

Study on Augmented Reality Method for Unlabeled Recognition Based on Filtering Processing Optimization

Sihao Chen

College of Information Science, Beijing Language and Culture University, Beijing 100083, China
xk18csh@126.com

Abstract. Augmented Reality (AR), as a technology that superimposes virtual objects and real scenes, expands the information amount of actual scenes through virtual information. In view of the shortcomings of the previous AR application methods, this paper proposes an AR application implementation method of unlabeled recognition based on filtering processing optimization. This algorithm uses filtering to preprocess and optimize the image, and solves the problem of poor matching effect of the previous recognition algorithm. The 3D model is established in the two-dimensional image and the model is rendered on the two-dimensional plane. The experimental results show that compared with the previous matching algorithm, the recognition efficiency of this method is improved by more than 10%, which has more obvious advantages in matching efficiency and better robustness.

1. Introduction

As a new type of man-machine interface and simulation tool, the application of augmented reality [1] has received more and more attention, and has played an important role, showing great potential. AR technology is a new technology that integrates real world information and virtual world information seamlessly. It superimposes virtual information on the real world after computer simulation, which is hard to experience in a certain time and space in the real world (visual information, sound, taste, touch, etc.). Then it is perceived by human senses, so as to achieve sensory experience beyond reality. AR gives full play to its creativity and provides a powerful means for the expansion of human intelligence, which has exerted a huge and far-reaching impact on the production mode and social life of human beings.

In general, spatial feature-based algorithms are used for visual enhancement. In the past, most of the applications of spatial feature-based algorithms were to achieve AR by matching marked targets. In addition, SIFT operator [2] or SURF operator [3] were also used for unmarked target matching. For marked target matching, the application suitability is reduced due to the limitation of the target. However, the matching of unmarked targets combined with SIFT operator can be performed for unmarked targets, but it is difficult to achieve good results due to the influence of the matching algorithm efficiency. SURF not only maintains the good performance of SIFT, but also gives consideration to better efficiency. However, it is still difficult to achieve satisfactory results in the treatment of video stream. Aiming at the shortcomings of the SURF algorithm in rate, this paper puts forward the optimized the ORB (oriented FAST and rotated BRIEF) [4] characteristic operator to achieve AR application. On the premise of guaranteeing SURF's recognition rate, the matching rate is improved.

However, the conventional ORB characteristic operator [5] not only improves the rate, but also increases the possibility of poor image matching effect with fewer feature points. In order to improve the robustness of ORB algorithm, image filtering processing is added in this paper.

In addition, how to build 3D model in AR is very important work. This paper uses 3D registration



technology [6], which can combine the detected objects to obtain the relationship between the camera and the real situation scene, and determine the relative position relationship between the camera and the object to be superimposed, so as to realize the "seamless" superposition of virtual objects and real images.

2. Image edge enhancement

Considering the rule of image feature point extraction and the characteristics of the method, this paper extracts feature points by enhancing the edge significance of the image [7]. This is a combination of optics and how the human eye perceives things. The relative brightness between objects and objects plays a very important role. Therefore, the image with strong contrast is more convenient for human eyes to observe, and it is also conducive to image feature extraction, and image edge enhancement is a good method. One of the effective ways to enhance the image edge is to stretch the pixel points around the edge of the image reasonably. The object of this paper is RGB color image. In order to reduce the computational complexity, the color image is converted into grayscale image for processing and finally displayed with color image.

2.1 Image edge pixel extraction

According to the above ideas, this section first detects and extracts the edge pixels. In view of the timeliness of pretreatment, the existing image processing method of Laplacian [8] edge detection operator is improved. According to the analysis of a large number of experimental results, the algorithm is optimized and improved for this project. In order to ensure the timeliness and accuracy of the algorithm, the following templates are determined:

$$\begin{pmatrix} -1 & -2 & -1 \\ -2 & 12 & -2 \\ -1 & -2 & -1 \end{pmatrix}$$

Calculate according to each pixel of the image:

$$\begin{aligned} \nabla f^2(i, j) = & 12f(i, j) - 2f(i+1, j) - 2f(i-1, j) - \\ & 2f(i, j+1) - 2f(i, j-1) - f(i-1, j-1) - \\ & f(i-1, j+1) - f(i+1, j-1) - f(i+1, j+1) \end{aligned} \quad (1)$$

When $\nabla f^2(i, j) \geq 0$, this edge can be identified as a positive edge. When $\nabla f^2(i, j) \leq 0$, the edge can be considered negative. When $\nabla f^2(i, j) = 0$, the edge can be considered to be zero. These zero-edge pixels can be ignored and not enhanced later.

2.2 Image edge pixel enhancement

After the detected edge pixels are obtained, corresponding enhancement processing can be carried out.

There are three cases as follows:

$$g(i, j) = \begin{cases} f(i, j) + e, & \nabla f^2(i, j) > 0 \\ f(i, j) - e, & \nabla f^2(i, j) < 0 \\ f(i, j), & \nabla f^2(i, j) = 0 \end{cases} \quad (2)$$

In the formula: $e \leq k \leq \text{grad}(i, j)$, k is an adjustable constant; $\text{grad}(i, j)$ refers to the gradient on the grayscale of the corresponding pixel, so the edge enhancement result can be expressed as:

$$g(i, j) = f(i, j) + k \times \text{grad}(i, j) \quad (3)$$

The edge enhancement amplitude can be controlled by adjusting the constant k . To simplify the process, k can be set to 1.

2.3 Combination of Gaussian filtering to achieve the enhancement effect

Through the observation of the test results, it is found that if the edge detection algorithm is directly carried

out, the effect is not satisfactory. Therefore, Gaussian filtering [9] is first used for pretreatment, and then the edge detection algorithm mentioned above is conducted. This can not only avoid noise interference, but also highlight key edge information. The final effect is shown in figure 1.



(a) original image (b) edge enhancement

Figure 1 Comparison of renderings

3. Unmarked target matching

In order to implement the augmented reality enhancement superposition of the model, it is necessary to know the target position that needs to be superimposed. In this case, non-specified target detection and matching, i.e. unmarked target matching, is adopted to determine the position of model superposition by matching results. Therefore, the realization of augmented reality also has higher requirements for the unmarked target matching of two-dimensional objects.

The research on local invariance has become a hot topic at present. The traditional SIFT detection operator performs very well in precision. The Later SURF detection operators improve the short board in the timeliness of SIFT detection operators, but the effect is not satisfactory. The matching algorithm is optimized based on traditional ORB detection operators. Traditional ORB detection operators are based on FAST feature descriptors [10] and BRIEF feature descriptors [11], which greatly improve the timeliness. After image edge enhancement, the matching effect of ORB detection operator can be greatly improved. The implementation of traditional ORB detection operator is improved to make the algorithm more suitable for the application scenario of this project.

3.1 Feature extraction

Feature point detection adopts a detection method similar to FAST operator, but traditional FAST operator does not have rotation invariance, so each feature point also needs to add a direction information.

The core idea of traditional FAST feature point detection is to find pixel points different from other pixels. Usually, the gray value of the pixel point is compared with the gray value of 16 pixels around the pixel, as shown in figure 2. The threshold value k can be set, when the gray value of n consecutive pixels in 16 points is larger than the sum of the central point gray value and k , or less than the difference between the central point gray value and k . The size of n is also important here. Experimentally tested, the effect is the best when n is 9. Therefore, the traditional FAST feature point detection method is not adopted here to reduce the extraction of feature points, thus simplifying the algorithm complexity and improving the running efficiency of the program.

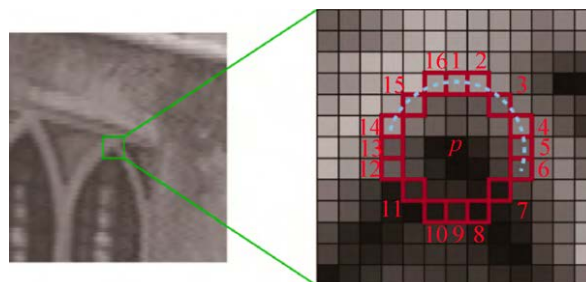


Figure. 2 Traditional FAST corner detection

Traditional FAST has the problem of insensitive edge response and lack of scale invariance. For this reason, this paper sets the number threshold of the search points higher than the expected value, determines the final result by Harris corner detection [12] sorting, constructs the image scale pyramid, and realizes scale invariance.

In terms of rotation invariance, the grayscale centroid method [13] is adopted in this paper, and the deviation between the grayscale and the centroid of the corner is taken as the direction of the vector.

The neighbour moment is defined as the traditional method:

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y) \quad (4)$$

Where: x and y should be guaranteed within the circular range of the neighbor radius r.

The centroid is:

$$C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (5)$$

The direction of feature point is:

$$\theta = \arctan(m_{01}, m_{10}) \quad (6)$$

3.2 Feature description

After the feature point is detected, the feature description with rotation invariance after optimization is used. It is the main idea to express image neighbourhood through relatively small intensity contrast. In order to reduce the length of the description feature and reduce the time and resources required to generate the description feature, the corresponding construction method of the original BRIEF descriptor is also modified.

Neighbourhood criterion τ of image P whose pixel size is $S * S$ is defined as:

$$\tau(p; x, y) = \begin{cases} 1, & p(x) < p(y) \\ 0, & \text{other} \end{cases} \quad (7)$$

Where: $p(x)$ is the pixel grayscale value at $x = (u, v)^T$ after the neighborhood p is smoothed, so that an n-bit binary string can be obtained:

$$f_n(p) = \sum_{1 \leq i \leq n} 2^{i-1} \tau(p; x_i, y_i) \quad (8)$$

Where, the coordinates of x and y are the Gaussian distribution centred on feature points, and n is equal to 128, 256 and 512. It has been tested that when n is 256, the effect is optimal and there are few errors.

Although the previous pre-treatment has solved the interference problem caused by image noise well, in order to achieve better results, when conduction comparison operation, not only a point is compared, but the 7x7 sub-window in the 31x31 pixel field is used. It obeys Gaussian distribution and uses integral image to perform efficient operation.

In addition, for the algorithm to have rotation invariance, the neighbourhood of the feature point needs to be rotated at a certain Angle, which can be set as the characteristic direction angle θ obtained above. However, the overall rotation efficiency is too low. In general, the matching points x_i and y_i in the previous neighbourhood are rotated.

Suppose n test point pairs are (x_i, y_i) and the matrix of $2 \times n$ is defined :

$$S = \begin{pmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \end{pmatrix}$$

Assuming that the rotation matrix formed by Angle θ is R_θ , the coordinates of the matching point after rotation can be obtained as:

$$S_\theta = R_\theta S \quad (9)$$

So the descriptor is obtained as:

$$g_n(p, \theta) = f_n(p) | (x_i, y_i) \in S_\theta \quad (10)$$

After the descriptor is determined, 256 pixel blocks that meet the requirements are searched in the pixel block pair using greedy search to ensure that they have the lowest correlation.

3.3 Optimization of feature matching

After predicting the target image, the feature extracted above can be used for matching recognition. First a threshold is set to eliminate the few feature points and reduce unnecessary matching. The point sets that appear in the same location can be merged.

Here, the matching feature points can be transformed to form two groups of one-dimensional point sets. W and h represent the width and height of the image respectively, and the transformation relation is as follows:

$$p_i(x, y) = x + y(w - 1) \quad (11)$$

Remove unnecessary points by comparing p point sets:

$$p_i = \begin{cases} (p_i + p_{i+1}) / 2 & |p_i - p_{i+1}| < k \\ p_i & \text{other} \end{cases} \quad (12)$$

When the distance between two points is less than the threshold value k , the two points in the concentration of P are merged and replaced by the midpoint of two points. After the new point set is obtained, the original two-dimensional point set is restored:

$$f_i(x, y) = \begin{cases} x = \text{mod}(p_i, w) \\ y = \lfloor p_i / w \rfloor \end{cases} \quad (13)$$

and then transfer the new points into:

$$Q_i(x, y) = x(h - 1) + y \quad (14)$$

The same operation is performed to remove points:

$$Q_i = \begin{cases} (Q_i + Q_{i+1}) / 2 & |Q_i - Q_{i+1}| < k \\ Q_i & \text{other} \end{cases} \quad (15)$$

The final set of feature points is obtained by combining the results of removal:

$$f_i(x, y) = \begin{cases} x = \lfloor p_i / h \rfloor \\ y = \text{mod}(Q_i, h) \end{cases} \quad (16)$$

Then, feature point matching can be carried out, and point matching process can be processed in parallel through OpenMP to improve resource utilization. The final effect is shown in figure 3.

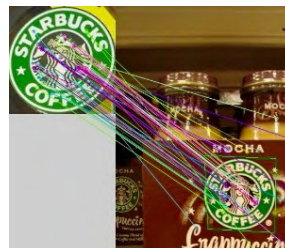


Figure 3 Target matching

4. Model construction and rendering

After determining the location of the target image through detection, the model can be built and the virtual information can be rendered to realize AR. The rendering of virtual two-dimensional images, mainly involving two-dimensional affine transformation [14], is relatively easy to achieve. However, to realize the rendering of 3D model, more tedious work is needed. Combined with the existing method of building 3D model and the demand analysis of this application scenario, the model can be rendered by the following methods.

4.1 Three-dimensional model construction

To add a 3D model perfectly on a two-dimensional plane, it is necessary to know the target's attitude or relative position about the camera. The attitude is represented by using the Euclidean space combination transform [15] in the rectangular coordinate system.

The three-dimensional coordinate position corresponding to the target position matched above has the following relations:

$$P = A * [R | t] * M \quad (17)$$

Where: P represents points in two dimension, which is projected from three dimensions. A represents the internal parameter matrix of the camera; $[R | t]$ indicates that the European transformation corresponds to a 3 x 4 matrix. M stands for points in three dimension.

Since P has been detected before, we mainly focus on how to obtain the internal parameter matrix A of the camera, set the parameter M, and determine the transformation matrix $[R | t]$.

4.2 Camera calibration

The main purpose of camera calibration is to determine the internal parameter matrix of the camera. In this paper, the checkerboard calibration method, which is more popular now, is selected, as shown in figure 4. This method mainly takes the checkerboard images of different samples. As the checkerboard is a very regular image, the camera's internal parameter matrix can be solved by detecting the checkerboard corners and simulating the 3D coordinates. Because this method is already a very mature method, it is not described in detail here.

4.3 Attitude estimation

In three-dimensional space, the precise position coordinates of the corners can be used to estimate the transformation relationship between the camera and the markers by marking, which can be called attitude estimation from two-dimension to three-dimension. This process can be understood as Euclidean space transformation between 2D image and camera.

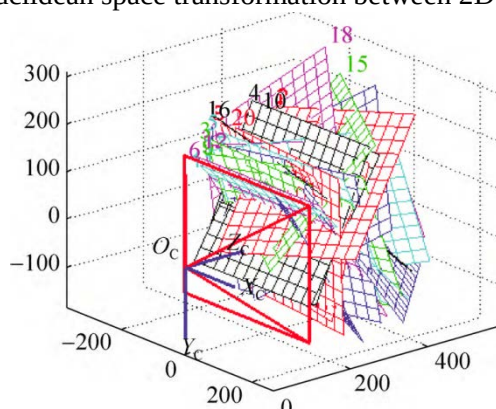


Figure 4 Camera calibration

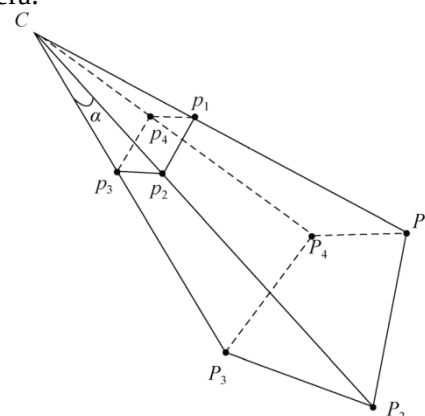


Figure 5 Sketch map of attitude estimation

In figure 5, C refers to the position of the center of the actual camera, P_1, P_2, P_3, P_4 are the corners of the target detected in real space, p_1, p_2, p_3 and p_4 are the corners projected by the actual target on the two-dimensional plane of the camera image (the corners detected above).

Here P_1, P_2, P_3, P_4 are all determined, and the attitude estimation is to find the relationship from P_1 ,

P_2, P_3, P_4 to P_1, P_2, P_3, P_4 :

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (18)$$

Where, (X, Y, Z) is the three-dimensional point in the world coordinate system; (u, v) is the two-dimensional point in the coordinate system of imaging surface. (c_x, c_y) is the main optical axis point, which is generally the center of the image. f_x and f_y are focal lengths.

4.4 The solution of the transformation matrix rendering three-dimensional model

After the two-dimensional point set is obtained, the information of three-dimensional point set and the internal parameter matrix of the camera can be obtained, as shown in figure 6. We only need to calculate the transformation matrix $[R | t]$ to complete the three-dimensional model rendering. There are many methods and tools for calculating the transformation model. The solvePnP function in OpenCV is used to calculate the transformation matrix $[R | t]$. After obtaining rotation and translation matrix, 3D model rendering can be carried out on the matching target. Using the camera's internal parameters, the European transformation matrix and the matching corner coordinates, and combining with OpenGL, the rendering of the three-dimensional model can be carried out to complete the AR effect of the three-dimensional space, as shown in FIG. 7.

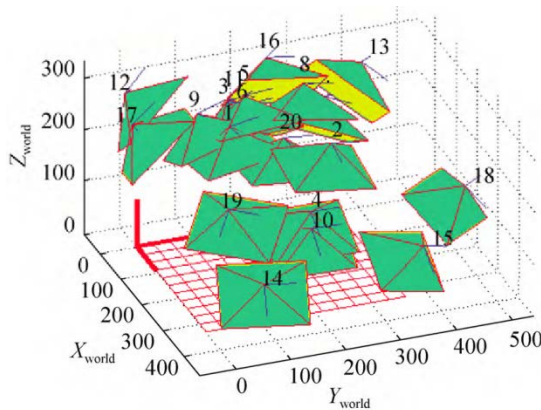


Figure 6 AR three-dimensional modeling **Figure 7** Three-dimension model rendering

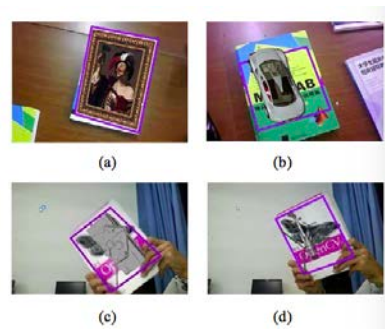
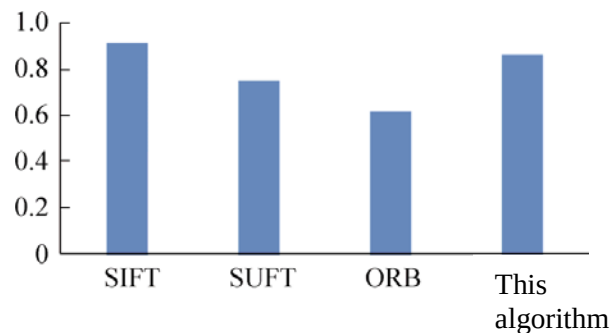
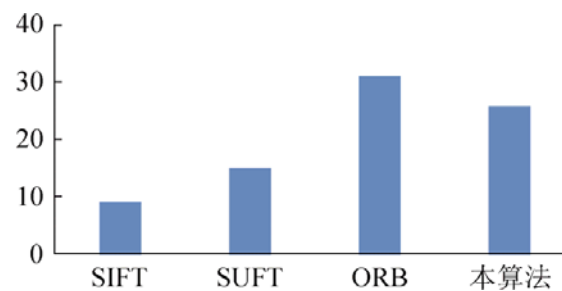
5. Analysis of experimental results

The experimental environment in this paper is a PC with Intel(R) Core(TM) i5-2410m CPU and 8G memory. The experiments in this paper mainly match unmarked targets to achieve AR, including camera calibration, two-dimensional conversion, three-dimensional construction and other methods and steps, and finally achieve the realistic AR effect of two-dimensional plane and AR effect in three-dimensional space.

According to the experimental records of different scenes and different targets shown in FIG. 8, the comparison of accuracy and frame rate of different methods can be obtained, as shown in FIG. 9-10 and table 1.

Table 1 Comparison of test results of different algorithms

Algorithm	Frame number	Successful frame number	Accuracy rate	Frame rate
SIFT	1000	913	91.3	9
SURF	1000	745	74.5	15
ORB	1000	617	61.7	31

**Figure 8** AR renderings**Figure 9** Accuracy comparison between different algorithms**Figure 10** Comparison of frame rate

Combined with the data in table 1 and 2 histograms, it is not difficult to find that the traditional SIFT algorithm has a very good accuracy, but due to the large amount of data processed by features, the performance of its rate is not ideal. The improved SURF algorithm simplifies feature extraction. Although it is not as accurate as SIFT algorithm, its rate performance has been greatly improved. But considering the AR treatment of video streaming, SURF is not going to be very effective. Traditional ORB algorithm combines FAST operator and BRIEF descriptor, greatly reducing the complexity of feature description while retaining a large number of features, which makes it perform very well at rate. However, due to the reduced description complexity, the effect of the traditional ORB algorithm is greatly reduced for some less significant images. In order to make up for the shortcoming of ORB algorithm in image significance, the image is enhanced significantly. Experiments show that the improvement in accuracy is obvious. But it is also because the image pretreatment operation is added that it is not as fast as the ORB algorithm itself. However, its operation efficiency and the requirement of AR matching algorithm are satisfactory.

6. Conclusion

In this paper, the optimized matching algorithm is combined with AR. From the perspective of target feature extraction and target matching, image feature extraction and matching after optimized

pretreatment are conducted, and AR effect is realized through 3D construction. The test results show that this method is effective for various scenarios under more complex environments. The next step is to improve the efficiency and practicability of the algorithm and further improve the application effect of AR based on the usability of AR and multi-core synchronization processing.

7. References

- [1]Yang Jiabo, Yang Gang, Yang Meng. Augmented reality system based on depth image segmentation and object tracking [J]. Journal of system simulation, 2017(11): 2788-2795.
- [2]Zhang C, Gu Y Z, Hu K L, et al. Face recognition using SIFT features under 3D meshes [J]. Journal of Central South University (S2095-2899), 2015, 22(5): 1817-1825.
- [3]Yu B, Li Z, Bo C, et al. Surf feature extraction in encrypted domain [C]//Proceeding of IEEE International Conference on Multimedia and Expo. Chengdu, China, 2014: 1-6.
- [4]Zhuo L, Geng Z, Zhang J, et al. ORB feature based web pornographic image recognition [J]. Neurocomputing (S0925-2312), 2016, 173(3): 511-517.
- [5] Zhou Kaiting, Zheng lixin. Multi-pose Face Recognition Based on Improved ORB Feature [J]. Journal of Computer-Aided Design & Computer Graphics, 2015, 27(2): 287-295.
- [6]Miao S, Wang Z J, Liao R. A CNN Regression Approach for Real-time 2D/3D Registration [J]. IEEE Transactions on Medical Imaging (S0278-0062), 2016, 35(5): 1352-1363.
- [7]Roy H, Bhattacharjee D. Heterogeneous face matching using geometric edge-texture feature (GETF) and multiple fuzzy-classifier system [J]. Applied Soft Computing (S1568-4946), 2016, 46: 967-979.
- [8]Hu S, Qi L. The Laplacian of a uniform hypergraph [J]. Journal of Combinatorial Optimization (S1382-6905), 2015, 29(2): 331-366.
- [9]Zhou Zhiqiang, Wang Bo, Li liguang, et al. Image fusion based on a hybrid decomposition via bilateral and Gaussian filters [J]. Systems Engineering and Electronics, 2016, 38(1): 8-13.
- [10]Rosten E, Porter R, Drummond T. Faster and better: a machine learning approach to corner detection [J]. IEEE Trans. Pattern Analysis and Machine Intelligence (S0162-8828), 2010, 32(1): 105-119.
- [11]Calonder M, Lepetit V, Strecha C, et al. BRIEF: Binary Robust Independent Elementary Features [C]// Proceedings of European Conference on Computer Vision, Heraklion, Greece, 2010: 778-792.
- [12]Hsiao P Y, Lu C L, Fu L C. Multilayered Image Processing for Multiscale Harris Corner Detection in Digital Realization [J]. IEEE Transactions on Industrial Electronics (S0278-0046), 2010, 57(5): 1799-1805.
- [13]Jin Zhanlei, Liu Xuefeng, Wang Chun, et al. Dynamic Error Analysis of Facula Centroid Algorithm Based on CCD [J]. Opto-Electronic Engineering, 2011, 38(6): 7-11.
- [14]Yamashita Y, Wakahara T. Affine-transformation and 2D-projection invariant k-NN classification of handwritten characters via a new matching measure [J]. Pattern Recognition (S0031-3203), 2016, 52(3): 459-470.
- [15]Bok Y, Ha H, Kweon I S. Automated checkerboard detection and indexing using circular boundaries [J]. Pattern Recognition Letters (S0167-8655), 2016, 71(2): 66-72.