Journal of Computer Vision, 2009, 81(2):155-166.
[8] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, “Orb: an efficient alternative to sift or surf,” in 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2564–2571, IEEE, 2011. Pham C C, Jeon J W. Domain Transformation-Based Efficient Cost Aggregation for Local Stereo Matching [J]. IEEE Transactions on Circuits & Systems for Video Technology, 2013, 23(7):1119-1130.
[9] D. Galvez-Lopez and J. D. Tardos, “Bags of binary words for fast place recognition in image sequences,” IEEE Transactions On Robotics, vol. 28, no. 5, pp. 1188–1197, 2012.
[10] M. Muja and D. G. Lowe, “Fast approximate nearest neighbors with automatic algorithm configuration,” in VISAPP (1), pp. 331–340, 2009.
[11] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, “Bundle adjustment: a modern synthesis,” in Vision algorithms: theory and practice, pp. 298–372, Springer, 2000.