Delayed Initialization of Monocular SLAM Based on Improved Least Squares

Huang Shuai[1, a], Fu Guangyuan[1], Wu Ming[1]

[1] Xi'an Research Inst. Of Hi-Tech, Hongqing Town, Xi'an, P. R.710025 China

[a]511508354@qq.com

Keywords: Depth estimation; monocular SLAM; least squares; triangulation; camera model.

Abstract: In order to solve the problem of lacking depth information of monocular SLAM, this paper proposes a method of delay depth initialization based on improved least squares. According to the principle of camera imaging, the robot performs two different observations on the same calibration point respectively, and the triangulation method is used to solve the initial depth information of the target pixel. Aiming at the problem of depth solution caused by the error of monocular wheeled robot, this paper improves the least squares method to estimate the optimal depth of target point, and the simulation experiment is carried out by using Matlab software. The experimental results show that the method proposed in this paper is simple and feasible for monocular SLAM. The method proposed in this paper is very accurate to initialize the depth information of monocular robot and the improved algorithm is superior to the least squares algorithm. Experiments show that the method proposed in this paper can solve the problem of lacking deepth information of monocular SLAM, and having the feasibility of applying to monocular SLAM.

1. Introduction

In recent years, with the development of robot-related industries, SLAM problems have been paid more and more attention[5,6,7]. In monocular SLAM problems, the depth estimation has always been a difficult problem and people's exploration of this problem has not been stopping[8,9,10]. The existing methods can be divided into three categories[11,12,13]: feature-based, gradient-based and optical-based. Among them, the feature-based approach [3,4] is extensively researched .The selected features can be points, lines, planes and so on. The gradient-based method [2], also called the direct method, estimates the depth and motion parameters according to the spatiotemporal differential of the gray scale image. This method is generally based on the brightness variation limit equation and the least squares as the estimator, so the method is sensitive to non-Gaussian residuals [2]. The optical flow based method [1] is sensitive to the optical flow calculation error because Optical flow is estimated to be a pathological problem, only the introduction of other restrictions (such as smooth) can be resolved. For the feature-based approach, it is now popular for the inverse depth method. In this paper, the feature-based method is used to estimate the depth of the target feature points by the triangulation method. The least squares method is improved according to the geometric meaning, and the feasibility of the method is verified by simulation experiment[4].

2. Camera model

The process by which a camera maps a point in a three-dimensional space to a two-dimensional imaging plane can be described by a geometric model. Using a simple model to explain the imaging process, as shown in Fig.1.

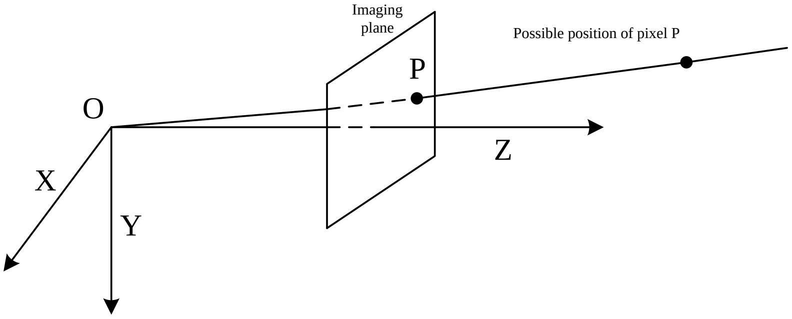

Fig. 1 Imaging model

It can be seen that the position of the pixel in the three-dimensional space can be constrained to a ray based on the coordinates of the camera and the pixel in the imaging plane. Theoretically, if two observations are made for the same target point, you can solve the three-dimensional coordinates of the target. This is similar to the binocular measurement model, as shown in Fig.2

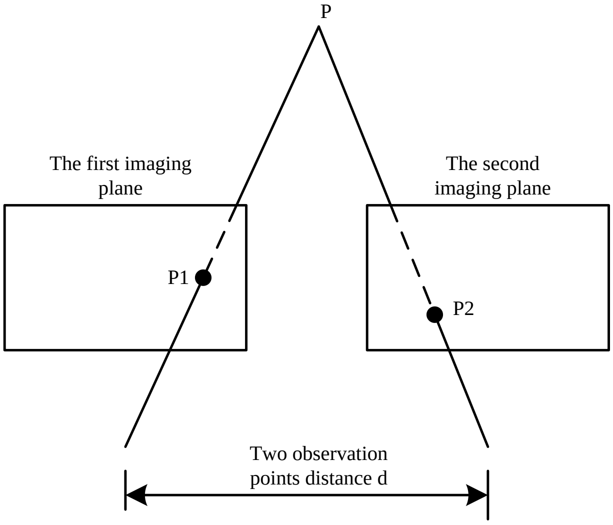

Fig.2 ideal two-time observation depth model

Further simplifying the model as shown in Fig.3

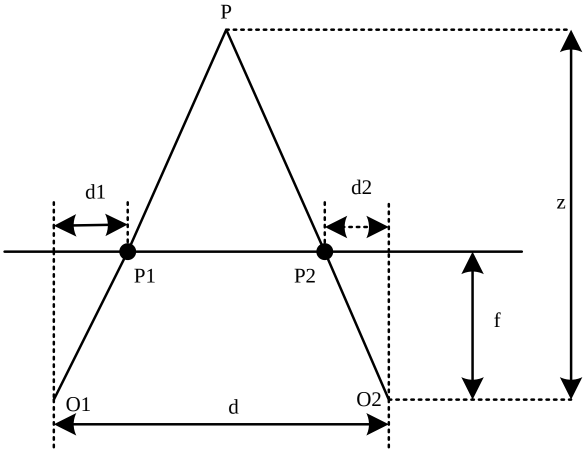

Fig.3 Geometric model

In Fig.3, f is the focal length of the camera, d is the distance between the two measurements, d_1 and d_2 represent the absolute values of the abscissa of the first and second target pixels, respectively, and z is the Z-axis direction of the three-dimensional target point Value. According to geometric knowledge

$$\frac{d}{d-d_2-d_1}=\frac{z}{z-f} \tag{1}$$

Thus

$$z=\frac{fd}{d_1+d_2} \tag{2}$$

In the actual situation, due to the presence of noise in the image, there are measurement errors between the two observation points, etc., which may result in the inability to solve the depth information of the target pixel.

The pixel coordinate system is usually defined as the origin O is located in the upper left corner of the image, the u axis is parallel to the x axis, and the v axis is parallel to the y axis. The difference between the pixel coordinate system and the imaging plane is a scaling and a translation of the origin. Assuming that the pixel coordinates are scaled α times on the u axis and β times on the v axis, At the same time, the origin shifted $[c_x, c_y]^T$. The relationship between the coordinates of $P^{'}$ and the pixel coordinates $[u, v]^T$ is

$$u=\alpha X^{'}+c_x \tag{3}$$

$$v=\beta Y^{'}+c_y \tag{4}$$

Merging αf into f_x and merging βf into f_y

$$u=\frac{f_x X}{Z}+c_x \tag{5}$$

$$v=\frac{f_y Y}{Z}+c_y \tag{6}$$

Among the units, the unit of f is meter, and the unit of α and β are pixel per meter, so the units of f_x and f_y are pixel. Putting the above formula into a matrix form as shown in equation (7)

$$Z\begin{bmatrix}u\\v\\1\end{bmatrix}=\begin{bmatrix}f_x & 0 & c_x\\0 & f_y & c_y\\0 & 0 & 1\end{bmatrix}\begin{bmatrix}X\\Y\\Z\end{bmatrix}\triangleq KP \tag{7}$$

In the formula, K is the internal parameter matrix of the camera. It is generally believed that the camera's internal reference is fixed after leaving the factory and will not change during use. And because the camera moves, so the camera coordinates of P should be its world coordinates (recorded as P_ω), according to the camera's current pose to the camera coordinate system results. The position of the camera is described by its rotation matrix R and the translation vector t:

$$ZP_{uv}=Z\begin{bmatrix}u\\v\\1\end{bmatrix}=K\left(RP_\omega+t\right)=KTP_\omega \tag{8}$$

Where R, t is called the camera's external parameters, and the internal parameters are different, the external participants will change with the camera movement, but also SLAM to be estimated goals, representing the robot's trajectory.

3. Triangulation

Triangulation is the distance between the target point by determining the angle of the same target point in two places. Triangulation, first proposed by Gaussian, applied to measurement. This paper mainly uses triangulation to estimate the depth information of the target pixel.

In two observations, The imaging points of point P are P_1 and P_2, respectively. Theoretically the line O_1P_1 and O_2P_2 will intersect at the real target point P, that is, the positions of P_1 and P_2

correspond to the positions in the three-dimensional scene. However, in the actual situation, due to image noise and distance error and other reasons, the two lines are often no intersection. This leads to the inability to find the analytical solution of the depth information. Therefore, this paper first uses the least squares method to find the optimal estimate of the depth information.

According to the definition of the polar geometry, assuming that x_1 and x_2 are the normalized coordinates of the two feature points, then they are satisfied:

$$s_1 x_1 = s_2 R x_2 + t \tag{9}$$

Through the introduction of the camera model, R, t is already known, what need solve is the depth of the two feature points s_1 and s_2. And the two depth information can be separated to seek. Such as s_2 , Multiply the two sides of the equation (9) by $x_2{}^T$ on the left side, there are:

$$s_1 x_1{}^T x_1 = s_2 x_1{}^T R x_2 + x_1{}^T t \tag{10}$$

The left side of the formula is zero, the right side can be seen as an equation of s_2, which can be obtained the value of s_2 directly. Solving s_1 is very easy after solving s_2. Due to noise and distance errors and other reasons, this paper seeks the depth of the improved least squares solution.

4. Simulation

In the global coordinates, the coordinates of the calibration target point are $X_m\left(x_m,\quad y_m,\quad z_m\right)$, the coordinates of the robot at the starting point are $X_c\left(x_c,\quad y_c,\quad z_c\right)$, the coordinates of the robot at the second point are $X_z\left(x_z,\quad y_z,\quad z_z\right)$, the first pixel coordinates are $X_1\left(x_1,\quad y_1,\quad z_1\right)$, the second pixel coordinates are $X_2\left(x_2,\quad y_2,\quad z_2\right)$ and the distance between the two recording points is d. Assuming that the coordinates of the target point to be solved are $\mathrm{X}\left(x,\quad y,\quad z\right)$:

$$\frac{x-x_1}{x_1-x_c}=\frac{y-y_1}{y_1-y_c}=\frac{z-z_1}{z_1-z_c}=k_1 \tag{11}$$

$$\frac{x-x_2}{x_2-x_z}=\frac{y-y_2}{y_2-y_z}=\frac{z-z_2}{z_2-z_z}=k_2 \tag{12}$$

Thus

$$\begin{cases} x-\left(x_1-x_c\right)k_1=x_1 \\ y-\left(y_1-y_c\right)k_1=y_1 \\ z-\left(z_1-z_c\right)k_1=z_1 \\ x-\left(x_2-x_z\right)k_2=x_2 \\ y-\left(y_2-y_z\right)k_2=y_2 \\ z-\left(z_2-z_z\right)k_2=z_2 \end{cases} \tag{13}$$

The coefficient matrix is

$$A=\begin{bmatrix} 1 & 0 & 0 & -\left(x_1-x_c\right) & 0 \\ 0 & 1 & 0 & -\left(y_1-y_c\right) & 0 \\ 0 & 0 & 1 & -\left(z_1-z_c\right) & 0 \\ 1 & 0 & 0 & 0 & -\left(x_2-x_z\right) \\ 0 & 1 & 0 & 0 & -\left(y_2-y_z\right) \\ 0 & 0 & 1 & 0 & -\left(z_2-z_z\right) \end{bmatrix} \tag{14}$$

$$B=\begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ x_2 \\ y_2 \\ z_2 \end{bmatrix} \tag{15}$$

Now, the solution of (13) can be rewritten as

$$\begin{bmatrix} x \\ y \\ z \\ k_2 \\ k_2 \end{bmatrix} = A^{+}B \tag{16}$$

However, since the least squares method does not take into account the geometric meaning represented by the established equations, the calculated accuracy of the coordinates is not high. In this paper, we consider the geometric meaning and improve the concept of least squares algorithm to solve the coordinates of three-dimensional space points, so as to achieve the purpose of seeking depth. It can be shown that the error in the geometric point of use of the midpoint of the out-of-plane straight line approximates the spatial point P is small (see [6] for the relevant proof).

Set foot down for:

$$m_1=[x_1,\quad y_1,\quad z_1]+t_1\bullet[(x_1-x_c),\ \ (y_1-y_c),\ \ (z_1-z_c)] \tag{17}$$

$$m_2=[x_2,\quad y_2,\quad z_2]+t_2\bullet[(x_2-x_z),\ \ (y_2-y_z),\ \ (z_2-z_z)] \tag{18}$$

And $\overrightarrow{m_1m_2}$ are vertical two straight lines $\overrightarrow{X_1X_c}$ and $\overrightarrow{X_2X_z}$:

$$\overrightarrow{m_1m_2}\bullet\overrightarrow{X_1X_c}=0 \tag{19}$$

$$\overrightarrow{m_1m_2}\bullet\overrightarrow{X_2X_z}=0 \tag{20}$$

From the equation (19) (20), the solution of the binary system can solve the value of t_1 and t_2 , so that the coordinates of the two ends of the common line segment can be obtained, and finally the midpoint coordinates can be obtained.

The simulation results are as follows:

Table 1 Comparison of experimental results

Corner number	Theoretical value(1)			Least squares Value(2)			Midpoint method value			Error(1)	Error(2)
1	2	2	0	1.9807	1.9343	-0.0432	1.9908	1.9643	-0.0329	0.08096	0.04941
2	3	2	0	3.0656	1.9449	-0.0258	3.0356	1.9849	-0.0158	0.08947	0.04177
3	5	2	0	4.9939	1.9595	-0.0124	4.9938	1.9897	-0.0127	0.04279	0.01749
4	6	2	0	6.0709	1.9555	-0.0074	6.0608	1.9857	-0.0079	0.08403	0.06296
5	8	2	0	7.9773	1.9658	0.0046	7.9972	1.9759	0.004	0.0413	0.02459
6	9	2	0	9.0564	1.9522	-0.0198	9.0363	1.9826	-0.0205	0.07654	0.04517
7	11	2	0	10.982	1.9699	0.0034	10.9819	1.9802	0.0026	0.03524	0.02695
8	12	2	0	12.0701	1.9502	-0.0279	12.0399	1.9806	-0.0188	0.0904	0.04819
9	14	2	0	14.0039	1.9655	-0.002	14.0038	1.9859	-0.0029	0.03478	0.01489
10	2	3	0	1.982	3.01	-0.0345	1.982	3.01	-0.0145	0.04018	0.02518
11	3	3	0	3.0699	3.0213	-0.0162	3.0399	3.0113	-0.0161	0.07485	0.04448
12	5	3	0	4.9891	3.0312	-0.0001	4.9891	3.0111	0	0.03305	0.01556
13	6	3	0	6.0644	3.0291	-0.003	6.0244	3.0192	-0.0032	0.07073	0.03121
14	8	3	0	7.961	3.0466	0.0114	7.961	3.0166	0.0114	0.06183	0.04389
15	9	3	0	9.0457	3.0272	-0.0115	9.0456	3.0173	-0.0116	0.05441	0.05013
16	11	3	0	10.9635	3.058	0.0139	10.9635	3.0181	0.0139	0.06992	0.04305
17	12	3	0	12.0569	3.0254	-0.0314	12.0269	3.0155	-0.0116	0.06978	0.03314
18	14	3	0	13.9767	3.0693	0.0049	13.9767	3.0294	0.0048	0.07328	0.03782
19	2	5	0	1.9653	4.9219	-0.0385	1.9854	4.9618	-0.0179	0.09373	0.04464
20	3	5	0	3.0642	4.939	-0.0203	3.0243	4.969	-0.01	0.09086	0.04064
21	5	5	0	4.9865	4.9431	-0.0017	4.9866	4.9431	-0.0015	0.0585	0.05848
22	6	5	0	6.0632	4.9392	0.0055	6.0232	4.9692	0.0055	0.08787	0.03895
23	8	5	0	7.9599	4.9577	0.0132	7.9899	4.9578	0.0129	0.05976	0.04527
24	9	5	0	9.0325	4.9399	-0.0159	9.0324	4.9702	-0.0164	0.07015	0.04698
25	11	5	0	10.9601	4.9621	0.0061	10.9601	4.9623	0.0057	0.05537	0.05519
26	12	5	0	12.0547	4.9384	-0.0295	12.0247	4.9687	-0.0301	0.0875	0.04996
27	14	5	0	13.9893	4.9606	0.0045	13.9893	4.961	0.0037	0.04107	0.04061
28	2	6	0	1.9701	6.0097	-0.0439	1.9702	6.0096	-0.0436	0.05399	0.05368
29	3	6	0	3.0665	6.0193	-0.0155	3.0665	6.0193	-0.0154	0.07096	0.07094
30	5	6	0	4.9721	6.0261	-0.0037	4.9722	6.0259	-0.0033	0.03838	0.03814

5. Conclusions

It can be seen from the experimental results that the error of the depth value is smaller by replacing the traditional least squares solution with the midpoint of the outright straight line. This method provides another feasible method for single-head SLAM depth estimation, which has certain theoretical and engineering significance.

References

[1] Gao Xiang, Zhang Tao, Visual SLAM 14, Electronic Industry Press, March 2017, Beijing.

[2] Luo Shimin,Li Maoxi, binocular vision measurement of three-dimensional coordinates of the method of research, computer engineering and design, in October 2006, Nanchang.

[3] Ma Feng, Li Qiongyan, Zhao Yadong, the three-dimensional point reconstruction based on the midpoint algorithm of the cross-line vertical line segment, modern manufacturing engineering, July 2009, Beijing

[4] Christian Forster, Zichao Zhang, Michael Gassner, Manuel Werlberger, and Davide Scaramuzza，SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems, IEEE TRANSACTIONS ON ROBOTICS, APRIL 2017.

[6] SUN Jun. Intelligent Integrated Performance Test Instrument for Motor Vehicle [J]. Mechanical and Electrical Engineering, 1997,14 (1): 21.

[7] Niclas Zeller, Franz Quint, Uwe Stilla,Depth estimation and camera calibration of a focused plenoptic camera for visual odometry, ISPRS Journal of Photogrammetry and Remote Sensing,2016.

[8] Sotirios Diamantas, Stefanos Astaras, Aristodemos Pnevmatikakis, Depth Estimation in Still Images and Videos Using a Motionless Monocular Camera, IEEE Instrumentation and Measurement Society,2016.

[9] George Bardas,Stefanos Astaras,Sotirios Diamantas,Aristodemos Pnevmatikakis,3D Tracking and Classification System Using a Monocular Camera, October 2016.

[10] Jishnu Keshavan, HectorEscobar-Alvarez, J.SeanHumbert,An adaptive observer framework for accurate feature depth estimation using an uncalibrated monocular camera, Control Engineering Practice,2016

[11] Zhong Zhiguang, Yi Jianqiang, Zhao Dongbin, a depth and motion estimation method based on point pair, robot, Beijing, 2005.03.

[12] TagawaN, Inagaki A,Minagawa A. Parametric estimation of optical flow from two perspective views[J]. IEICE Transactions on Information and Systems, 2001, E84 -D(4): 485 -494.

[13] Park SK, Kweon IS. Robustand directestimation of 3-Dmotion and scene depth from stereo image sequences[J]. Pattern Recognition,2001, 34(9): 1713 -1728.