Available online www.jsaer.com

Journal of Scientific and Engineering Research, 2020, 7(7):8-15

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Design of Automated Guided Vehicle (AGV) Visual Odometry Based on Oriented Features from Accelerated Segment Test and Rotated Binary Robust Independent Elementary Features (ORB) Algorithm

Hui Li*, Ya-qian Zhou, Jiang-hui Gao, Yi Wang, Xiao-qing Sang

School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo, Shandong, China
*Corresponding author: lihui619517439@163.com

Abstract The design of visual AGV odometry based on ORB algorithm is proposed by Astra camera. The depth information of image frame is obtained by using depth camera and visual odometer. For the extraction of ORB feature points, image frames are matched according to Hamming distance, false matches are removed by progressive sampling consensus (PROSAC) algorithm, and the position and pose of the camera are processed iteratively by singular value decomposition (SVD) method in iterative closest point (ICP) algorithm to the motion of visual AGV. In this paper, we use the linear motion of the experimental car to test the effectiveness of the algorithm. The experimental results show that the error range is maintained at 0.783% in the linear motion test. Therefore, the visual odometer designed can solve the problem of self-localization and estimation of visual AGV in the running process.

Keywords AGV, Visual odometry, Feature matching, Homogenization, Location

1. Introduction

The visual odometry [1] can deduce and estimate a series of high frequency image frames by using cameras, to obtain the relative motion of the robot. Meanwhile, with the appearance of red green blue depth map (RGB-D) camera, the visual odometer method based on RGB-D has been widely used in recent years [2-6]. At present, the design of visual odometry based on RGB-D camera algorithm is mainly divided into two categories: sparse feature point method and dense method. Based on the sparse feature point method, the key frame is obtained by using the feature point method such as speeded up robust features [7], scale-invariant feature transform [8] and ORB[9]. The motion of the camera is estimated according to pixel information based on dense method. The advantages of ORB feature point visual odometer are that it requires less hardware, less computation, and more real-time performance.

Domestic and foreign scholars have done a lot of research on visual odometer based on sparse feature point method. Henry, et al. [10] used scale-invariant feature transform algorithm to detect and extract features, the author also used the random sampling consistency PROSAC algorithm to eliminate wrong matches. Although this method improves the accuracy, it sacrifices the computation and has high requirements for the configuration of upper computer. Therefore, this algorithm is ideal for short distance application. Some scholars applied the RGB-D camera to the visual odometry system, which makes it possible to obtain the scale by using the monocular odometry. They used the method of random sample consensus to purify the matched feature point pairs. Although this method can reduce the accumulated error of odometer, it does not consider the problem of high image feature point density. Endres, et al. [11] used RGB-D SLAM method, the camera still exist error transformation matrix. Therefore, there are still some improvements in RGB-D SALM front-end.

The above research methods have the problems of large computation, poor real-time performance and low efficiency. In order to solve the above problems, this paper proposes a visual odometer method based on ORB feature point method, which can well solve the problem of camera pose estimation based on ORB feature points.

2. Design of visual mileage calculation methods

Visual odometry is mainly to solve the problem of positioning estimation of visual AGV in operation. First, the RGB image and depth image of the outside world are obtained through RGB-D camera. Then, the ORB feature points of the current frame image are extracted and matched with the feature points of the current key frame. Then, ICP algorithm iterative estimation is carried out based on the matched 3D points to obtain real-time bit and pose information of visual AGV. The visual odometer structure is shown in Figure 1.

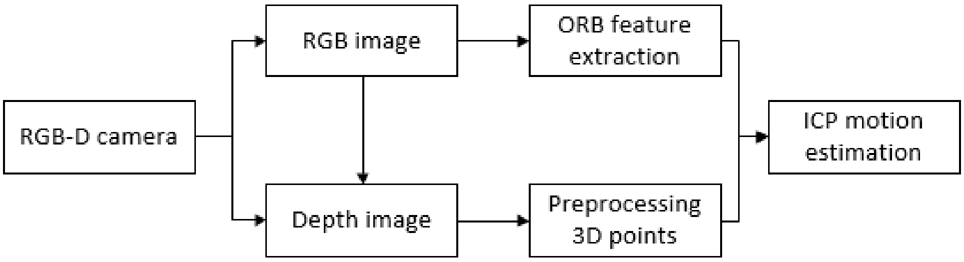

Figure 1: Visual odometer framework

3. ORB feature extraction and matching

ORB algorithm is divided into two parts: feature point extraction and feature point description. Feature extraction is developed by features from accelerated segment Test (FAST) algorithm, feature point description is improved according to the binary robust independent elementary features (BRIEF) description algorithm. First of all, from the need to extract the characteristics of the image to select a pixel P, in order to determine whether there is a feature point, we choose the area brightness value is set to the I_p, and set up an initial threshold T, then we make circle with radius r pixels, the center of the circle is this pixel point. So there must be m pixels on the boundary of this discrete circle, if the size of n discrete circle of pixel, there are n consecutive pixels. In addition, the continuous pixel values are bigger than I_p+T or are smaller than I_p-T, so it can be defined as an angle. At the same time, in order to calculate the main direction of feature points, in terms of rotation, the main direction of the feature point is calculated using the extracted FAST feature points in ORB feature extraction, so the descriptor adds rotation invariance. The specific steps are as follows:

In an image, the moment of the image block is used to calculate the centroid of the feature point with radius r. The moment is defined as:

$$m_{pq} = \sum x^p y^q I(x, y),\ p, q = \{0,1\} \tag{1}$$

The relationship between the center of mass and moment of the image block is:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \tag{2}$$

By connecting the center of mass C with the center of mass O of the image block, its direction vector $\overline{OC}$ can be obtained. The direction of the feature point is:

$$\theta = \arctan\left(\frac{m_{01}}{m_{10}} \right) \tag{3}$$

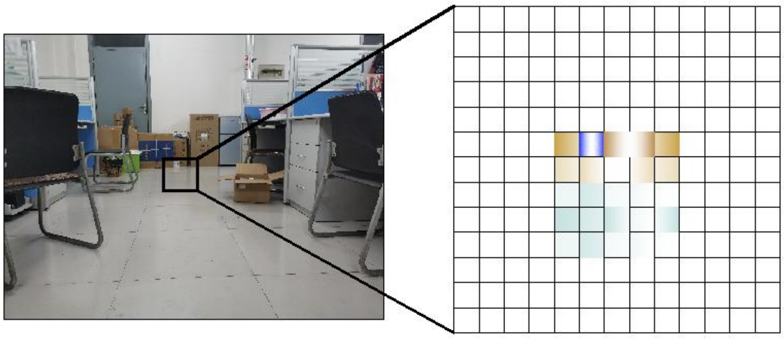

Figure 2: FAST feature points

After the key points of Orient FAST were extracted, feature matching was carried out using BRIEF algorithm. BRIEF is a binary descriptor, which adopts the method of randomly selecting points. Its description vector is usually composed of many zeros and ones, which has the characteristic of FAST matching speed. In the ORB algorithm, the random points we selected are generally conform to the Gaussian distribution and have length *n*. In this paper, the Gaussian kernel is used in a region of reasonable size. In order to avoid the influence of rotation on the feature matching of BRIEF, it is necessary to make BRIEF have anti-rotation characteristics. Therefore, the direction to start searching for key points of FAST is recorded as the direction of BRIEF. The specific steps are as follows:

Defining any n binary set as a 2n matrix:

$$S = \begin{pmatrix} x_1, x_2 \dots x_n \\ y_1, y_2 \dots y_n \end{pmatrix} \tag{4}$$

Taking points to compute the BRIEF descriptor from the coordinates after the calculation:

$$g_n(p,\theta) = f_n(p) \mid (x_i, y_i) \in S_\theta \tag{5}$$

Where, $S_\theta = R_\theta S$, $R_\theta = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$.

We can judge whether the ORB algorithm matching complete by calculating hamming distance of BRIEF descriptor. In addition, fast library for approximate nearest neighbors (FLANN) algorithm was used to obtain the nearest neighbor point and the second neighbor point. The hamming distance between the feature point and the nearest neighbor point was set as *d*, and the hamming distance between the secondary neighbor point and the feature point was set as L. When the distance ratio $\gamma = d/L$ is less than the given threshold, the match is successful; otherwise the match failing, the smaller the γ, the higher the match [12].

4. Remove the wrong match

Random Sample Consensus (RANSAC) algorithm needs a best homography matrix H, the size of the matrix is 3×3, PROSAC algorithm usually selects high quality points for priority sampling, sorts them according to their matching degree with the model, and computes the corresponding homologous matrix. Generally speaking, the motion attitude model is estimated by matching high quality priority sampling, which can greatly reduce the number of iterations and improve efficiency. By finding the optimal parameter matrix, RANSAC algorithm increases the data points satisfying the matrix by finding the optimal parameter matrix. In order to facilitate calculation and analysis, it is usually necessary to normalize the matrix. Since there are eight unknown parameters in the homography matrix, eight linear equations need to be solved. Two equations can have a set of corresponding points, so at least four sets of matching point pairs are included.

$$s\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{6}$$

Where, *s* is the scale parameter of the target image, *(x,y)* represents the corner position of the selected image, *(x',y')* is the corner position of the scene image.
PROSAC algorithm solves the Euclidean distance ratio β by calculating the minimum Euclidean distance d_{min} of the feature points of the target image. Then calculate quality factor to measure the effect of matching and quality, and according to the descending order of matching effect must be, four sets of points with high matching quality are calculated first, then the homography matrix H is calculated and compared with the projection point. Two points are used to judge whether it is an inside point or an outside point. If the number of inside points is greater than the set threshold, the corresponding inside points and their related model parameters will be output, and the iteration will continue on the contrary.

$$\beta = \frac{d_{\min}}{d_2} \tag{7}$$

Where β is the ratio of Euclidean distance, d_{min} is the minimum Euclidean distance and d_2 is the secondary minimum Euclidean distance.
Generally speaking, PROSAC algorithm and RACSAC algorithm usually input some observation data with large noise or invalid points, as well as parametric models and some trusted parameters for interpreting observations and analyzing data. PROSAC algorithm can quickly remove the mismatches by matching, sorting, calculation and iteration.

5. Camera pose estimation

The design of visual odometer based on RGB-D camera is to consider the problem of 3D-3D pose estimation. Therefore, we assume that there is a set of 3D points that have been matched:

$$P = \{p_1, \dots p_n\} \ , \ P^{'} = \{p_1^{'}, \dots p_n^{'}\} \tag{8}$$

After Euclidean transformation:

$$\forall i, \ p_i = Rp_i^{'} + t \tag{9}$$

ICP has been solved in two ways: singular value decomposition (SVD) method using linear algebra and nonlinear optimization method similar to bundle adjustment (BA) method. To solve the camera pose by SVD algorithm, the error term of point i should be defined first:

$$e_i = p_i - \left(Rp_i^{'} + t\right) \tag{10}$$

Then, the least square method is constructed to minimize the error square of R and t:

$$\min_{R,\ t} \frac{1}{2}\sum_{i=1}^{n} \left\| \left(p_i - \left(Rp_i^{'} + t\right)\right) \right\|^2 \tag{11}$$

Then define the two sets of points at the center of mass:

$$p = \frac{1}{n}\sum_{i=1}^{n}(p_i) \ , \ p^{'} = \frac{1}{n}\sum_{i=1}^{n} p_i^{'} \tag{12}$$

We treat formula (11) as follows:

$$\frac{1}{2}\sum_{i=1}^{n}\left\| p_i - \left(R_{p_i} + t\right)\right\|^2 = \frac{1}{2}\sum_{i=1}^{n}\left(\left\|p_i - p - R\left(p_i^{'} - p^{'}\right)\right\|^2 + \left\|p - Rp^{'} - t\right\|^2 + 2\left(\left(p_i - p - R\left(p_i^{'} - p^{'}\right)\right)^T \left(p - Rp^{'} - t\right)\right)\right) \tag{13}$$

Through equation (12), the objective function can be optimized as:

$$\min_{R,t} J = \frac{1}{2}\sum_{i=1}^{n}\left\| p_i - p - R\left(p_i^{'} - p^{'}\right)\right\|^2 + \left\|p - Rp^{'} - t\right\|^2 \tag{14}$$

Calculate the positions of the two groups of centers of mass:

$$q_i = p_i - p \ , \ q_i^{'} = p_i^{'} - p^{'} \tag{15}$$

Then, the rotation matrix is obtained as follows:

$$R^* = \arg\min_R \frac{1}{2}\sum_{i=1}^{n}\left\| q_i - Rq_i^{'} \right\|^2 \tag{16}$$

$$t^* = p - Rp^{'} \tag{17}$$

Expanding the error term of *R*:

$$\frac{1}{2}\sum_{i=1}^{n}\left\| q_i - Rq_i^{'} \right\|^2 = \frac{1}{2}\sum_{i=1}^{n}\left(q_i^T q_i + q_i^{'T} R^T R q_i^{'} - 2q_i^{'} R q_i^T\right) \tag{18}$$

Sorting out equation (18), we finally get:

$$\sum_{i=1}^{n} - q_i^T R q_i^{'} = \sum_{i=1}^{n} - tr\left(Rq_i^{'} q_i^T\right) = -tr\left(\sum_{i=1}^{n} q_i^{'} q_i^T\right) \tag{19}$$

The W matrix is:

$$W = \sum_{i=1}^{n} q_i q_i^{'T} \tag{20}$$

Decomposing W by SVD:

$$W = U \sum V^T \tag{21}$$

$\sum$ is composed of diagonal matrix singular value, the value of the diagonal matrix from big to small sort, type of U and V is a diagonal matrix. When W matrix is full rank, R can be obtained:

$$R = UV^T \tag{22}$$

Then t is calculated by $t^*=p-Rp^{'}$, and if the determinant of R is negative, then -R is the optimal value.

Continuous frame images are obtained through RGB-D camera, and relevant 3D points are obtained through feature matching. Then the pose transformation of the camera is calculated by ICP-SVD algorithm, and the pose transformation of visual AGV is finally obtained.

6. Experimental verification

6.1 Experimental environment and methods

The list of this equipment is shown in table 1 below:

Table 1: Equipment list

Equipment and software	Brand or software name	Specification
depth camera	Astra	30FPS、Depth: 0.6-4 m
computer	Dell	i7-7700、16GB RAM
operating system	Ubuntu16.04	ROS kinetic
visual software	Opencv	3.4.5
test car	Ruiqu	Quattro

In order to verify the effectiveness of the algorithm in this paper, the test car was used to carry the above equipment for linear motion, and the validity of the algorithm was tested by comparing the side deviation error of ICP-SVD algorithm in linear motion.

Figure 3: Test car

6.2 Feature extraction and matching optimization

This paper used the FAST algorithm for feature extraction, the results are as follows in figure 4, thus extracting effect is poor, most of the local feature points are very intense. In this paper, the method of image contrast adaptive threshold is used to homogenize the distribution of feature points.

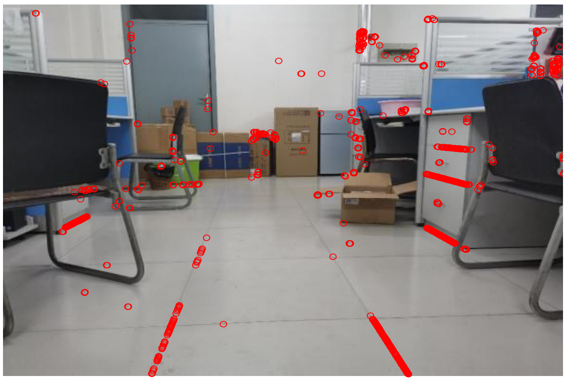

Figure 4: Feature extraction

As shown in figure 4, the image is gray-processed to reduce the amount of computation, and then processed by adaptive threshold to reduce the density of feature points and complete the homogenization of feature points.

Figure 5: Homogenization of feature points

After feature point extraction, to test the feature matching effect of continuous frame image, we directly used open source computer vision library to match ORB. In order to test the matching effect of ORB, we selected two graphs in the visual AGV operation. The two graphs were obtained based on the tiny movement generated by the camera. The matching effect is shown in figure 6:

Figure 6: Feature matching

After preliminary feature matching, because there are a many wrong matching feature points matching. In this paper, PROSAC algorithm was used to remove the wrong match, as is shown in figure 7. It is found from the figure that in the process of selecting matching point pairs based on ORB algorithm, after measuring the BRIEF descriptor with hamming distance, the feature matching obtained by PROSAC algorithm can efficiently screen out the wrong matching point pairs and get accurate results.

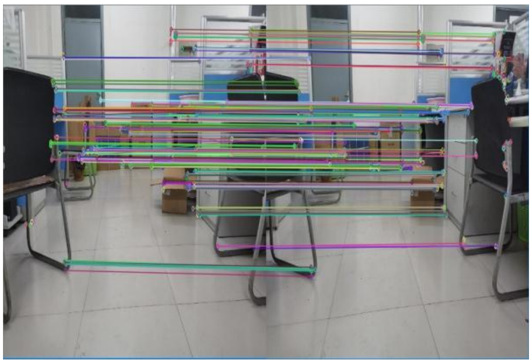

Figure 7: Remove error feature matching graph

6.3. Movement experiment

The experiment of this paper is based on the linear motion of Rikirobot, It is used to detect the position and pose estimation effect of visual odometer. The speed was set at 0.3*m/s* and the motion distance was 3*m*. ICP-SVD algorithm was used to complete the estimation of camera pose motion, so as to obtain the displacement of the robot. As shown in Figure 8 below, ICP - SVD algorithm has some good effects on camera motion pose, and the error range is maintained at 0.783%, which is within an acceptable range.

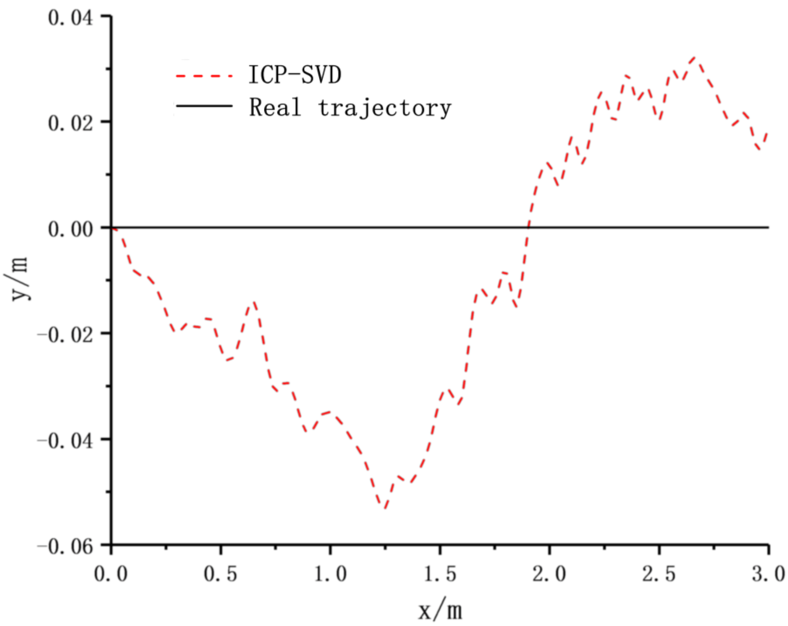

Figure 8: Linear motion

7. Conclusion

This paper uses Astra depth camera to obtain depth information of images and designs a visual odometer suitable for visual AGV. In the process of FAST algorithm to extract feature points of image frames, in the process of FAST algorithm to extract feature points of image frames, the problem of uneven distribution of feature points in feature extraction is effectively improved by using image contrast adaptive threshold. PROSAC algorithm can effectively remove errors in feature matching. The ICP-SVD algorithm is used to obtain the motion estimation between frames, and the motion pose of the visual AGV is obtained by combining the key frames, which can realize the high-precision autonomous positioning. The experiment proves that the algorithm in this paper can effectively estimate the pose of the car and play a basic role in the local map construction of the vision SLAM algorithm. Due to the limited experimental equipment, the effectiveness of the algorithm in the case of long distance is not verified, and the accumulated error is not taken into account, so there are some areas to be improved.

References

[1]. Scaramuzza, D., Fraundorfer, & Friedrich. (2011). Visual odometry: part I: the first 30 years and fundamentals. *IEEE Robotics & Automation Magazine,* 18(4): 80-92.

[2]. Kerl, C., Sturm, J., & Cremers, D. (2013). Robust odometry estimation for RGB-D cameras. *IEEE International Conference on Robotics and Automation,* 2013:3748-3754.

[3]. Whelan, T., Johannsson, H., Kaess, M., Leonard, J. J., & McDonald, J. (2013). Robust real-time visual odometry for dense RGB-D mapping. IEEE International Conference on Robotics and Automation, 2013:5724-5731.

[4]. Kerl, C., Sturm, J., & Cremers, D. (2013). Dense visual SLAM for RGB-D cameras. IEEE /RSJ International Conference on Robotics and Automation, 2013:2100-2106.

[5]. Gutierrez-Gomez, D., Mayol-Cuevas, Walterio., & Guerrero, J. J. (2016). Dense RGB-D visual odometry using inverse depth. Robotics and Autonomous System, 75:571-583.

[6]. Jaimez, M., Kerl, C., Gonzalez-Jimenez, J., & Cremers, D. (2017). Fast odometry and scene flow from RGB-D cameras based on geometric clustering. IEEE International Conference on Robotics and Automation, 2017:3992-3999.

[7]. Herbert, B., Andreas, E., Tinne, T., & Luc, V. G. (2008). Speeded-up robust features (surf). Computer Vision and Image Understanding, 110(3):346-359.

[8]. Lowe, D.G.(2004). Distinctive Image Features from Scale-Invariant eypoints. International Journal of Computer Vision, 60:91–110.

[9]. Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. R. (2011). ORB: an efficient alternative to SIFT or SURF. IEEE International Conference on Computer Vision, 2011:2564-2571.

[10]. Henry, P., Krainin, M., Herbst, E., Ren, X., & Fox, D. (2012). Rgb-d mapping: using kinect-style depth cameras for dense 3d modeling of indoor environments. The International Journal of Robotics Research, 31(5), 647-663.

[11]. Endres, F., Hess, J., Sturm, J., Cremers, D., & Burgard, W. (2014). 3-d mapping with an rgb-d camera. IEEE Transactions on Robotics, 30(1), 177-187.

[12]. Xiang, Q., Qiang, W., & Li, Y. U. (2017). Realization of visual odometry based on rgb-d camera. Journal of Zhejiang University of Technology, 45(6), 634-638.