Corner Detection Algorithm with Improved Harris

Li Wan(✉), Zhenming Yu, and Qiuhui Yang

School of Information and Electronic Engineering, Wuzhou University,
Wuzhou, Guangxi, China
{lee_wan,24032784,839296242}@qq.com

Abstract. Traditional algorithm of Harris needs to select a parameter for computing interest values of pixels, and its recognition ability for some types of corners is poor. To solve this problem, this paper proposes a corner detection method which is based on local standard deviation and logarithmic computing. The method decreases effecting to the response values of corners that near the candidate interested points through computing the logarithms of gradient, so it can detect different types of corner more effectively. It can redefine the interest value function according to the statistical features of the standard deviation to decide the corners. The function could avoid selecting value of parameters by person, and it could directly judge whether a candidate interested point is a corner, to make the algorithm has a higher objectivity. The experimental results show that the method can effectively detect the corners of various types, and it can achieve a more accurate effect of positioning.

Keywords: Corner detected · Harris algorithm · Standard deviation · Position

1 Introduction

Feature extraction is a basic procedure in image processing such as image stitching, target tracking or motion estimation. An extracted image feature can be the image's line, point or obvious region, etc., for example the region's edge, pathway, river, intersection of lines, and point of extremum curvature in the curve. Within the image's local area, exist features containing great information contents. Post-processing like image stitching can be performed by extracting these features. Different feature extraction operators need to be adopted for different types of image features. At present, feature extraction operators has three categories: Point feature extraction operators, region feature extraction operators and line feature extraction operators. Among these, region feature extraction operators are obtained mainly through region segmentation, where the precision of feature extraction depends on the accuracy of region segmentation; line feature extraction operators comprise mainly of gradient operators, second order difference operators, Gauss-Laplace operators, Hough Transform and other common linear extraction and descriptive operators; point feature extraction operators are based on different definitions to corner. A corner can be the point with drastic change of image gradation or the point of extremum curvature in the curve on the image's edge, such as the crossroad in the image, curve or

T. Tan et al. (Eds.): IGTA 2015, CCIS 525, pp. 260–271, 2015.
DOI: 10.1007/978-3-662-47791-5_30

intersection of lines, the central point of lakes, farmlands or other areas, the local extreme point in Wavelet Transform, and other points of significance that can be extracted by some similarity measure criterion. These points not only reserve rich important image information including the image's structural information contained within line or region features, but also their invariability with respect to the image's geometrical shape is easily accepted by the observer. Therefore, corners are usually used as a feature to extract.

Corner detection methods are mainly classified into two types: Edge-based methods and image gradation based methods. The edge-based methods extract and encode the image's edge, and extract corners by computing the edge curvature. Some methods perform immediate fitting of curves while others use some types of functions to perform segmented fitting of curves before computing the position of curvature extreme point according to these fitted segmented equations[1]. For example, some researchers use cubic polynomial, B-spline and other methods separately to fit curves[2-3], and detect the corners by computing the curvature of the points in the curve[4]. This category of methods largely depends on segmentation of image and edge extraction. When local changes occur to the detected object, operational errors or failure may result. Moreover, the result of corner detection is related to edge strength and the gradient of edge direction, hence it is rather sensitive to noise. The neighborhood grayscale based methods give the major consideration to the variation of grayscale within a neighborhood in extracting corners. In Harris Corner Detection Method[5-7], the structure of a 2*2 local area is determined under the Gaussian Window. The KLT[8] and MIC[9-10] Algorithms also calculate the local structure under a window. Independent of other local features of image but only using the features of the corner itself, the SUSAN algorithm[11-12] extracts corners by the criterion of the grayscale similarity between a central point of a round-shaped stencil and a neighborhood point. No Need to calculate the first order curvature, the algorithm runs fast; but the detection area of feature points is the range where grayscale similarities are centralized, without other information about other angles and directions of feature points, which brings about inconvenience to the follow-up image matching and other applications. The SIFT algorithm[13-15] proposed in 2004 seeks for the extreme point(s) in the scale space. Its extracted corners demonstrate sound invariability with respect to changes in translation, rotation, scale, brightness and noise, etc., as well as desirable stability with respect to partial cover on scene, viewpoint change, and affine transformation, etc. A large number of SIFT eigenvectors with rich information contents can be created inside a few objects. However, the process is rather complicated, and the computing time is long. Vast floating number operations are required for repeated use of convolution smoothness and weighted histogram statistics. Moreover, the feature points extracted by this algorithm cannot reflect the image's structural features.

Through the study and analysis into of the original Harris algorithm, this paper adopts the sifting method with the pixel points near the edge as the candidate points, save computation of points in vast smooth regions. By taking the logarithm of gradient value in extracting corners, the effect of the edge response of the edge points near the corners on corner's response value is reduced, so that complicated types of corners are detected with more efficacies, which can enhance the positioning accuracy of corners. Finally, in order to reduce the subjectivity that parameters are determined empirically in the response value function in the original algorithm, standard deviation is used to redefine the response value function of corners.

2 Harris Algorithm

The Harris Algorithm was put forward by Chris Harris in 1988. Based on the Moravec algorithm of corner extraction and enlightened by the self-correlation function in signal processing, it uses boundary curvature detection as a substitute for first order partial derivative in describing grayscale gradation.

For input images, the Harris Algorithm firstly calculates the difference value of each point in the grayscale image along the horizontal and vertical directions, denoted as Ix and Iy, respectively. Next, the Gaussian Function is used to apply convolution to Ix2, Iy2 and IxIy to get a self-correlation matrix M. In the expressions in this paper, the symbol "$\otimes$" denotes convolution, and W is Gaussian Window.

$$I_x = \frac{\partial I}{\partial x}, \quad I_y = \frac{\partial I}{\partial y}, \quad A = I_x^2 \otimes W, \quad B = I_y^2 \otimes W, \quad C = I_x I_y \otimes W, \quad M = \begin{bmatrix} A\,C \\ C\,B \end{bmatrix} \tag{1}$$

M is a real symmetrical matrix, so there exist two eigenvalues. Both eigenvalues reflect the characteristic of image pixels. The cases of the eigenvalues can offer to discriminate among flat region, edge or corner. The value of both eigenvalues falls into the following three cases:

(1) When the matrix's two eigenvalues are both larger positive values, and the grayscale gradation is larger in any direction, with the local self-correlation function centered on pixel point (x, y) as the peak, then this pixel point is considered as a corner;

(2) When an eigenvalue is large and the other is small, the local self-correlation function renders a shape of ridge, and along the direction of the ridge the grayscale gradation is smaller than along the direction perpendicular to the ridge, then the pixel point is located in the edge line;

(3) When both eigenvalues are small, and the local self-correlation function is smooth, then the pixel point is located within a smooth region.

Thus, the response value of Harris corners can be determined by the determinant and trace of the matrix M, as shown in Formula (2).

$$\begin{aligned} C(\mathrm{x}, y) &= Det(M) - \mathrm{k} \times Tr^2(M) \\ &= (AB - C^2) - k(A+B)^2 \end{aligned} \tag{2}$$

$$Det(M) = \alpha\beta = (AB) - C^2, \quad Tr(M) = \alpha + \beta = A + B .$$

where k is an empirical value ranging between 0.04 and 0.07. A point of extremum response value within a local range is considered as a corner. Since the number of extreme points are generally large, all extremum points can be sorted so that the optimum point is selected according to the requirement for the number within the neighborhood.

The Harris Algorithm for corner extraction is simple in computation, only using first order difference and smoothing without massive calculation as in the SIFT Algorithm. Harris corners contain rich local information about image, making up for the deficiency of the SUSAN Algorithm that it lacks information about other angles and directions of feature points. The extracted corners are uniformly and rationally distributed in the image. For regions with rich textural information, Harris operators can extract large amounts of useful feature points; while for regions with less textural information, the extracted feature points are fewer. This algorithm is with good stability, and insensitive to image rotation, grayscale gradation, noise effect and other changes and viewpoint transform.

However, the Harris method of corner detection is out of invariability with respect to changes in scale. Pixel points which are detected as corners in a larger-scale image may become edge points or non-corner points under a smaller scale. In recent years, domestic and foreign researchers have been mainly devoted to modifying the Harris Algorithm of corner detection into a feature point detection algorithm with scale invariability. Zhan Wang et al. [16] have also accomplished the scale invariability by using the method of Wavelet Transform. By studying the CSS corner detection operators, Literature [17] implemented a feature point extraction algorithm with stronger noise immunity, in combination with the curvature scale space based Harris corner detection operators. Xianfeng Xu et al. [18] have singled out the feature points with scale invariability which can best represent the local structure by performing tracking and grouping while detecting the multiscale Harris feature points.

However, it has an inferior detection effect for T-, Y- or X-shape corners with complicated structure(as shown in Fig. 1). When there are large amounts of corners with complicated structure in an image, the Harris method of corner detection may miss a lot of corners, which may result in a reduction of accuracy in the follow-up image analysis.

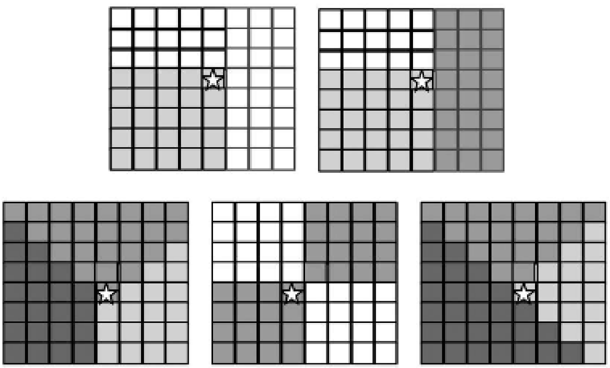


Fig. 1. Corners of various shapes

3 The Improved Algorithm

From the process of the Harris Algorithm in the foregoing text, it can be seen that all pixel points need to be calculated. But the corners are located near the edge, so this paper begins by sifting the image, eliminating the pixel points within the smooth region, and only saving the points near the edge as candidate points, which can reduce the calculated quantity of the algorithm. In other words, the algorithm can achieve the goal of extracting corners only by accurately distinguishing the corners from the edge points among the candidate points. When the candidate points are located near the corners, since the window W contains a vast number of edge points whose gradient changes rapidly along a same direction, namely one of the eigenvalues α and β is larger than the other, the response value derived from these edge points may be approximate to the response value of the corners, which mixes up the algorithm's discrimination on the corners by misjudging these points near the corners as the corners so that the algorithm reduces the accuracy in corner positioning. This paper makes a modification over this point by adopting the technique of taking the logarithm of the gradient of the candidate points. And it reduces the effect of edge response on computation of the response value of corners and enhances the positioning accuracy of corners by inhibiting one of the eigenvalues of the candidate points near the edge from getting too large.

$$G(x,y)=\sqrt{I_x^2+I_y^2}$$
$$G_\theta(x,y)=\arctan\left(\frac{I_x}{I_y}\right). \tag{3}$$

Formula (3) demonstrates the computation of the gradient value of the candidate points.

$$\log(G'(x,y))=b*\log(G(x,y)) \tag{4}$$

Learn from Formula (4) that the computation only changes the magnitude, not the direction of gradient. In order to reach the end G'(x, y)<G(x, y), the range of the parameter b is 0<b<1.

According to Formula (2) and the evaluation formula of gradient, we get:

$$\begin{aligned} \mathrm{I}_x' &= G'(x,y)*\cos(G_\theta'(x,y)) \\ &= G'(x,y)*\cos(G_\theta(x,y)) \\ &= ((\sqrt{I_x^2+I_y^2})^b*\frac{I_x}{\sqrt{I_x^2+I_y^2}} \end{aligned} \tag{5}$$

$$\mathrm{I}_y^{'} = (\sqrt{I_x^2 + I_y^2})^b * \frac{I_y}{\sqrt{I_x^2 + I_y^2}} \quad (6)$$

The improved algorithm replaces Ix and Iy in Formula (1) of the Harris Algorithm with I'x and I'y, respectively. Thus, the effect of the response value of the edge points near the corners on judgment of corners can be inhibited.

In addition, in Formula (2), the parameter k needs to be determined in order to calculate the response value C(x, y), which varies, depending on the variations of grayscale gradation between different images. And then the empirical value is selected to distinguish the corners from non-corners among the candidate points. This parameter is of subjectivity. It can make a difference to the detection effect of corners and reduce the accuracy of detection. This paper replaces the determination of the parameter k with the calculation of standard deviation, and redefines response value function, so that the algorithm can automatically screen out corners.

$$\begin{aligned}
&R(x,y) = \frac{Det(M^{'}(x,y)) - d_1}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(Det(M^{'}(x,y)) - d_1)^2}} - \frac{Tr^2(M^{'}(x,y)) - d_2}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(Tr^2(M^{'}(x,y)) - d_2)^2}} \\
&M^{'} = \begin{bmatrix} (I_x^{'})^2 \otimes W & I_x^{'} I_y^{'} \otimes W \\ I_x^{'} I_y^{'} \otimes W & (I_y^{'})^2 \otimes W \end{bmatrix} \\
&d_1 = \frac{1}{N}\sum_{i=1}^{N} Det(M^{'}(x,y)) \\
&d_2 = \frac{1}{N}\sum_{i=1}^{N} Tr^2(M^{'}(x,y))
\end{aligned} \quad (7)$$

Where i = 1, 2, 3, ..., N is the total number of candidate points. d1 and d2 denote the determinant of the matrix M' and the mean of the squares of its traces. The two components in the response value function are the determinant and the standard deviation of the squares of the traces.

Both eigenvalues at the corner are larger, while only one of them is larger on the edge, and the edge points outnumber the corners considerably among the candidate points, therefore,

(1) When R(x, y)>0, the corner response is larger than the edge response, and the candidate point is identified as a corner.

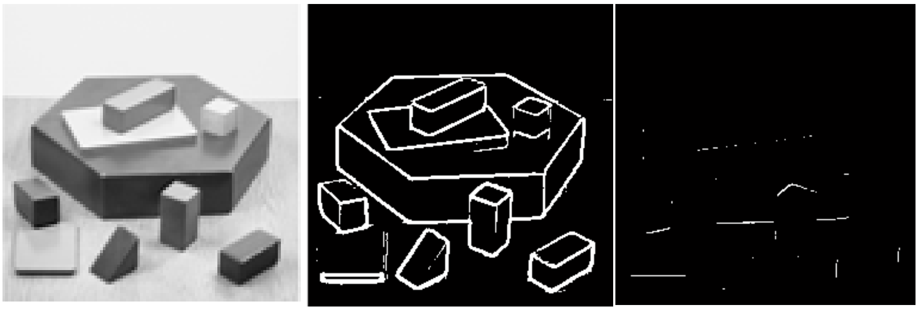
(2) When R(x, y)<0, the edge response is larger than the corner response, and the candidate point is identified as an edge point.

In computing the response value of a candidate point (x, y), if R(x, y)>0 and presents a local maximum, then this point is identified as a corner which is distinguished immediately from edge points; conversely, if R(x, y)<0, then it is identified as an edge point and filtered out. A finally obtained set of points is a set of the extracted corners. A modest number of corners can be selected as required.

4 Implementation and Performance Analysis of the Algorithm

Base on the analysis of the original Harris Algorithm, the modification has been presented for the algorithm in the above section. The following is a description of the implementation procedure of the algorithm.

Firstly, find the image's gradient value and Ix, Iy in the Harris Algorithm in order to distinguish the pixel points between the smooth region and the marginal region, namely to screen out the candidate points. The image's grayscale is unequally distributed, therefore there may be different effects when gradient value is directly used to distinguish the points near the edge from those within the smooth region. If the image's grayscale is equally distributed, then the gradient value can well distinguish the edge from the smooth region. But if the grayscale in the image is largely correlative so that grayscale gradates rapidly in most parts but slowly in another small part, then the edge information in this small part may be eliminated for being mistaken for a smooth region when gradient value is used alone to distinguish the edge from the smooth region. As mentioned in the above section, the effect of too large a gradient value may be reduced by taking the logarithm of the gradient. After taking the logarithm of the image's grayscale, therefore, this paper applies binaryzation to the derived image to get an image with only the points near the edge, as shown in Fig. 2:



(a)original image (b)gradient logarithmic transformation and binaryzation (c) gradient binaryzation

Fig. 2. Detected interested points

The value of the parameter b needs to be determined in taking the logarithm of gradient. Firstly, 0<b<1. According to the mathematical nature of logarithm, as the value of b becomes smaller, it affects a larger value more significantly than a smaller value so that the former decreases drastically and that the difference diminishes between the larger value and the smaller value after transform. Conversely, as b becomes larger, a larger value decreases less substantially, with a smaller effect on a smaller value, so that the difference remains larger between the larger value and the smaller value after transform. Since the value of b is dependent on the image's grayscale distribution, assume that,

$$b = \frac{n_1}{n_2}. \tag{8}$$

Where n1 and n2 represent the number of pixel points whose gradient value is above the average gradient value and the number of pixel points whose gradient value is below the average gradient value, respectively.

According to the candidate points obtained in the first step and the matrix M' calculated through Formulas (1), (5) and (6). In the calculating process, use the binarized image to enhance I'x and I'y of which the logarithm has been taken, so as to enhance the responses of edge and corners. Then determine the matrix M' according to I'x and I'y.

Calculating the response value function. As mentioned in the above section, screen out the candidate points and extract the corners according to the value of the response value function.

Generally repeatability is applied to test the performance of feature extraction algorithm. The computational formula of repeatability is as follows:

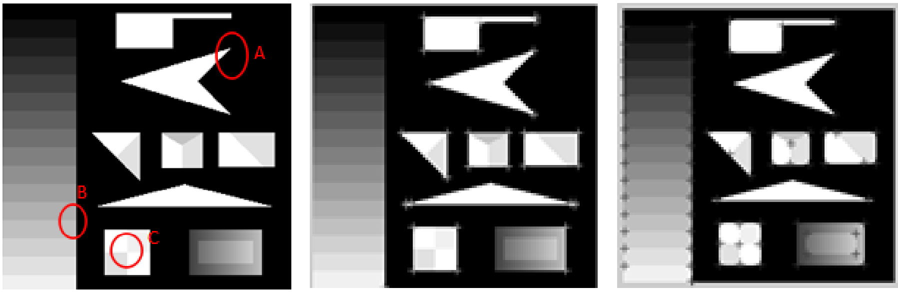
$$r = \frac{2n}{n_1 + n_2} \tag{9}$$

where n is the number of feature points detected in both images, n1 represents the number of feature points detected in the original image while n2 represents the number of feature points detected in the image already changed. Since the change in scale may change the size of feature points, substantial change in scale may results in oversize or undersize change of feature points, lead to an absence of corresponding feature points in the image after transform. Otherwise, if the original image keeps unchanged in size during a rotation transform, then a portion of information in the image may get lost, while no feature points will be found in correspondence to the feature points in this portion of information in the original image. Therefore, these feature points without corresponding points for a loss of information as a result of the transform shall be precluded in computing n1 and n2.

In order to decide whether the feature points are correspondent ones, according to model and parameters of transform, transforming the set of extracted feature points under the same coordinate system. Regard the points between which the coordinate difference is within a certain range as the corresponding feature points. In the following experiment, assume the points whose differences between x-coordinate and y-coordinate are all within 3 pixels as the corresponding feature points.

5 Experimental Result

The experiment was conducted in a microcomputer with 3.00GHz Pentium(R) Dual-Core CPU, 2GB DDR3 RAM and an OS of Windows XP SP3. The images were processed with MATLAB7.10.



(a) original image (b) result of Harris (c) result of improved Harris

Fig. 3. Image 1

(a) improved Harris of A (b) Harris of A (c) improved Harris of B (d) Harris of B (e) improved Harris of C (f) Harris of C

Fig. 4. The details of image 1

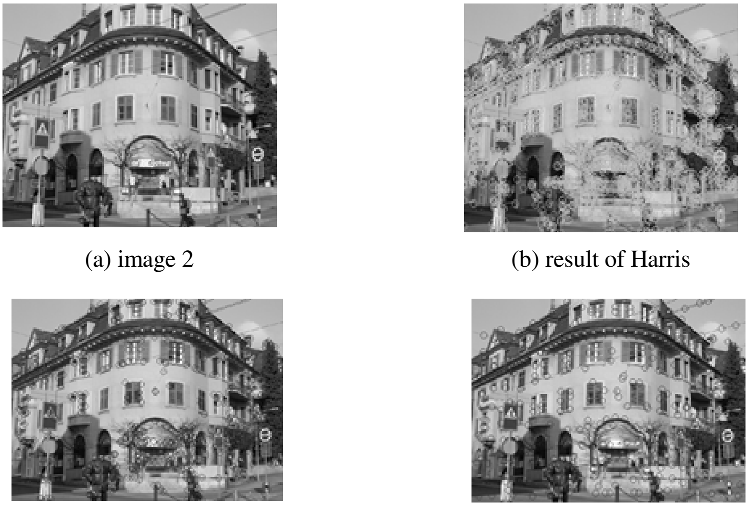
Fig. 4 is the image after the circled part in Image 1 is magnified. The above experiment reveals the improved algorithm performs better in detecting higher-order points, such as T-, X- and Y-shape corners. The difference can be observed between the positions of corners detected by both algorithms. The improved Harris Algorithm can locate corners more accurately, and detect some corners which the Harris Algorithm fails to detected effectively, as shown in Fig. 4(d) and (f).

For the corner extraction effect in Image 1 of Fig. 3, a comparison between both algorithms is shown as in Table 1, easy to see that the improved algorithm can extract a sufficient number of corners with a smaller number of omitted or erroneous corners, and can extract various types of corners more accurately.

Table 1. Comparison of two algorithm

Image	Algorithm	Number of corners	Number of detected corners	Number of erroneous corners	Number of correct corners	Number of omitted corners
image1	Harris	78	33	0	33	45
	improved Harris		76	0	76	2

The experiment evinces that the algorithm has a good effect for corner extraction of real scene images. As shown in Fig. 5, the corners detected by the circular marks in the figure can procure a more sufficient number of equally distributed corners, compared with the Harris Algorithm and the SUSAN Algorithm. Moreover, the stability of the Harris Algorithm is retained with respect to rotation, gradation and noise effect, while the positioning of corners is even approximate to the corner positions of real objects.

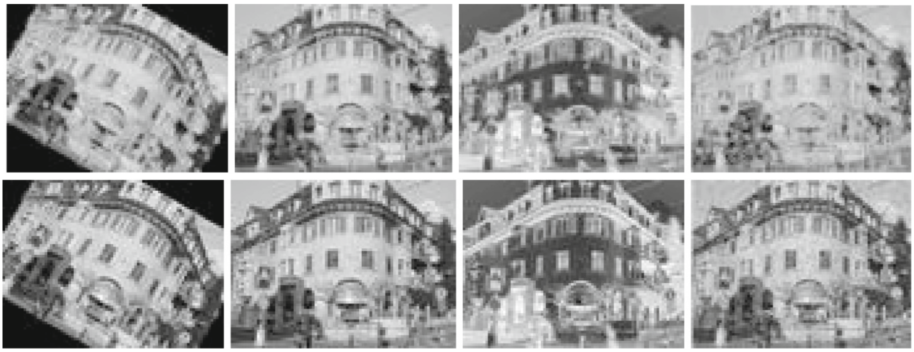


(a) image 2 (b) result of Harris

(c) result of improved Harris (d) result of SUSAN

Fig. 5. Comparison of results of three algorithms

To verify the effect of corner extraction algorithm more objectively, calculate the respective repeatability of the above three original images as well as the corner extraction algorithm. The transformed images of the original images and the extraction effect after transform are shown in Fig. 5. And computed their repeatability separately, with the result showing in Table 2. The transforms over the images are rotation by 30 degrees, zooming by a factor of 0.9, and finding the reversal of grayscale (namely subtracting the grayscale from 255). In addition, a salt-and-pepper noise is added to the images with a coefficient of 0.01.



(a)rotation image (b)zoom image (c)gray inversion (d)image with noise

Fig. 6. Comparison of two algorithms

In Fig. 6, the images in the first row are the corner extraction after transform over the original images by the Harris Algorithm; the images in the second row are the corner extraction after transform over the original images by the improved Harris Algorithm. By comparison, the improved algorithm retains the stability of the original algorithm with respect to rotation, grayscale gradation and noise. Also, the improved algorithm demonstrates better resistance to the salt-and-pepper noise in the images. Moreover, the improved algorithm performs more accurately in corner extraction. On the premise of ensuring sufficient number, it avoids a large number of redundant corners in the Harris Algorithm, reduces calculating workload in the follow-up image processing and enhances accuracy.

Table 2. Comparison of repeatability of the image 2

Algorithm	Item	Original image	Rotation image	Zooming image	Gray revesal image	Noise image
Harris	feature points	2669	2807	2109	2666	4584
	repeatability		44.1%	44.2%	53.4%	68.3%
improved Harris	feature points	72	562	384	466	541
	repeatability		67.6%	66.3%	84.9%	76.7%

The data in Table 2 reveals that the presented improved algorithm has higher repeatability of feature points than the Harris Algorithm under the condition of rotation, zooming, grayscale gradation and noise transform, it achieves an overall better effect.

6 Conclusion

This paper offers an improved Harris algorithm for corner extraction, which can locate the position of corners more accurately and which has effectively improved the problem with the traditional Harris Algorithm that the complicated T, X, Y and other types of corners cannot be well detected. Meanwhile by means of the statistical property of data, the response value function of the algorithm is redefined to remove the empirical evaluation of parameters in response value computation in the traditional algorithm. The algorithm's subjective effect is reduced to implement corner extraction more stably and objectively and achieve a better experimental result than the original algorithm. The algorithm in this paper is easy to implement with sound effectiveness. It not only demonstrates higher accuracy in corner positioning, but also keeps the desirable stability of the algorithm with respect to rotation, grayscale gradation, scale and noise. Under the same transform condition, the improved algorithm has higher repeatability than the Harris Algorithm. However, the accuracy of corner positioning remains at the pixel-level, namely the coordinates in corner positioning remain the coordinates of integral pixels maintained at the same accuracy as image pixels. Moreover, the algorithm fails to be concerned with scale invariability. How to achieve better positioning accuracy and extract more stable and accurate corners is what the author shall study and resolve in the next step.

Acknowlegments. This paper is financially supported by National Natural Science Foundation of China (Grant No. 61062014) and Guangxi University of Science and Technology Research Program (Grant No.KY2015YB290).

References

1. Kitchen, L.: A Rosenfeld.: Gray Level Corner Detection. J Pattern Recognition Letters 3(1), 95–102 (1982)
2. Langridge, D.J.: Curve Encoding and the Detection of Discontinuities. J. CVGIP 20(1), 58–71 (1982)
3. Mediono, G., Yasumoto, Y.: Corner Detection and Curve Representation Using Cubic B-splines. J. Computer Vision Graphics and Image Processing 39(3), 267–278 (1987)
4. Zhong, B., Liao, W.: Corner Detection Based on Accumulative Chord Length of Refined Digital Curves. J. Journal of Computer Aided-Design and Computer Graphics 16(7), 939–943 (2004). (in Chinese)
5. Harris, C.G., Stephens, M.J., Kesselman, C.: A combined Corner and Edge Detector. In: Proceedings Fourth Alvey Vision Conference, Manchester (1988)
6. Mikolajczyk, K., Schmid, C.: Scale and Affine Invariant Interest Point Detectors. J. InternSational Journal of Computer Vision 60(1), 63–86 (2004)
7. Cheng, B., Tang, X.: Study of Harris scale invariant keypoint detector. J. Journal of Zhejiang University. 43(5), 855–859 (2009). (in Chinese)
8. Husbands, P., Mill, F., Kesselman, C.: Simulated Co-evolution as the Mechanism for Emergent Planning and Scheduling. In: Belew, R.K., Booker, L.B. (eds.) Proceedings of the 4th International Conference on Genetic Algorithms, pp. 264–270 (1991)
9. Hillis, D.W.: Co-evolving Parasites Improve Simulated Evolution as an Optimization Procedure. J. Physica D: Nonlinear Phenomena 42, 228–234 (1990)
10. Trajkovic, M., Hedley, M.: Fast corner detection. Image and Vision Computing 16(2), 75–87 (1998)
11. Smith, S., Brady, M.: A new approach to low level image processing. J. International Journal of Computer Vision 23(1), 45–789 (1997)
12. Zhang, Q., Liu, Z., Pang, Y., et al.: Automatic registration of aerophotos based on SUSAN operator. J. Acta Geodaeticact Cartographica Sinica 32(3), 245–250 (2003). (in Chinese)
13. Distinctive, D.L.: Image Features from Scale-Invariant Keypoints. J. International Journal of Computer Vision 60(2), 91–110 (2004)
14. Wu, Y.: An image matching method based on SIFT and SUSAN features. J. Image Processing and Multimedia Technology 30(12), 33–39 (2011). (in Chinese)
15. Zou, B., Ruan, P., Xiang, Y., et al.: An automatic panoramic images mosaic algorithm with precise matching. J. Computer Engineering and Science 32(8), 60–63 (2010). (in Chinese)
16. Wang, Z., Huangfu, K., Wan, J., et al.: Multi-scale wavelet based two dimensional cornet detection. J. Journal of National University of Defense Technology 21(2), 46–49 (1999). (in Chinese)
17. Li, Z., Shen, Y., Kesselman, C: A robust corner detector based on curvature scale space and harris. In: Image Analysis and Signal Processing, pp. 223–226. Hubei (2011)
18. Xu, X., Tan, J.: An improved Multi-scale Harris deature point detection mothod. J. Jisuanji Gongcheng/ Computer Engineering 38, 174–177 (2012). (in Chinese)