

Circle Center Automatic Extraction and Sorting based on Improved Circular Target

Shijie Deng^{*}, Jiang You, Liwei Tang, and Xujun Su

Department of Artillery Engineering, Army Engineering University, Shijiazhuang, 050003, China

Abstract

In order to solve the problems that the traditional circular target is not convenient for automatic camera calibration, the accuracy of circle centering is not high, and the circle centering array is greatly influenced by the rotation angle of the circular target, a circular target suitable for automatic camera calibration was designed. For this target, a least squares center-fitting method based on random sampling and a centroid lattice sequential sorting method based on vector angle were proposed. The experimental results showed that the average centering error was 0.0061 pixels using this center extraction algorithm. Compared with the traditional least squares fitting method, the accuracy was higher; for the calibration target 0-360 degree rotation, it could more quickly and accurately complete the center-point lattice sorting. The average sorting time was 0.0096s, and it was easier to implement on-line camera automatic calibration.

Keywords: center-fitting; centered lattice sorting; random sampling uniform; least squares

(Submitted on November 22, 2018; Revised on December 18, 2018; Accepted on February 21, 2019)

© 2019 Totem Publisher, Inc. All rights reserved.

1. Introduction

Machine vision-based spatial measurement technology has been widely used in the industry, and camera calibration is a key factor affecting accuracy [1]. Zhang proposed a calibration method based on planar calibration target. This method completes camera calibration through the relationship between the coordinate points corresponding to the plurality of images using a nonlinear optimization algorithm [2], which is simple, practical, and has high precision. However, during the use of the method, the feature points must be accurately extracted, and the feature points must be arranged in a certain order.

The Matlab camera calibration toolkit [3] based on Zhang's calibration method is widely used, but in the course of its use, in order to complete detection and sorting, it is necessary to manually click on the four boundary points of the calibration image. There is a large naked eye error, which increases the calibration workload. In addition, the OpenCV Calibration Toolbox [4] also has related corner detection and sorting functions. However, in cases where the deflection angle is large, there will be sorting confusion that will affect the camera's automatic calibration. Therefore, it is necessary to study reliable and high-accuracy corner detection and sorting algorithms.

Many researchers have improved the design of the classic black and white checkerboard to achieve automatic detection and sorting of corner points [5-8]. However, the checkerboard itself has limitations, in which good quality images are required and the corner extraction accuracy is easily affected by noise [9]. On the contrary, circular targets have lower requirements on image quality and have better practical value. In paper [10], by setting five azimuth circles in the calibration target, the automatic detection and matching of the marker points were completed, but the entire sorting process was more complicated. Sorting disorder would occur when the distortion was large. Paper [11] proposed a design scheme of the planar calibration plate characterized by gradual circles, and the detection and sorting algorithm of feature points were given. However, when the calibration target plane was more than 90 degrees from the camera optical axis, the automatic sorting algorithm would fail.

^{*} Corresponding author.

E-mail address: dsj_sjz@163.com

In view of this situation, this paper added the rectangular frame and the boundary circle to the traditional circular target to complete the automatic calibration of the camera. When the center of the circle was fitted, the idea of random sampling was added to the least squares method to eliminate the error. The edge points improved the accuracy of the center of the circle. When sorting the center points, the ordering method of the center point lattice based on the vector angle was proposed. By setting the floating reference, the automatic sorting of the feature points at the arbitrary rotation angle of the target was achieved. The experimental results showed that the algorithm of circle center automatic extraction had high precision, and the center-centered lattice sort algorithm had rotation invariance and high operating efficiency.

2. Improved Design of Calibration Target

Traditional circular calibration targets are not convenient for automatic extraction in complex backgrounds and require manual determination of boundary points. Therefore, in order to realize the automatic segmentation and extraction of the calibration target and the automatic determination of boundary points, the final improved circular target looks like Figure 1.

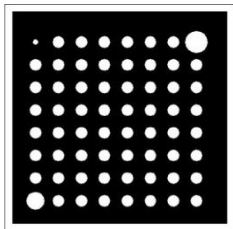


Figure 1. Target structure

Two improvements compared to traditional round targets:

- (1) A 10mm rectangular frame is added to facilitate the automatic segmentation and extraction of the calibration target by detecting the area and squareness of the connected area and setting the threshold.
- (2) The radius of the three boundary circles is 4mm, 12mm, and 15mm in order of the upper left corner, the lower left corner, and the upper right corner. The radius of the remaining circles is 8mm so that the boundary points can be determined by detecting the number of pixels of the characteristic circle. The boundary point can be used as the azimuth reference to achieve automatic sorting of the center point lattice.

3. Feature Point Detection Algorithm

3.1. Characteristics Analysis of Feature Points

The circular target generally uses the center of the characteristic circle as a feature point to complete camera calibration. In order to ensure the extraction accuracy of the feature points and the stability of the automatic sorting of the center point lattice, this paper chooses the method of fitting the edge pixels to obtain the center of the circle. For the processing flow in Figure 2, the following three improvements have been made:

- (1) Gaussian fitting is used to obtain the sub-pixel position of the edge pixels and improve the edge detection accuracy;
- (2) Join the idea of random sampling, eliminate marginal pixel coordinates with large deviations, and improve fitting accuracy;
- (3) The method of sorting center point based on the vector angle is proposed. By setting a floating reference, the interference of camera distortion and other factors on the sorting algorithm is solved.

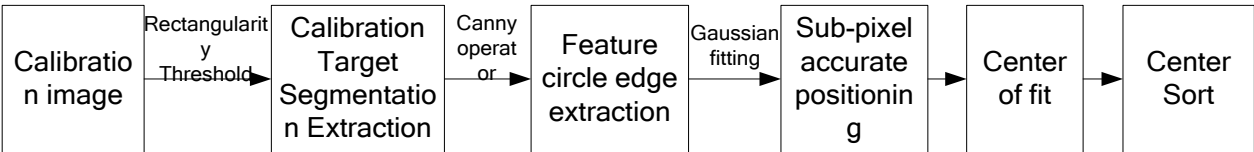


Figure 2. Process flow

3.2. Feature Circle Edge Extraction

For the automatically segmented calibration target, the canny operator is used to initially extract the characteristic circle edge. In order to more accurately position its edges, in the gradient direction of the edge points, the gray-scale interpolation method is used to acquire the coordinates and gray values of several neighboring points. Based on the Gaussian fitting method, the coordinates of the extreme values of the gray-scale variation are obtained, that is, the sub-pixel coordinates of edge points.

Let any of the features' circular edge point coordinates be denoted as W_i . As shown in Figure 3, taking the edge point P in W_i as an example, L is the gradient direction line at point P , k is the slope of the line at the point P in the gradient direction, and the horizontal and vertical whole pixels near the line L and the edge point P are calculated respectively. Find the coordinates of the intersection, calculate its distance from the point P , and pick the last eight points.

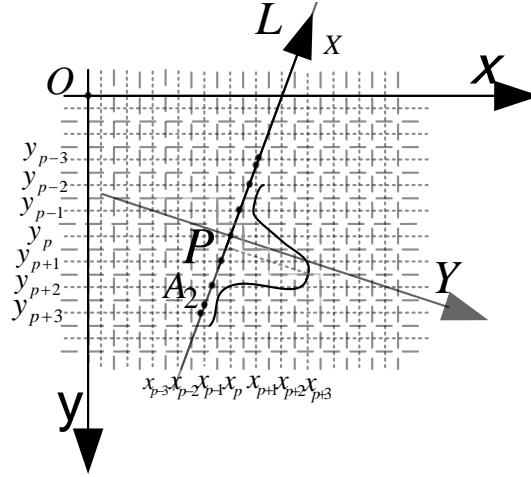


Figure 3. Schematic diagram of the neighboring edge

Set the linear L equation to $y = k(x - x_p) + y_p$. Take point A_2 as an example.

$$\begin{cases} x_2 = x_p - 1 \\ y_2 = k(x_2 - x_p) + y_p \end{cases} \quad (1)$$

The gray value of A_2 is obtained by linear interpolation using the gray values of A_2 in the two adjacent pixels in the y direction.

$$f(x_2, y_2) = (1 - \lambda) \times f(x_2, [y_2]) + \lambda f(x_2, [y_2] + 1) \quad (2)$$

Where $\lambda = y_2 - [y_2]$, $f(x, y)$ is the gray value of pixel coordinates (x, y) , and $[y_2]$ represents the integer part of y_2 . Similarly, the gray values of points $A_1, A_3, A_4, A_5, A_6, A_7$, and A_8 can be found.

Since the gray value of the true edge conforms to the Gaussian curve, the mean value of the difference before and after is used as the amount of gray change, and the fitting is performed [12]. Taking point P as an example, the pixel coordinate is (x_p, y_p) , the gray value is $f(x_p, y_p)$, the previous point A_4 pixel coordinate is (x_4, y_4) , the gray value is $f(x_4, y_4)$, the latter point A_5 pixel coordinate is (x_5, y_5) , the gray value is $f(x_5, y_5)$, and then the point P is a gray degree value.

$$f_0 = \frac{f_1 + f_2}{2} = \frac{f(x_p, y_p) - f(x_4, y_4)}{2} + \frac{f(x_5, y_5) - f(x_p, y_p)}{2} \quad (3)$$

It is the same for the $A_1, A_2, A_3, A_4, A_5, A_6, A_7$, and A_8 gray difference. As shown in Figure 4, a new coordinate system is

established. The x -axis is the pixel coordinate of the neighboring point, and the y -axis is the gray difference value of the neighboring point. Among them, the point P is a pixel-level edge point, and the gray-level difference value fits the maximum value of the curve corresponding to the pixel coordinate, that is, the sub-pixel edge point.

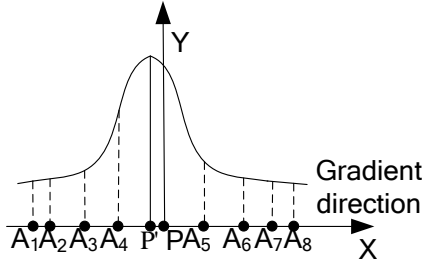


Figure 4. Gray difference value of neighboring points

In the same way, all sub-pixel edge point coordinates of the feature circle can be finally obtained.

4. Least Square Fitting Center based on Uniformly Randomized Samples

The coordinates of the sub-pixel edge points of the feature circle obtained in Section 3.2 are often obtained by least squares fitting to obtain the center of the circle. Since the traditional least squares method is based on the optimal estimation method derived from the maximum likelihood method when the random errors are evenly distributed, the actual acquired sub-pixel edge points may have large error points, and direct fitting may cause large errors. Therefore, the idea of random sampling consistency is added, large error points are eliminated, and the least square method is improved.

4.1. Establishing an Objective Function

According to projection theory, the characteristic circle is elliptical after projection, so the elliptic equation of the edge point is

$$ax^2 + bxy + cy^2 + dx + ey + f = 0 \quad (4)$$

The center coordinates are

$$\begin{cases} x_c = \frac{be - 2cd}{4ac - b^2} \\ y_c = \frac{bd - 2ae}{4ac - b^2} \end{cases} \quad (5)$$

In order to avoid zero solution and consider any integral multiple of the solution as the same solution, join the constraint condition and establish the objective function according to the principle of least squares:

$$f(a, b, c, d, e) = \sum_{k=1}^n (ax_k^2 + bx_k y_k + cy_k^2 + dx_k + ey_k + f) \quad (6)$$

The parameters a, b, c, d, e , and f are determined by the minimum value of the objective function.

4.2. Consistent Random Sampling

Taking into account the extracted feature circle edge point, there may be a larger point of error that will affect the accuracy of the fitting. Therefore, join the idea of random sampling, remove large error points, and then use least squares fitting.

The random sampling consensus algorithm is based on the framework of hypothesis and verification [13-14]. Randomly extract a certain number of data points from the data point set to determine the parameters of the estimated model, estimate the model parameters, and test the model with all points. Calculate the support degree of the model (the number of data points whose relative error is lower than the set threshold under this model), cycle k times, and select the model with the highest support degree as the final parameter model.

- (1) Support: The number of data points whose relative error is lower than the set threshold α under this model;
- (2) Inliers: Data points where the relative error under this model is lower than the set threshold;
- (3) Outliers: Data points whose relative error is higher than the set threshold α under this model;

(4) Confidence: The probability of randomly selecting all the points that make up the sample are internal points, denoted as η_0 .

It can be seen that, assuming that the occupancy rate of the inner point of the edge point is ε , the minimum number of points needed to determine the elliptic parameter is m , the number of random samplings is k , and the probability that all the m inner samples are not obtained is $(1 - \varepsilon^m)^k$.

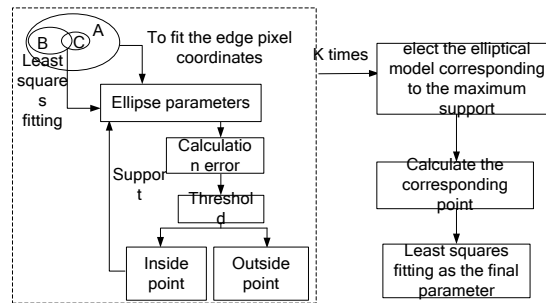
$$(1 - \varepsilon^m)^k = 1 - \eta_0 \quad (7)$$

Then,

$$k = \frac{\log(1 - \eta_0)}{\log(1 - \varepsilon^m)} \quad (8)$$

4.3. Algorithm Steps

- (1) Set the confidence level η_0 (this article takes 80%) and the internal point occupancy rate ε (this article takes 95%). The calculation requires the number of sampling k ;
- (2) In the edge lattice, m data points that can estimate the ellipse parameters are randomly selected (m is 8 in this paper), and the ellipse parameters are initially calculated;
- (3) Test the ellipse model with all the edge points, that is, calculate the relative error with the ellipse, and determine the support degree of the ellipse (the number of edge points whose error is lower than the set threshold; this article takes α as 0.015);
- (4) Cycle steps (2)-(3) k times;
- (5) Select the larger support in k elliptic models as the optimal model, and calculate all interior points satisfying the ellipse of the parameters as the final ellipse fitting data point set;
- (6) Fit all interior points of the optimal model based on the least squares method, obtain the final ellipse model, and calculate the relevant parameters. The fitting processing flow is shown in Figure 5.



A is the offset smaller edge pixel coordinates; B is the offset larger edge pixel coordinates; C is randomly selected to fit the edge pixel coordinates

Figure 5. Fitting process

5. Ordering in the Center of the Circle based on the Included Angle

The correct matching of feature points is the key to achieving automatic camera calibration, directly determining the

accuracy of the calibration results. When performing target circle edge detection, it is detected from left to right and top to bottom according to the entire picture. Therefore, the coordinates of the center-point lattice obtained by the final fitting are disorderly and unmatched with the physical coordinates of the characteristic circle. In order to complete the automatic calibration of the camera, the pixel coordinates of the center point lattice must be automatically sorted.

5.1. Boundary Point Determination

From the physical structure of the target, the three boundary circles and the remaining feature circles have different sizes. Therefore, regardless of the attitude of the target and the effects of camera distortion and uneven illumination, the boundary points can be determined according to the number of pixels occupied by the characteristic circle, which are denoted as A , B , and C .

5.2. Feature Point Sorting Algorithm

Due to the problems of mirror distortion and uneven illumination, the pixel coordinates of the center point array actually extracted do not have the same interval. From the camera distortion model, it can be seen that the peripheral dot pitch distortion is large. Therefore, based on theoretical calculations, the traditional method of setting the threshold method will cause disorder in the case of large distortion.

Therefore, this paper sets a floating benchmark and proposes a sequential sorting method based on the angles between the vectors. Firstly, according to the boundary point, the thresholds of the angles between the boundary vectors are used to find the feature points on the boundaries AB and AC , and the Euclidean distance from the point A is sorted to achieve the sorting of feature points on the boundary line. Furthermore, based on the AC and AB corresponding position points, the reference position angle threshold criterion is used to find the corresponding position feature point. Using the newly detected feature points as a reference, the next point is searched, and the matched points are removed from the lattice in time to improve the search efficiency until the ordering of the lattices is completed. The schematic diagram of feature array is shown in Figure 6. The specific steps are as follows:

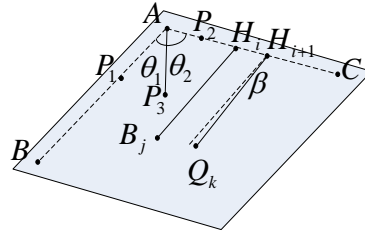


Figure 6. Schematic diagram of feature array

- (1) Determine the boundary points A , B , and C , and calculate the angle of vectors \mathbf{AB} and \mathbf{AC} ;

$$\theta = \arccos \frac{(x_B - x_A)(x_C - x_A) + (y_B - y_A)(y_C - y_A)}{\sqrt{(x_B - x_A)^2 + (y_B - y_A)^2} \cdot \sqrt{(x_C - x_A)^2 + (y_C - y_A)^2}} \quad (9)$$

- (2) Find the feature points of boundary lines AB and AC , and sort them;

① Take any point (excluding A, B, C) in the center point and set point P , and calculate the angle of vectors \mathbf{AP} and \mathbf{AB} , \mathbf{AC} respectively, recorded as θ_1, θ_2 ;

② Judging if $\theta_1 \leq \alpha$ and $|\theta - \theta_2| \leq \alpha$, it is on the boundary AB , such as point P_1 . Enter L_1 ;

Judging if $\theta_2 \leq \alpha$ and $|\theta - \theta_1| \leq \alpha$, it is on the boundary AC , such as point P_2 . Enter H ;

Otherwise, it is a non-boundary point, such as point P_3 . Enter M ;

③ Calculate the Euclidean distance between the L_i and H points and A respectively, and arrange the coordinates of the feature points from smallest to largest;

④ The coordinates of point A are entered in H and are in the first position;

(3) Find the column by referring to the feature point on the boundary line AC , feature points, and sorting;

① Take points H_i and H_{i+1} , ($i = 1$) separately;

② Take L_i as a reference column, and take point $L_{i,j}$ ($j = 1, 2, \dots, 7$), denoted as B_j (i.e., point $j+1$, row i);

③ Any point in the unsorted lattice M is set to Q_k ($k = 1, 2, \dots, \text{length}(M)$);

④ Calculate the angle of vectors $\mathbf{H}_i\mathbf{B}_j$ and $\mathbf{H}_{i+1}\mathbf{Q}_k$, denoted as β ;

If $\beta \leq \alpha$ and $\|\mathbf{H}_i\mathbf{B}_j\| - \|\mathbf{H}_{i+1}\mathbf{Q}_k\| \leq \gamma$, then point Q_k is the position of point j in row $i+1$, which is denoted as $L_{i+1,j}$;

⑤ Until $j = 7$, we find all row positions in column $i+1$, that is, L_{i+1} ;

⑥ Unsorted, remove the corresponding point L_{i+1} . Order $i = i+1$, and repeat step ①;

(4) Final sorting: The first element of each column is recorded in all points in the matrix.

The target before and after target feature lattice sorting is shown in Figure 7.

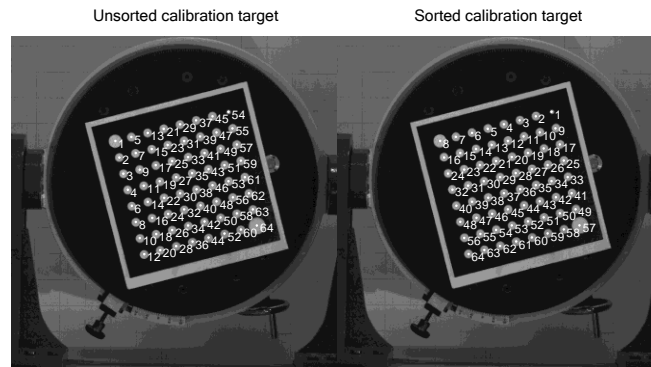


Figure 7. Before and after the sorting of the characteristic lattice

6. Experimental Verification and Analysis

6.1. Experimental System Composition and Procedure

The experimental system is mainly composed of a computer, an area array CCD, an image acquisition equipment Core, a solid circular array target 8×8 , and a CZJC_01 angle measuring instrument. The CCD pixel size is $5.5\mu m \times 5.5\mu m$, and the number of pixels is 2048×2048 . The focal length of the lens is 65-80mm. The camera transmits the core of the recording acquisition device through the 26pins camera interface line. Core exchanges data with the computer through the e-SATA interface.

(1) Connect the computer, the acquisition device Core, and the camera correctly and fix the target in the CZJC_01 corner measuring instrument;

(2) Open the computer and run the CoreView recording software to set the initial camera parameters so that the target image is clear;

- (3) The control rotation angle measuring instrument rotates along the x-axis and z-axis directions respectively, drives the target to rotate, and simultaneously stores and stores calibration images;
- (4) Automatically segment the target, extract the edges of the feature sub-pixels, and fit the center of the circle using a least squares method based on random sampling;
- (5) Use the ordering method based on the angles of the vectors to complete the centering array;
- (6) Analyze the accuracy of the circle center extraction, the timeliness of the sorting algorithm, and the invariance of the calibration target rotation.

6.2. Comparison of Center Accuracy

According to the method shown in [12], the CCD imaging principle is simulated to generate an ideal simulated ellipse image, and the theoretical center-point coordinates are (x_c, y_c) . According to the sub-pixel localization algorithm in this paper, the sub-pixel coordinate of the circle edge is detected and denoted as $P_i = (x_i, y_i)$, $i = 1, 2, \dots, n$. Sampling the traditional least squares method and RANSAC-based least squares fitting edge respectively, acquiring the ellipse parameters, and calculating the circle center coordinates, the traditional algorithm center circle extraction error is 0.0103 pixels, while the traditional algorithm is 0.0142 pixels. According to the ellipse parameters obtained by the two methods, the error in the entire ellipse circumference is calculated relative to the theoretical ellipse circle, as shown in Figure 8.

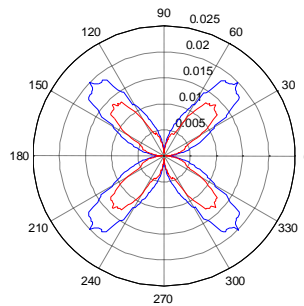


Figure 8. Fit error distribution on a 360-degree circle

In Figure 8 above, the radius direction is the error (unit: pixel) relative to the theoretical circumference of the fitted ellipse, where red is the RANSAC-based least squares method and blue is the traditional least squares method. It can be seen that the two methods have larger errors in the directions of 45° and 135° , mainly due to the large radius of curvature of the curved edge in this direction and the relatively large errors in the coordinates of the detected edge pixels. Compared to traditional algorithms and improved fitting algorithms, the relative error is relatively small, mainly due to the addition of a random sampling uniform algorithm before the fitting, which removed the edge pixel coordinates with relatively large relative edge deviations and improved the to-be-fitting sample accuracy.

Using the center of the paper's automatic extraction algorithm for the same image, with the OpenCV Calibration Toolbox, the center of the circle is extracted and the resulting centerline lattice error is shown in Figure 9(a). Calculate the sum of 64 center absolute error distances on each calibration target image, as shown in Figure 9(b).

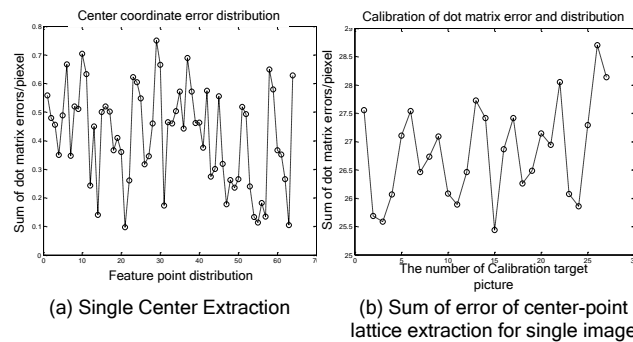


Figure 9. The center of the circle error distribution

6.3. Sorting Algorithm Efficiency and Adaptive Verification of Template Rotation

In this paper, the improved calibration target and the corresponding sorting algorithm have the adaptability of rotation sorting. In the scene of 0-360°, the calibration targets with different angles, different directions, and different illumination conditions are taken, and 25 calibration images are randomly selected for testing. The detection and sorting results are shown in Figure 10.

As can be seen from the Figure 10, since the floating reference is used in this paper, the newly detected feature points are used to update the benchmark in real time, which reduces the interference of factors such as distortion. The vector angle is used as the search index. Compared with the distance index, noise and other disturbances are reduced. In the case where the calibration target rotates at a relatively large angle or the light is dark, the automatic sorting of the feature points can be accurately performed, and it has good adaptability.

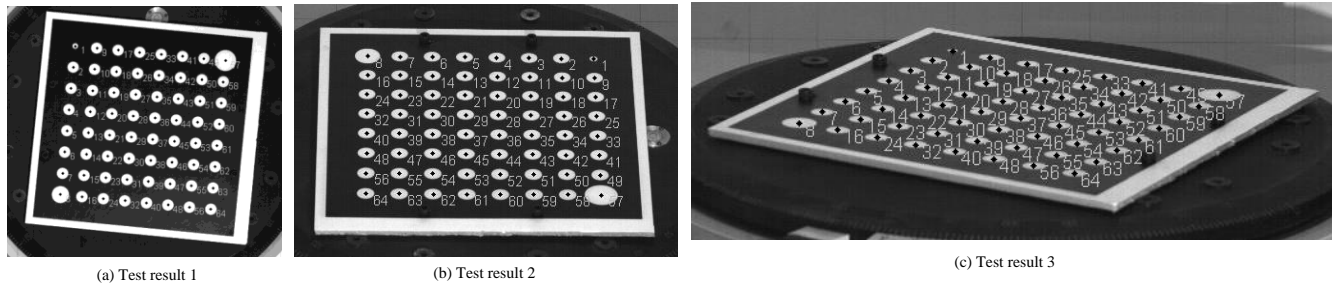


Figure 10. Corner detection and sorting results at different rotation angles

6.4. Time-Consuming Analysis of the Center Extraction and Sorting Algorithm

Randomly select 25 calibration target images in different poses, and follow the process of automatic extraction and sorting of the circle center shown in Figure 2. The processing steps, time consumption, and required operation times are shown in Table 1.

Table 1. Steps, time consumption, and frequency of calibration target centering and sorting

Steps	Time/s	Count/time
Calibration target segmentation	0.0054	1
Feature circle edge extraction	0.0064	64
Sub-pixel positioning	0.0862	64
Center-fitting	0.1324	1
Center sort	0.0096	1
The whole process	6.0738	1

For 25 calibration target images of different poses, the algorithm of this paper and the sorting algorithms in [6] and [10] are used respectively to sort the centered lattice obtained. The time-consuming ordering is shown in Figure 11.

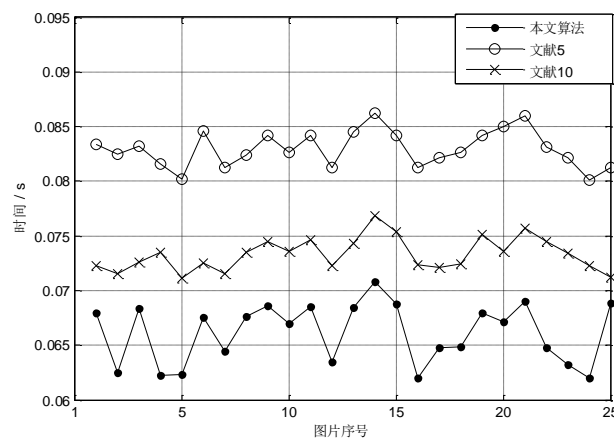


Figure 11. Comparison of sorting algorithm time

As can be seen from the above figure, the average sorting time of the sorting algorithm in this paper is 0.0096s. Compared with the other two methods, the sorting efficiency has been greatly improved. [6] In the detection process, it is necessary to combine the camera imaging model to calculate the corresponding focal length and increase the amount of calculation. Literature [10] needed to detect five marker points, the sorting had to be compared with five marker points, and the calculation amount was large. This paper only needs to set three marker points. When the search is sorted, it only needs to compare with the reference vector and updates the unsorted feature points in real time, which reduces the sample space for search ordering and improves the sorting efficiency.

7. Conclusions

On the basis of traditional circular targets, a rectangular frame and a boundary circle were added to facilitate the automatic segmentation of the calibration target and the determination of boundary points, thereby realizing the automatic calibration of cameras. The problems that the accuracy of the conventional circle center extraction is not high and the center-centered array is strongly influenced by the rotation angle of the calibration target were addressed. A least squares center-fitting algorithm based on random sampling and a center-centered sorting method based on vector angles were proposed. Experimental results showed the following:

- (1) Based on the random least squares center-fitting algorithm with uniform sampling, the average fitting error was 0.0061 pixels, which demonstrated high precision;
- (2) Based on the vector-angle center-order sorting method, by setting the floating reference, the calibration target feature points of any rotation angle could be automatically ordered with rotation invariance, the average sorting time was 0.0096s, and the efficiency was high.

Acknowledgments

We would like to thank the reviewers for their valuable comments. This work was partially funded by the National Natural Science Foundation of China and the Military Research Foundation of China.

References

1. F. F. Gu, H. Zhao, and P. H. Bu, "Analysis and Correction of Projection Error of Camera Calibration Ball," *Acta Optica Sinica*, Vol. 32, No. 12, pp. 1215001, December 2012
2. D. Zhang, J. Duan, L. J. Gu, and W. B. Jing, "Method of Camera Calibration based on Image Processing," *Infrared and Laser Engineering*, Vol. 36, Supplement, pp. 293-297, June 2007
3. J. Bouguet, "Visual Methods for Three-Dimensional Modeling," California Institute of Technology, Pasadena, 1999
4. G. Bradski and A. Kaehler, "Learning OpenCV," O'Reilly Media, Inc., Sebastopol, 2008
5. P. H. Bu, H. Zhao, F. F. Gu, Y. Y. Ma, and H. H. Li, "A Fully Automatic Camera Calibration Method based on Planar Target," *Acta Optica Sinica*, Vol. 34, No. 10, pp. 1015004, October 2014
6. Y. Xu, P. Y. Guo, G. C. Long, X. H. Zhang, and Q. F. Yu, "Research on Fully Automatic Camera Calibration Method," *Acta Optica Sinica*, Vol. 33, No. 6, pp. 0612007, June 2013
7. T. N. Zhang, Y. Q. Tu, and D. C. Wang, "Novel Algorithm for Fast Detecting Checkerboard Corners," *Computer Engineering and Applications*, Vol. 44, No. 31, pp. 236-238, 2008
8. B. Zhao and J. Zhou, "Automatic Detection and Sorting of Corners by Improved Chessboard Pattern," *Optics and Precision Engineering*, Vol. 23, No. 1, pp. 237-244, January 2015
9. A. Abellard, M. Bouchouicha, and M. M. B. Khelifa, "A Genetic Algorithm Application to Stereo Calibration," in *Proceedings of the International Symposium on Computational Intelligence in Robotics and Automation*, pp. 285-290, Finland, 2005
10. R. B. Xia, W. J. Liu, J. B. Zhao, and J. T. Xu, "Fully Automatic Camera Calibration Method based on Circular Markers," *Chinese Journal of Scientific Instrument*, Vol. 30, No. 2, pp. 368-373, February 2009
11. L. Liang, D. F. Yin, and C. Wang, "Design and Detection Methods for Accurate Camera Calibration Targets," *Journal of Xi'an Jiaotong University*, Vol. 45, No. 4, pp. 82-85, April 2011
12. G. D. Liu, B. G. Liu, F. D. Chen, and T. Hu, "Study on the Method of the Accuracy Evaluation of Subpixel Location Operators," *Acta Optica Sinica*, Vol. 29, No. 12, pp. 3446-3451, December 2009
13. J. Zhou, L. T. Chen, Q. H. Liu, Y. M. Li, and Y. B. Rao, "Fast and Accurate RANSAC based on Optimal Sequential Probability Test and Local Optimization," *Chinese Journal of Scientific Instrument*, Vol. 33, No. 9, pp. 2037-2044, September 2012
14. C. Y. Zhang and W. Zou, "Design, Detection and Sorting Method for a Kind of Fisheye Camera's Calibration Board," *Computer Engineering and Applications*, Vol. 51, No. 15, pp. 188-192, 2015