ISSN: 2414-1895
DOI: 10.6919/ICJE.202009_6(9).0047

Real-time Pose Calculation of Monocular Vision Cooperative Target Based on Optical Flow Method and Ray Triangulation Method

Xiaowei Feng[1], Kaixun Zhang[1, a]

[1]School of Shanghai Maritime University, Shanghai 200000, China.

[a]zhangkaixun0514@163.com

Abstract

Aiming at the problem of long time, large matching error and many registration models for cooperative object based on orb feature matching, a global cooperative object pose calculation method based on sparse optical flow method and ray triangulation method is proposed. Based on the sparse optical flow method, the two-dimensional feature points are tracked, the points with bad matching are eliminated, and the pose calculation is realized by pnp algorithm. If the matching points are less, the feature points in the current frame are extracted by ray triangulation method and the world coordinates are estimated to improve the robustness of the algorithm. When the camera parallax changes too much, in order to prevent the accumulation of new feature point error, the feature matching is carried out in the original registration model, and then the current pose is calculated. Experimental results show that the speed of pose calculation is 6 times higher than that of orb feature matching algorithm, the error of feature matching is reduced by 19%, and the registration of model is reduced by more than 50%. And the algorithm has good applicability to illumination change, angle change and target surface similarity.

Keywords

Sparse optical flow method; Ray triangulation method; Feature matching; Cooperative object pose calculation.

1. Introduction

The cooperative object pose real-time computing is a widely studied problem in the field of computer vision and robot. It is widely used in robot navigation, robot arm picking up target, spacecraft space rendezvous and docking [1]. The cooperative object pose calculation can be divided into binocular visual pose calculation and monocular visual pose calculation according to the number of cameras. Binocular camera can directly obtain the feature point depth information, the position and pose calculation speed is fast, the accuracy is high, but it has the defects of small measurement range, stereo matching difficulty and so on, the application scene is limited. The monocular camera has simple structure and little restriction on the use environment, which is concerned by many researchers [2]. The cooperative target is known for its three-dimensional information, and the model can be registered in advance in each perspective of the target [3]. The model information can contain feature point descriptors, feature point 3d coordinates, and feature point 2d coordinates in the registration perspective imaging plane. cooperative target pose calculation can be divided into three steps: current frame feature point descriptor extraction, feature point matching with each perspective model, pnp algorithm to calculate cooperative target pose. Feature point extraction has orb, sift, surf, etc. Orb feature extraction combines fast corner extraction with brief descriptors, which ensures the real-time

ISSN: 2414-1895 DOI: 10.6919/ICJE.202009_6(9).0047

performance of feature extraction, but the accuracy is poor. Based on the orb feature matching, it takes a lot of time to match the current frame with the registration model, which can affect the real-time and accuracy of pose calculation [4]. In order to achieve global pose calculation and improve the accuracy of pose calculation, multiple views registration model is often needed, and a large number of registration models further increase the matching time and storage space consumption. In order to further improve the real-time, accuracy and the number of models can still achieve the cooperative goal of global pose calculation. A feature matching optimization algorithm based on sparse optical flow is proposed to replace traditional orb feature matching, to reduce error matching and improve matching speed.

2. Basic Algorithm

2.1 Model registration

The actual target size and camera parameters are known and the target is photographed from any angle of view. In the program interaction, the photos are marked, and the pixels of the known 3D coordinates on the target are clicked in turn to realize the initial matching between the 3D points and the 2D pixels, and the triangular mesh is generated according to the vertex combination to fit the surface of the object. In the case of a cube, the three-dimensional coordinates of eight vertexes are known, and 12 triangles of eight coordinates are known. Model registration process flow chart and registration effect as follows .

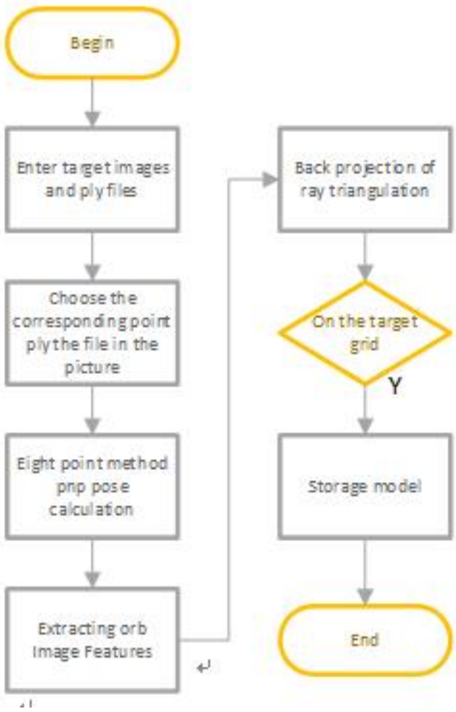

Fig.1 Registration process flow

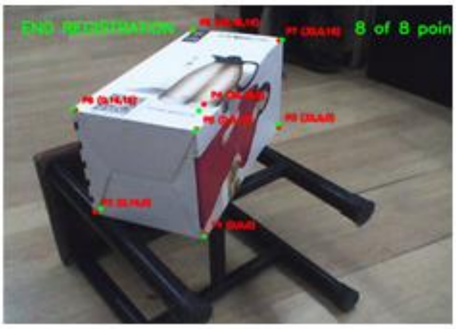

Fig.2 End registration

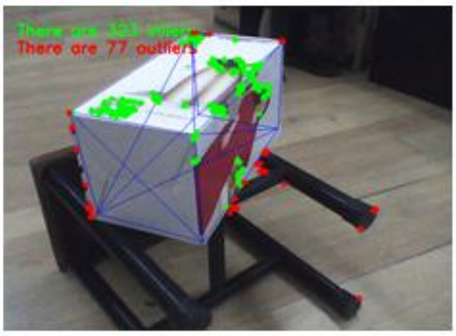

Fig.3 Registration model

2.2 ORB algorithm.

ORB algorithm combines the fast feature points with the brief descriptor, which is less affected by the image noise and runs fast, and is widely used in the scene with high real-time requirements. fast corner detection determines whether each pixel is the corner point to be detected by calculating the difference between the gray value of each pixel and a certain range of pixels around it. brief a pair of pixel points are randomly selected from the defined neighborhood around the key points to compare their brightness, the descriptor corresponding bit allocation 1 or 0 indicates the comparison result, and the descriptor of the key point can be obtained by repeated extraction 256 times. moreover, orb

ISSN: 2414-1895 DOI: 10.6919/ICJE.202009_6(9).0047

establish a two-dimensional coordinate system with the line of the key points and the centre of point-taking region as the x axis in the calculation of descriptors to solve the problem of rotational invariance that brief descriptors do not possess. In OpenCV, the algorithm uses image pyramids to improve the scale consistency problem.

2.3 LK Optical Flow Method

LK optical flow method minimizes the photometric error by some corner points between two images, calculates the moving direction of the corner points, and then tracks these corner points [5]. The method is based on three assumptions:

The image constraint equation can be obtained under the assumption of constant brightness:

$$I(x,y,t) = I(x+\delta x, y+\delta y, t+\delta t) \tag{1}$$

If the time continuous or the motion between images is small, then the image constraint equation can be expanded by Taylor because the moving is small enough to ignore the second and higher order terms.

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t \tag{2}$$

Combining (1) with (2), and t derivation $u = \frac{\delta x}{\delta t}, v = \frac{\delta y}{\delta t}$ Represents the velocity of light flow in the horizontal and vertical directions.

$$I_x u + I_y v = -I_t \tag{3}$$

spatial consistency, the adjacent points have similar motion, and the adjacent points in the original image are also adjacent in the image to be matched. Nine equations can be obtained by bringing 9 pixels into the (3) formula. The summary is as follows:

$$\begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ . & \\ . & \\ I_{x9} & I_{y9} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} -I_{t1} \\ -I_{t2} \\ . \\ . \\ -I_{t9} \end{bmatrix} \tag{4}$$

Nine equations solve two unknowns, which is a super definite problem, which is solved by the least square method to predict the pixel position in the image to be matched.

2.4 Ray triangulation

The algorithm is used to determine whether a ray in a three-dimensional space can intersect a triangular plane in space, and if it can intersect, the three-dimensional coordinates of the intersection point can be obtained[6] .The ray $l(m)$ is seen as composition of origin O and direction vector N.

$$l(m) = O + mN \tag{5}$$

The triangle slice consists of three vertexes, V_0, V_1, V_2. A point in a triangle can be composed of three vertexes linearly. Include $a \geq 0, b \geq 0, a+b \leq 1$.

ISSN: 2414-1895
DOI: 10.6919/ICJE.202009_6(9).0047

$$P(a,b)=(1-a-b)V_0+aV_1+bV_2 \tag{6}$$

If the ray intersects the triangle face .

$$O+mN=(1-a-b)V_0+aV_1+bV_2 \tag{7}$$

Can be converted to matrix form as follows:

$$\begin{bmatrix}-N, V_1-V_0, V_2-V_0\end{bmatrix}\begin{bmatrix}m\\a\\b\end{bmatrix}=O-V_0 \tag{8}$$

If the a,b is satisfied in the solution, that is, there is an intersection point between the ray and the triangle, the three-dimensional coordinates of the intersection point can be obtained by the solution of (6). Compared with the traditional algorithm, this algorithm does not need to solve the plane equation of the triangle slice. The calculation is simple, the storage space is small, and the program is more efficient.

3. Our algorithm flow

The sparse optical flow method is used to predict the moving direction of the feature points, and the pixel coordinates of the feature points in the next frame image are obtained. Without calculating the feature point descriptor, it also saves the time consumed by the feature point matching with the model. After the two-dimensional coordinates in the feature point image are obtained in the new image frame, the camera pose can be solved pnp algorithm with the corresponding world coordinates in the model. With the increase of the number of frames, the optical flow method continuously eliminates the points with poor tracking effect. In order to ensure the correct pose, the ray triangulation method is used to increase the number of feature points under the current estimated pose. In order to ensure the accuracy of global pose calculation and reduce the accumulation of optical flow tracking errors, when the camera perspective changes too much, the appropriate feature points are re-matched in the registration model for tracking.

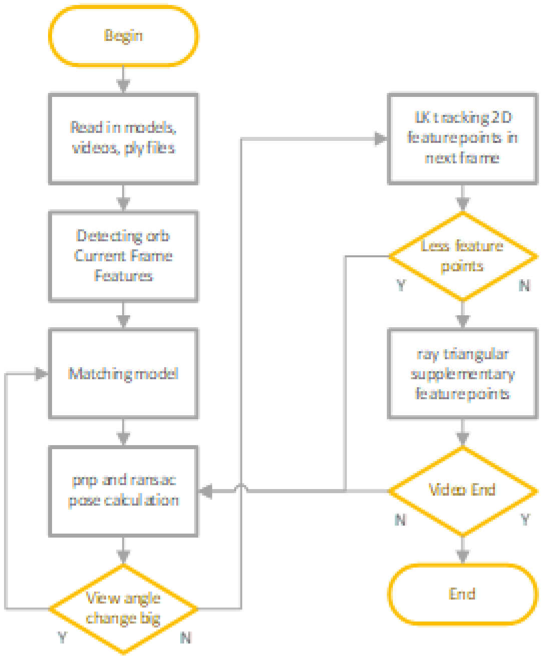

Fig.4 Algorithm flow

ISSN: 2414-1895 DOI: 10.6919/ICJE.202009_6(9).0047

This algorithm can self-adaption increase the number of feature points in the pose calculation, and ensure the continuous stability of the pose calculation results. The registration quality of the model is low, and four objects can be created evenly around the object. If the memory space allows, the number of registration model can improve the stability of the pose calculation.

4. Experimental result

The system is ubuntu14.04, processor intel i5-8300@2.30GHz, running memory 3GB, video pixels is 640×480, a total of 2850 frames.

Table 1 Two algorithm comparing

	Feature matching time	Average frame rate	Number of model matches	Error matching ratio	Number of models
Optical flow and ray triangulation	0.03136	12.053	17	0.458%	4
orb feature matching	0.40410	2.191	291	19.357%	10

The experimental data are shown in Table 1. The feature matching time is the time spent in the program to predict the feature points in the current frame pixel coordinates. The average frame rate is divided by the total number of frames divided by the total time spent by the program. During the pose calculation, the feature points decrease with the change of camera angle of view, and find the most matching model with the current angle of view in the registered model, the less the number of re-matching, indicating that the program depends on the model less. When the feature points are matched, the results are ransac filtered while pnp the algorithm, and the ratio of the number of external points to the total number of matching points is taken as the proportion of false matching. The experimental data show that the algorithm takes less time, has high precision of feature matching, has few target models and is not dependent on the model.

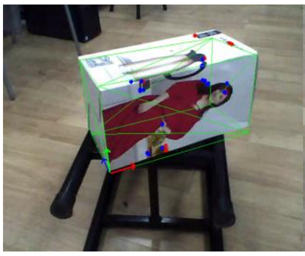
Fig.5 LK-1


Fig.6 LK-2


Fig.7 LK-3


Fig.8 LK-4


Fig.9 ORB-1

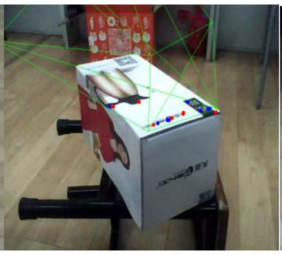
Fig.10 ORB-2


Fig.11 ORB-3

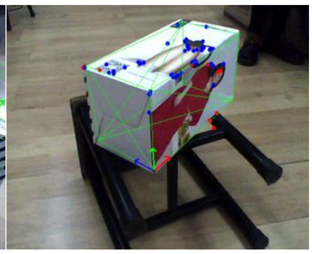
Fig.12 ORB-4

In the process of running the program, the object surface mesh is projected to the image by using the calculated pose, like Fig.5-Fig.12, and the accuracy of the pose calculation can be obtained by

ISSN: 2414-1895 DOI: 10.6919/ICJE.202009_6(9).0047

comparing the coincidence degree between the mesh and the actual object. It can be seen from the graph that the algorithm has higher accuracy of pose calculation, stable operation of all global perspectives, adaptive change of the number of feature points, and enhanced robustness of the program.

5. Summary

Cooperative target pose calculation is a hot spot in the field of machine vision. In this paper, optical flow method is used to track the target surface feature points. Compared with the traditional orb matching takes less time. when the number of feature points is small, the adaptive supplementary feature points based on ray triangulation method are tracked. The program runs more robust and has low dependence on the model. In order to realize the accurate calculation of global pose, the camera parallax judgment is introduced and the registration model is re-matched. The experimental results show that the algorithm is fast, the accuracy of optical flow tracking matching is high, the number of models is small, and the dependence degree is low.

References

[1] Junjie Ma, Daqing Huang, Menhao Qiu, Yongfu Gong. Relative pose estimation of UAV based on cooperative target identification[J]. Electronic Design Engineering, 2020,28(10):p.1-6.

[2] Ning Xu. Real-time visual positioning of monocular camera [D]. Master's degree, Shanghai jiao tong university, China 2008.p.1-87.

[3] Yabing Wang, Xiaowei Feng, Jianmei Xiao, Xihuai Wang. Real-time Target Distance Measurement Based on Monocular Vision [J]. Industrial Control Computer, 2018, 31(01): p.110-112.

[4] Feihong Xu, Lin Sun, Peng Qi, Yuqi Wang, Hao Hu. Summary of Sitting Position Recognition Based on orb Algorithm [J]. Wireless Internet Technology, 2019,16(06): p.115-117. .

[5] Fast stereo visual odometry based on LK optical flow and ORB-SLAM2.Tang, Chuanye (Automotive Engineering Research Institute, Jiangsu University, Zhenjiang; 212013, China); Zhao, Xinwen; Chen, Jianfeng; Chen, Long; Zhou, Yazhou Source: Multimedia Systems, 2020 Article in Press

[6] Tomas Möller & Ben Trumbore (1997) Fast, Minimum Storage Ray-Triangle Intersection, Journal of Graphics Tools, 2:1, p.21-28