Overview of Image Matching Based on ORB Algorithm

Overview of Image Matching Based on ORB Algorithm

Chuan Luo, Wei Yang, Panling Huang, *Jun Zhou

Department of Mechanical Engineering, Shandong University, Jinan 250061, China

zhoujun@sdu.edu.cn

Abstract. ORB image matching is of great significance in the field of image processing, which is mainly used in navigation, target recognition and classification, image stitching and remote sensing registration. Based on the existing literature on ORB algorithm research, this paper first introduces the classic ORB algorithm. Then, the various improved algorithms of ORB are expounded, and the performance index of image matching algorithm based on feature points is introduced. Finally, the shortcomings of the ORB algorithm are pointed out and the development direction of the algorithm is prospected.

1. Introduction

In recent years, with the rapid development of artificial intelligence, Simultaneous Localization and Mapping (SLAM) has become a key research direction in the field of robotics. The sensor of the visual SLAM is mainly a camera and it is important to roughly estimate the camera motion based on the information matching between adjacent images captured by the camera. For image matching, according to the difference of image information used in the matching process, it is mainly divided into feature point based matching, gray level based matching and transform domain based matching [1]. The feature point-based matching method has become a mainstream method for image matching because of its simple and fast calculation, high matching accuracy and insensitivity to grayscale, illumination, graphic distortion and occlusion. With the rapid development of image matching methods based on feature points, Lowe [2] proposed the famous Scale-Invariant Feature Transform (SIFT) algorithm in 1999, which fully considered the change of rotation, scale, lighting and noise and so on in image transformation. In 2006, Bay et al. [3] proposed an improved SIFT algorithm, the SURF (Speeded-Up Robust Features) algorithm, which improves the efficiency of computation; Rublee et al. [4] proposed the ORB (Oriented FAST and Rotated BRIEF) algorithm in 2011. Compared with SIFT and SURF, the algorithm further improves the computational efficiency and has strong real-time performance. It has received extensive attention in the current SLAM scheme.

In this paper, the image matching method based on ORB algorithm is reviewed, and its characteristics and existing problems are summarized. On this basis, the improved ORB algorithm is proposed, and its development trend is prospected. At the same time, the performance index commonly used evaluation feature point matching is introduced.

2. ORB algorithm

The ORB image matching algorithm is generally divided into three steps: feature point extraction, generating feature point descriptors and feature point matching. The specific flow chart is shown in Fig.1.
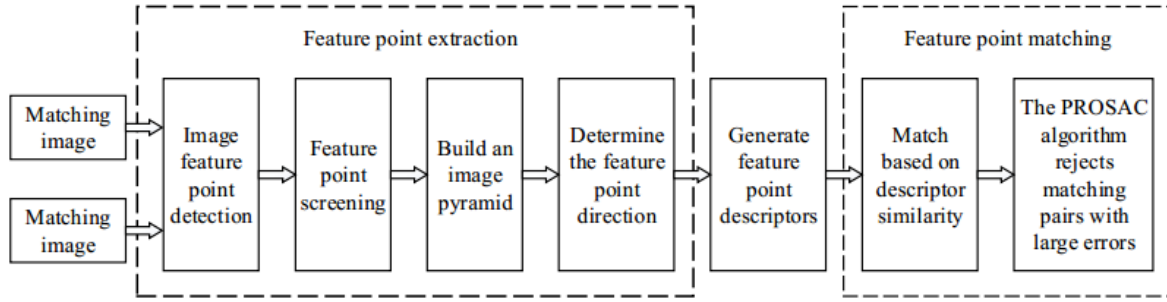
Fig.1 Image matching flow chart based on ORB algorithm

2.1. Feature point extraction

The ORB algorithm uses the improved FAST (features from accelerated segment test) [5] algorithm to detect feature points. The idea is that if a pixel is significantly different from the neighborhood pixels then it is more likely to be a corner point. The detection process is as follows:

(1) Image feature point detection. First, select the pixel p in the image and assume its brightness is I_p. Set a brightness threshold T. Then, take pixel p as the center, select 16 pixels on a circle with a radius of 3 and compare the gray value between pixel p and other pixels on the circle. If the bright-ness of consecutive N points on the selected circle is greater than $I_p + T$ or less than $I_p - T$, then pixel p can be considered as a feature point.

(2) Feature point screening. Since the calculation of the FAST corner point is only to compare the difference in brightness between pixels, the number is large and uncertain and there is no direction information. Therefore, the ORB algorithm improves the original FAST algorithm which calculates the Harris response values for the original FAST corner points and sorts them according to the gray value and take the first N points. Harris response value calculation formula is as shown in equation (1) and (2):

$$R = det(M) - k(trace(M))^2 \tag{1}$$

$$M = \sum w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \tag{2}$$

where R is the Harris response value, M is a 2×2 matrix, k ranges from 0.04 to 0.06[6], $w(x,y)$ is the image window function, I_x is the variation of the feature point in the horizontal direction, and I_y is the variation of the feature point in the vertical direction.

(3) Image scale pyramids are constructed and sampled on each layer of the pyramid to extract FAST features and add scale invariance to feature points.

(4) Determine the feature point direction. In order to make the extracted feature points have rotational invariance, the direction of the feature points is obtained by using the Intensity Centroid method [7]. First, in a small image block B, the moment of the image block is defined as

$$m_{pq} = \sum_{x,y \in B} x^p y^q I(x,y), \quad p,q = \{0,1\} \tag{3}$$

where x and y are pixel coordinates, and $I(x,y)$ is the gray value of the corresponding pixel. Then, find the centroid of the image block by the moment:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \tag{4}$$

where the 0th moment (m_{00}) is the mass of the image block and the 1st moment (m_{10}, m_{01}) is the centroid of the image block. Finally, the geometric center O and the centroid C of the image block are connected to obtain a direction vector $\overrightarrow{OC}$ and the direction of the feature point is defined as:

$$\theta = arctan \left(\frac{m_{01}}{m_{10}} \right) \tag{5}$$

Through the above steps, the FAST corner points have scale invariance and rotation invariance, which greatly improves their robustness in different images.

2.2. Generate feature point descriptors

After extracting the Oriented FAST feature points, the ORB algorithm uses the improved BRIEF algorithm [8] to calculate the descriptors for each point. BRIEF is a binary vector descriptor whose vector consists of a number of 0 and 1:

$$\tau(p; x, y) = \begin{cases} 1, p(x) < p(y) \\ 0, p(x) \geq p(y) \end{cases} \tag{6}$$

where $p(x)$ is the gray value at the field x around the image feature point, and $p(y)$ is the gray value at the field y around the image feature point. To reduce the effects of noise, Gaussian filtering is first performed on the image. Based on the feature point p taken as the central point, take a neighborhood window of size $S \times S$ and randomly select N (N is usually taken as 256) pairs of pixels in the neighborhood window. Then, the brightness value of each pair of points is compared according to equation 5 and binary assignment is performed. Finally, an N-dimensional vector consisting of N binary strings is obtained:

$$f_N(p) = \sum_{1 \leq i \leq N} 2^{i-1} \tau(p; x_i, y_i) \tag{7}$$

Since the original BRIFE descriptor does not have rotation invariance, it is easy to lose data when the image is rotated. Therefore, the ORB algorithm uses the Steer BRIEF algorithm to calculate the main direction of each feature point, so that the descriptor has direction information. A rotation matrix R_θ is obtained:

$$R_\theta = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \tag{8}$$

Matrix R_θ and N pairs of pixel points form a matrix Q:

$$Q = \begin{bmatrix} x_1, x_2, \cdots, x_N \\ y_1, y_2, \cdots, y_N \end{bmatrix} \tag{9}$$

Then a rotation correction is performed to get Q_θ:

$$Q_\theta = R_\theta Q \tag{10}$$

Finally, we can get a directional descriptor:

$$g_{N(p,\theta)} = f_N(p) | (x_i, y_i) \in Q_\theta \tag{11}$$

2.3. Feature point matching

After determining the scale and rotation information of the image feature points, it is necessary to determine the similarity between the feature point descriptors in the two different time images to determine whether they match. Suppose that feature point $x_t^m, m = 1,2, \cdots, M$ is extracted in image I_t, and feature point $x_t^n, n = 1,2, \cdots, N$ is extracted in image I_{t+1}. The easiest way to match is Brute-Force Matcher. The method measures the distance between each feature point x_t^m and all x_{t+1}^n measurement descriptors and then chooses the nearest one as the matching point. For a binary BRIEF descriptor, the Hamming distance is used to measure the number of different characters between two equal-length strings.

In the visual SLAM process [9], the ORB algorithm uses the FLANN (Fast Library for Approximate Nearest Neighbors) algorithm to match and establish multiple random KD trees. A plurality of dimensions is randomly selected from the dimension with the highest variance in the data set. The data set is divided by this method to quickly search for the most similar feature points of the matching information. This method is more computationally efficient than the Brute-Force Matcher. In addition, the PROSAC algorithm is generally used to eliminate some matching pairs with large matching errors, which further improves the accuracy of matching.

3. Improved ORB algorithm

The concept of the ORB algorithm was proposed in 2011 and has been developed for less than a decade. There are not many researches on improved ORB, mainly the following:

Xiaohong Li et al. [10] proposed a fast target detection algorithm based on ORB in the scene of detecting dynamic targets. In order to meet the real-time requirements, an eight-parameter rotation model was used to solve the global motion parameters in combination with the least squares method.

Finally, the frame difference method was used to obtain the moving target. The algorithm not only maintains the superiority of SIFT and SURF, but also improves the detection speed and can accurately detect moving targets in real time.

Yunsheng Zhang et al. [11] proposed a feature extraction method based on grid filtering for the problem of non-uniformity of ORB feature extraction and combined RANSAC method to improve the matching accuracy.

Kaiting Zhou et al. [12] proposed a multi-pose face recognition method based on improved ORB feature and improved the sampling mode of descriptors. By selecting multiple template samples and extracting the same number of key points for different individuals, this method reduced mutual interference between individuals and improved the robustness of features to face angle changes.

Zhuqing Hu et al. [13] proposed an improved ORB descriptor: Gravity-ORB, which is mainly used for mobile device detection such as mobile phones. The method uses gravity acceleration sensor to calculate the feature point direction angle, which simplifies the feature extraction step and improves the computational efficiency of the algorithm while maintaining ORB robustness.

Jiawang Bian et al. [14] proposed a feature optimization algorithm based on grid motion statistics and transformed the motion smoothing constraint problem into statistical measurement. At the same time, they proposed an efficient mesh-based fractional estimator to achieve false matching rejection.

However, the above studies have not solved the problem that the accuracy of the ORB algorithm is reduced when the scale of the image of a large external environment changes greatly. Based on the original ORB image registration method, Yanyan Qin et al. [15] combined the original SIFT method with the ORB method and proposed the SIRB (SIFT and ORB) algorithm. SIRB has solved the defect of ORB scale inconsistency while maintaining the advantage of ORB in matching speed.

Mur-Artal et al. [16] proposed an improved ORB algorithm based on quadtree. By performing quadtree partitioning on all feature points, the feature extraction is more uniform and the detection matching effect is more accurate. But on the other hand, it also indirectly weakens the efficiency of feature extraction.

4. Algorithm performance evaluation index

The performance evaluation of feature matching algorithm is the basis for judging the pros and cons of its algorithm. Because of the different research fields, different purposes, and different application scenarios of image matching, it is difficult to judge the performance of an algorithm with a unified standard. Therefore, it is usually necessary to use diverse indicators to process the results of image matching and comprehensively evaluate them, thereby selecting an algorithm with superior comprehensive performance [17-18]. Commonly used algorithm performance evaluation indicators are precision, recall and matching score etc.

4.1. Precision

Precision refers to the proportion of feature points that are correctly matched among all the matched feature points in the total matching. The calculation formula is as follows:

$$P_{precision} = \frac{N_{correct}}{N_{all}} \tag{12}$$

Precision is also affected by matching criteria. The stricter the matching criteria is, the greater the number of correct matches is and the higher the accuracy.

4.2. Recall

Recall refers to the proportion of the feature points that are correctly matched among all the feature points which should be matched. The calculation formula is as follows:

$$P_{recall} = \frac{N_{correct}}{N_{should}} \tag{13}$$

4.3. Matching Score

Matching Score refers to the proportion of the feature points that are correctly matched among all the feature points and is also affected by the matching criteria. The calculation formula is as follows:

$$P_{ms} = \frac{N_{correct}}{N_F} \tag{14}$$

In the performance evaluation of the algorithm, the Precision-Recall curve is usually adopted, which can be obtained by changing the threshold of the matching distance and changing the matching criterion of the algorithm.

5. Conclusion

This paper describes the general process of image matching based on ORB algorithm and introduces some improved methods of ORB in recent years. Finally, it introduces several commonly used feature-based image matching performance evaluation indicators. Although the ORB algorithm has strong real-time performance and has a great improvement in computing speed, it still has many shortcomings. At present, the ORB algorithm is widely used in SLAM and the accuracy of image matching is relatively high. It can be seen that the future development of ORB mainly focuses on the following points:

(1) For image extraction features, it is more stable and has higher anti-interference;

(2) Use a more efficient algorithm to eliminate the effects of extraneous features and further improve the matching speed;

(3) ORB is combined with other algorithms to make its performance even better.

References

[1]   Moon Y S, Loh W K. Triangular inequality-based rotation-invariant boundary image matching for smart devices[J]. Multimedia Systems, 2015, 21(1):15-28.

[2]   David G Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol.50, No. 2, 2004, pp.91-110.

[3]   Bay H , Ess A , Tuytelaars T , et al. Speeded-Up Robust Features (SURF)[J]. Computer Vision and Image Understanding, 2008, 110(3):346-359.

[4]   Rublee E , Rabaud V , Konolige K , et al. ORB: An efficient alternative to SIFT or SURF[C]// 2011 International Conference on Computer Vision. IEEE, 2012..

[5]   Rosten E , Porter R , Drummond T . Faster and better: a machine learning approach to corner detection[J]. IEEE Trans Pattern Anal Mach Intell, 2008, 32(1):105-119.

[6]   Xingteng J , Xuan W , Zhe D . Image matching method based on improved SURF algorithm[C]// IEEE International Conference on Computer & Communications. IEEE, 2016.

[7]   Rosin P L . Measuring Corner Properties[M]. Elsevier Science Inc. 1999

[8]   Calonder M, Lepetit V, Strecha C, et al. BRIEF: Binary Robust Independent Elementary Features[J]. Computer Vision-Eccv 2010, Pt Iv, 2010, 6314:778-792.

[9]   Kerl C, Sturm J, Cremers D. Dense visual SLAM for RGB-D cameras[C]// IEEE/RSJ International Conference on Intelligent Robots & Systems. 2014.

[10] Li X H, Xie C M, Jia Y Z, Zhang G F. Fast target detection algorithm based on ORB feature[J], Journal of Electronic Measurement and Instrument, 2013, 27(5):455-460.

[11] Zhang Y S. Automatic registration method for remote sensing images based on improved ORB

algorithm,2013,25(03):20-24.

[12] Zhou K T, Zheng L X. Multi-pose face recognition based on improved ORB feature[J]. Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao/Journal of Computer-Aided Design and Computer Graphics, 2015, 27(2):287-295.

[13] Hu Z , Jiang Y . An improved ORB, gravity-ORB for target detection on mobile devices[C]// Intelligent Control & Automation. IEEE, 2016.

[14] Bian J , Lin W Y , Matsushita Y , et al. GMS: Grid-Based Motion Statistics for Fast, Ultra-Robust Feature Correspondence[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Computer Society, 2017.

[15] Qin Y , Xu H , Chen H . Image feature points matching via improved ORB[C]// International Conference on Progress in Informatics & Computing. IEEE, 2014.

[16] Mur-Artal R , Montiel J M M , Tardos J D . ORB-SLAM: a Versatile and Accurate Monocular SLAM System[J]. IEEE Transactions on Robotics, 2015, 31(5):1147-1163.

[17] Mikolajczyk K , Tuytelaars T , Schmid C , et al. A Comparison of Affine Region Detectors[J]. International Journal of Computer Vision, 2005, 65(1-2):43-72.

[18] Mikolajczyk K , Schmid C . A performance evaluation of local descriptors[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(10):1615-1630.