An Algorithm of Planar Coded Pattern Recognition for Camera Calibration

WANG Yuzhou
Graphics Lab of EECS
Peking University
Beijing, China.
wyz@graphics.pku.edu.cn

WANG Guoping
Graphics Lab of EECS
Peking University
Beijing, China.
wgp@graphics.pku.edu.cn

***Abstract*—We propose an algorithm for finding out the single or multiple camera calibration planar coded patterns from an image with a complicate background, and provide a kind of patterns design for multi-pattern calibration accordingly. Until recently, the camera calibration planar pattern recognition methods proposed are mainly the recognition of the calibration pattern from images with a single pattern and a simple background. Our approach expands the scope of application of object reconstruction in two aspects. The first is that the reconstruction can be operated in a natural environment instead of an arranged scene. The second is that the large objects, or objects immovable, can be reconstructed conveniently and accurately. We begin by contour detecting with geometric constraints in the given image. Then we build the contours' adjacent relations by generating planar points set Delaunay triangular subdivision. Thirdly, we obtain the correspondence between image contours and pattern elements based on the Delaunay net, pattern geometric constraints and perspective projection invariants. We demonstrate the validity of our approach by providing recognition results on real scene images.**

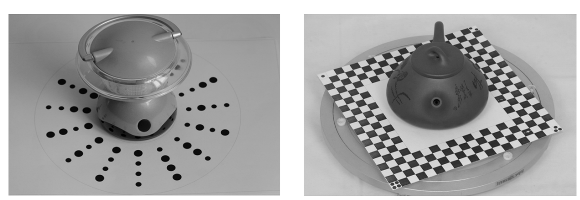
Keywords-computer vision; camera calibration; pattern recognition; image analysis.

I. INTRODUCTION

Images based three-dimensional reconstruction technique is an important technology and a research hot spot in the field of computer vision. Camera calibration is an important step in image based reconstruction, and directly affects the precision of the result of three-dimensional reconstruction. In theory, by a large number of corresponding point pairs matching between images, self-calibration based objects reconstruction method can obtains the camera calibration parameters and the objects reconstruction of geometric surface simultaneously. However, we couldn't always get sufficient non-coplanar point pairs correspondence from the images of the actual scenes. In practice, camera calibration pattern is still necessary for us to get the required accurate three-dimensional reconstruction.

Until recently, images based rigid objects three-dimensional reconstruction is usually done in such a way: Firstly, choose or arrange a suitable scene to ensure that the image background of the target objects taken at every view is simple enough. Secondly, puts a calibration pattern near the target objects, or under the target objects. Thirdly, obtains the images of the target objects at several different views, and makes sure that both the target objects and the calibration pattern should be in the images of every view clearly. Finally, reconstructs the geometric surface of the target objects with these images.

To implement such a way of reconstruction, the pattern design and pattern recognition method should be considered. In the aspect of pattern design, or pattern choice, there are a lot of simple planar patterns can be selected to meet the requirement, such as chessboard, circular dots, circle ring pattern and linear beam pattern. And the pattern recognition is not a difficult thing also, because there is only one pattern in an image and the background of the image is simple. The main work of the pattern recognition for camera calibration is to get pattern feature marks accurately. For the chessboard corner planar pattern, Forstner proposes method [1] to extract corners in image. Chen proposes an approach [2] use a corner detector similar to Harris' algorithm [3] to obtain image's second order Taylor expansion to get its saddle point as a corner location. For the circular dots planar pattern, a common method for elliptic dots geometric center is calculating window's center of gravity with a weight calculated using the square of the grayscale values of pixels in the image. Hu Z. Z. proposes a method [4] to calibrate camera by conics fitting. Meng X Q gives a method [5] to carry out the camera calibration based on circular points. WANG Jun introduce a method [6] to calibrate camera by simple planar pattern with a circle and a rectangular. Zhang G [7] gives a model of ellipse center for perspective projection to deal with the position distortion problem in circular dot positioning. Figure 1 shows two instances of planar calibration pattern and target object under a simple background.



(a) dots pattern under a simple background

(b) chessboard pattern under a simple background

Figure 1. Instances of planar calibration pattern in simple background.

There are several shortages in the method of reconstruction we mentioned above. First of all, it requires a

978-0-7695-4541-7/11 $26.00 © 2011 IEEE
DOI 10.1109/ICIG.2011.146

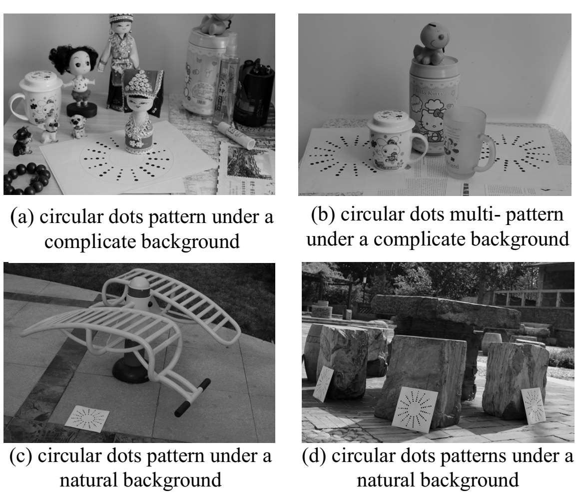
proper environment to simplify the background, so to ensure the calibration pattern could be detected easily and reliably. Because there is few scenes could meet this condition, special arrangement for the scene is needed. Secondly, there is a limitation on the dimension of target objects to be reconstructed. Big objects need big calibration pattern to avoid the occlusion in some views, and the big pattern is inconvenient in making and using. Finally, it is difficult to place the calibration pattern properly to avoid occlusion in all views in the condition that the target object is immovable. And it is difficult also in the arrangement for the background.

To overcome these shortages, we propose a new method to carry out the images based objects reconstruction. Our approach makes use of a set of planar patterns instead of one piece only. The pattern can be placed near the target of objects. There is no limitation on the background of the views, and there is no limitation on the pose and the position of the patterns too. But in our approach, there are two requirements in the aspect of patterns layout. The first is that at least partial pattern should be seen in each view. The second is that in all the views, two different patterns should be co-occurrence directly or indirectly on an image at least once.

Here, the co-occurrence directly means that two different patterns can be seen in an image. The co-occurrence indirectly means that pattern A and pattern B isn't co-occurrence directly, but pattern A and pattern X is co-occurrence directly or indirectly, and B and X is co-occurrence directly or indirectly also.

Our approach doesn't need a special arrangement of the background, doesn’t need to the movement of the target objects, and doesn't need to make and use big pattern. Obviously, our method has advantage over the old one. Figure 2 shows the multi-pattern application instances.

In this paper we will discuss our method in details. In the second section, we discuss the design of planar coded multi-patterns. In the third section, the recognition algorithm is discussed. The fourth section, the experimental results and algorithm analysis will be given, and the final section is the conclusions.



(a) circular dots pattern under a complicate background

(b) circular dots multi- pattern under a complicate background

(c) circular dots pattern under a natural background

(d) circular dots patterns under a natural background

Figure 2. Instances of planar multi-pattern for camera calibrations.

II. PLANAR CODED PATTERN FOR CALIBRATION

Our method needs a series of pattern for multi-pattern calibration. We design a kind of multi-pattern which consists of a series of patterns. These patterns are similar and any one can be distinguished from the others by coding. The number of patterns of the series can be one or more. In the patterns, we use circular dots as marks for camera calibration. The size of dots is one of the different two. We divide all the dots in the patterns into groups. In a group, all dots are arranged in a kind of order. We can assign a status to a dot according to its size. And a group can be assigned a status according its dots’ status and order. We call this status as group status, or group-code. All the group-codes in the patterns should be different to distinguish the groups. We use group-code for marks identification and pattern identification.

To simplify our method, our patterns designed to meet the following conditions:

Mark: in the same pattern set, the marks are dots with one of the two different sizes, and can be distinguished visually.

Group: on the same pattern, marks are divided into several groups according to collinear condition, called mark group; all the mark groups in the patterns have the same number of marks;

Collinear: on the same pattern, the geometric centers of the marks in the same group are collinear. We call this line as group-line;

Concurrent: on the same pattern, any two group-lines intersect at a point, and the point is just the geometric center of the pattern;

Mark equidistance: on the same pattern, the Euclidean distances between the geometric centers of the two adjacent marks in the same group are equal. We call the distances as mark distance.

Group equidistance: On the same pattern, the Euclidean distances between the two adjacent groups are equal. Here, the group distance is defined as the smallest value of all the dot distance between the points of the two groups. The dot distance is defined as the Euclidean distance between the geometric centers of the two markers;

In addition, in a pattern set, every pattern has the same number of marks, the same number of groups, the same mark distance and the same distance of adjacent groups.

Figure 3 shows a pattern set we designed. It includes 4 different patterns and is in according with the pattern set constraints mentioned above. Every pattern has 15 groups. Every group has 6 dots. And every dot has one size of the two.

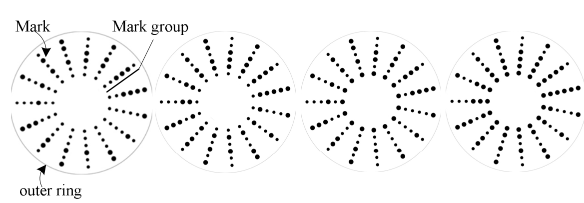


Figure 3. A sample of planar coded pattern set for camera calibration.

III. PLANAR PATTERN RECOGNITION

Through a perspective projection transformation, pattern mark becomes a circle or an ellipse, called as mark image or mark ellipse. The image of a pattern is called pattern image.

In the situation of complicate background calibration, or the situation of multi-pattern calibration, the pattern recognition algorithm should identify all the patterns from the contours in the image. Our algorithm consists of several steps. They are contour detection, contour grouping, group clustering, group decoding and validation.

When we get the correspondence between image marks and the pattern marks by our pattern recognition method, the camera calibration algorithm, such as Tsai's method [8], Zhang's method [9], Heikkila's method [10], Yasutaka's method [11], can be used to get camera calibration results. In this paper, we do not discuss camera calibration algorithms.

A. Contour Detections

In this step, our algorithm detects all contours in the image. At the beginning, most contours we've got are not pattern marks. For the efficiency considerations, contours with a pixel level edge meet the needs of our algorithm's reliability. But we use a sub-pixel level method, the local adaptive thresholding and contour detecting method, to retrieve the nested contour in the given image. Then, we get rid of some contours, which are unlikely to be the image of a pattern mark, according to geometric constraints of the pattern.

1) Local adaptive thresholding: We use local adaptive threshold to transform a grayscale image I_{src} to a binary image I_{dst} according to the formulas:

$$I_{dst}(x,y) = \begin{cases} 1 & if\ \ I_{src}(x,y) > T(x,y) \\ 0 & otherwise \end{cases} \tag{1}$$

where *T(x,y)* is a threshold calculated individually for each pixel. And it is the weighted sum of a $w*w$ neighborhood of the image at *(x,y)*, minus a constant *C*. *(x,y)* is the image coordinate and *w* is the aperture size.

We use Gaussian filter coefficients as the weights. The *w* matrix of separable Gaussian filter coefficients is calculated according the following formula:

$$G_i = \alpha e^{-\left(i-\frac{w-1}{2}\right)^2/(2\sigma)^2} \tag{2}$$

where, i=0, 1,…,w-1 and α is the scale factor chosen so that $\sum_i G_i=1$. w should be odd and positive. The Gaussian standard deviation σ is calculated according the following formula:

$$\sigma=0.3*((w-1)/2-1)+0.8 \tag{3}$$

We choose the value 5 or 7 for w and 4 or 5 for *C*, because we need to get the closed edge of the pattern marks for the following algorithm step of contour retrieving.

2) Contours detecting: We retrieve the non-nested contours using Suzuki's method [12] from the binary image we've got after the step of local adaptive thresholding.

3) Contours removing: We introduce a few basic constraints for the contour removing according to our patterns design.

Condition 1. The area of mark image should be greater than a given threshold. The marks' image should be seen clearly in a view and can be fitting to an ellipse. Therefore, contour with too small area is not the image of a mark.

Condition 2. The area of mark image should be less than a given threshold, because a group of pattern should be within the image of a view at least.

Condition 1 ensures that the mark's image is not too small to be used for decoding. Condition 2 means that the given image should contain enough marks to ensure enough camera calibration points.

Using these constraints, our algorithm can get rid of a large amount of contours, which are either too big or too small, effectively.

B. Contour Grouping

Mark groups on patterns meet some geometric constraints. If contours in the image meet the mark group geometric constraints, they could be the image of a mark. All the contours which cannot meet the group geometric constraints cannot be the image of a mark, and should be getting rid of from the contours set. We will discuss the constraints and algorithm to determine the grouping of image contour in this sub-section.

According to our pattern design, mark groups on a pattern should meet these perspective projection invariant constraints:

Condition 1. On the same pattern, the geometric centers of the marks in the same group are collinear.

Condition 2. The marks in the same group meet the cross ratio invariant;

Condition 3. The number of marks in a group is a constant value.

Based on these constraints, our method constructs contour groups. The algorithm is:

- Building a planar point set. The points in the planar point set are the gravity centers of the contours;
- Constructing a network of Delaunay triangular sub-division on the planar point set, known as Delaunay network;
- Constructing adjacent points groups on the Delaunay network, based on the constraints mentioned above;

At this point, the contour groups have been constructed. All contours not in a group have been discarded. But we are not sure which groups are really the images of mark groups.

C. Pattern Construction

Pattern consists of several groups. But, at this time, we don't know how many patterns could be in the given image, not to mention that a group could belong to which one of the patterns. Our pattern construction procedure is first to divide all the groups into several group clusters. A cluster could be a pattern. Here, we call the cluster as a pattern candidate. Then, we validate the candidates according to the pattern geometric constraints, and remove the invalid groups from the cluster. Our method for pattern construction consists of group clustering, cluster geometric center fitting and groups sorting.

1) Group clustering: According to the pattern geometry, all the group-lines on the same pattern should intersect at the same point, and the intersect point is just the geometric center of the pattern. Therefore, we can divide all the groups into several clusters based on the intersection point constrain and the cross-ratio invariant constraint. Then, we can remove the clusters which only have one group in it.

2) Geometric center fitting: On the consideration of algorithm efficiency, we don't fit geometric center of a cluster in the group clustering step. Now, we begin to fit the center of a pattern candidate with the linear mean square error minimum method with the groups in this cluster. When get the center, we use it to check the fitting error. If the Euclidian distance between a group-line and the center is too large, this group will be removed from the cluster, and the center fitting procedure restart to fit a new center point with the remained groups.

3) Group sorting: Group sorting consists of two steps. One is sorting the marks of a group. Another is sorting the groups of a cluster. For each group, according to the distances between marks and the cluster center, the marks' order can be arranged. As to the groups order arrangement, we can use the group distances as the sorting index.

D. Decoding and Verification

After the group sorting, we can decode the groups according to the contour size, the mark's order in the group, and the encoding rules.

After the decoding, we can verify the recognition result according to coding constraints. First of all, the group-code and the group-code order must be in according with the encoding rules. Secondly, the code difference between two groups must be in according with the intersection angle between the two group-lines. All the groups which do not meet these two constraints will be removed.

At this time, the pattern recognition task completes. The group clusters are the patterns identified.

IV. EXPERIMENTS

Our method can be directly used for applications of three-dimensional reconstruction camera calibration. In this section, we use about 450 real images to evaluate the performance of our algorithm, and use simulation image to evaluate accuracy of the local adaptive thresholding and contour retrieving method.

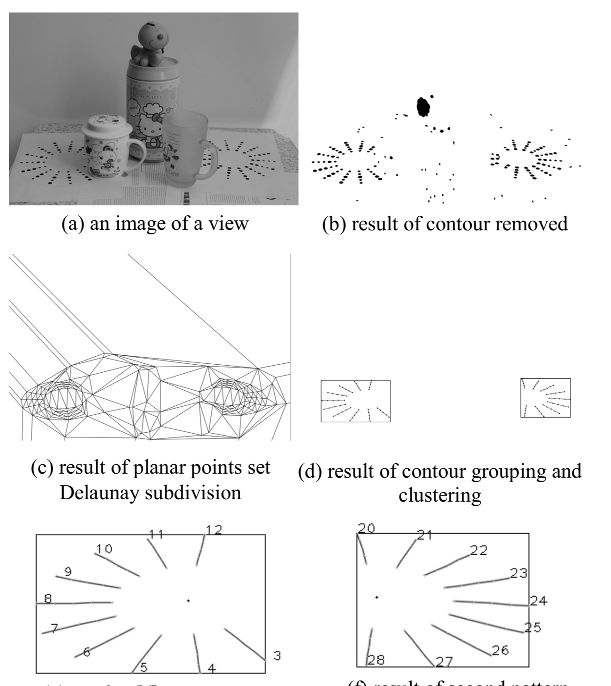
To calibrate a camera, eight points correspondence is sufficient [13]. Therefore, to the pattern we proposed in section 2, two mark groups should be seen at every view. In the multi-pattern calibration situation, it is no need that all the patterns should be seen at all views. What we need is that every two patterns should be partially seen directly or indirectly at least once in all views. This condition is loose and comfortable, and can be met naturally in practice.

A. Algorithm Accuracy

The algorithm accuracy means the marks positioning accuracy and the pattern recognition correctly.

1) Pattern groups identifying accuracy: In our experiments, we use about 450 images to evaluate our algorithms' performance. All the images are taken from 25 real scenes of three-dimensional reconstruction with a natural lighting. The experiments show that all the patterns in the images are recognized correctly, and all the mark group code is calculated accurately. Limit to the paper size, only two examples of the experimental images and results are shown in Figure 4 and Figure 5.

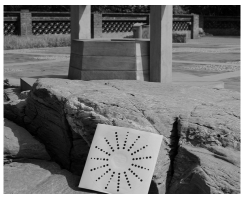
Figure 4 shows an experiment on planar multi-pattern camera calibration. Figure 4(a) shows one image of a series of multi-view image. Figure 4(b) shows the result of contour detection after the contours removing. Figure 4(c) shows the result of planar points set triangular subdivision. Figure 4(d) shows the result of contour grouping and clustering. The two rectangles show the two clusters and their groups. Figure 4(e) and 4(f) shows the pattern recognition result. The number near the line is the group code, and the end of the line section near the number is the header of the group for coding. The codes show that the pattern recognition and decoding is correct, because we do code these two patterns' groups from 1 to 15 and from 17 to 31, respectively.



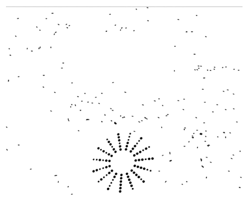
(a) an image of a view
(b) result of contour removed
(c) result of planar points set Delaunay subdivision
(d) result of contour grouping and clustering
(e) result of first pattern recognition
(f) result of second pattern recognition

Figure 4. Planar multi-pattern recognition for calibration in objects reconstruction.

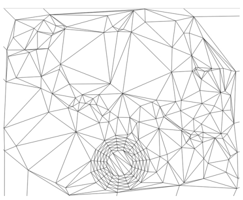
Figure 5 shows an example of outdoor multi-pattern camera calibration. Figure 5(a) shows one view of the scene, but only one pattern can be seen in this view. Figure 5(b) shows the result of contours detected. Figure 5(c) shows the network of Delaunay. Figure 5(d) shows the pattern recognition result. We can identify the pattern the 4th pattern of the four-pattern set from the codes, because the codes are from 49 to 63 respectively.



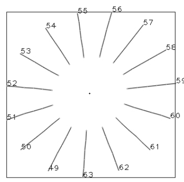
(a) an image of a view



(b) result of contour removed



(c) result of planar points set Delaunay subdivision



(d) result of pattern recognition

Figure 5. Planar multi-pattern recognition for calibration in an outdoor objects reconstruction.

2) Contour retrieving accuracy: We use two of simulation images to evaluate accuracy of our local adaptive thresholding and contour retrieving algorithm. One image is of the planar pattern shown in Figure 2, and the other is one of its perspective projection transformation with the 3*3 transformation matrix={{0.7866, -0.1115, 224.6}, {0.0001, 0.4457, 380.0}, {0.0001, -0.0001, 1.0}}. Gaussian noise is added to the 256 levels of gray images, with zero mean and standard deviation of σ=5, 8, 10, 12, 15, and 20. The pattern mark dots center coordinates are estimated with the retrieved contours, and the coordinates are then compared with the ground truth. The final results of pattern mark dots center position error are shown in Table I and Table II. In the two tables, the unit of position error is pixel. The experiment results show that the algorithm can get the sub-pixel center positioning precision, and the algorithm is not sensitive to Gaussian noise.

TABLE I. ERROR STATISTICS OF DOTS CENTER DETECTING OF THE PATTERN

Position Errors	**Gaussian Standard Deviation**						
	0	***5***	***8***	***10***	***12***	***15***	***20***
average	0.00	0.00	0.00	0.00	0.01	0.01	0.01
standard deviation	0.02	0.02	0.02	0.03	0.03	0.04	0.04

TABLE II. ERROR STATISTICS OF DOTS CENTER DETECTING OF PERSPECTIVE TRANSFORMATION OF THE PATTERN

Position Errors	**Gaussian Standard Deviation**						
	0	***5***	***8***	***10***	***12***	***15***	***20***
average	0.00	0.00	0.01	0.01	0.01	0.01	0.01
standard deviation	0.02	0.02	0.03	0.03	0.04	0.04	0.04

B. *Algorithm Efficiency*

As mentioned above, our method consists of local adaptive thresholding contour extraction, geometric constraints contours removing, planar points set Delaunay subdivision, geometric constraints contours grouping, groups clustering and codes calculating.

Among them, only the first step depends on image grey scale and image size, and needs more time in the processing. All the others are based on geometry, and depend on the number of contours. In fact, through the process of the step two, the number of contours is comparatively small, and there is no much time used in the geometry based processing.

In our experiments, the computer's CPU is Intel Core 2, T5470, 1.6G Hz, and all the real images have the same size of 1072*1063. The experiments show that our algorithm needs about 300ms in contours extraction step and need about 30ms in the other steps to get its results.

V. CONCLUSIONS

The results of our experiments indicate that our method can get perfect planar pattern recognition from the real images. It can be seen from the experiments that our algorithm has advantages in the tasks of pattern recognition. First of all, it can be used to recognize the planar coded pattern for camera calibration under the complicate background. Secondly, it can be used both for single pattern calibration and multi-pattern calibration. Finally, it is accuracy, efficiency and reliability.

At present, the limitation of our algorithm is that we cannot deal with too small pattern marks in image, because of the algorithm using the marks area for coding. We will carry out further research on this problem.

REFERENCES

[1] W. Forstner, E. Gulch. A fast operator for detection and precise location of distinct points, corners and centers of circular features, Proceedings of the ISPRS Workshop on Fast Processing of Photogrammetric Data, Switzerland. 1986. pp.281-305.

[2] Chen D., Wang Y., Zhang G. A new sub-pixel detector for grid target points in camera calibration. 20th Congress of the International Commission for Optics, 2005.

[3] Harris C., Stephens M., A combined corner and edge detector. Proceedings of the Forth Alvey Vision Conference, Manchester, England, 1988, pp.147-151.

[4] Hu Z Z,Tan Z. Camera calibration with conics fitting and circular points. Journal of Xian Jiaotong University, 2006, 40(10), pp.1065-1068.

[5] Meng X Q, Li H, Hu Z Y. A new easy camera calibration technique based on circular points. Proceedings of the British Maehine Vision Conference, Bristol, 2000, pp.496-501.

[6] WANG Jun, Xu Jian-long, Camera calibration method based on planar pattern. Computer Engineering and Design. 2009, 30(1), pp.259-261. (in Chinese)

[7] Zhang G., Wei Z. A position-distortion model of ellipse center for perspective projection. Measurement Science and Technology, 2003, 14, pp.1420-1426.

[8] Tsai R, An eficient and accurate camera calibration technique for 3D machine vision, Proceedings of Computer Vision and Pattern Recognition, 1986, pp.364-374

[9] Z. Zhang, A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell, 2000, 22(11), pp.1330-1334.

[10] Heikkila J, Silven O. A four-step camara calibration procedure with implicit image correction. Proc. Of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997, pp.1106-1112.

[11] Yasutaka Furukawa, Jean Ponce, Accurate Camera Calibration from Multi-View Stereo and Bundle Adjustment, Intl J Computer Vision, 2009, 84, pp. 257-268

[12] S. Suzuki and K. Abe. Topological structural analysis of digitized binary images by border following. CVGIP, 30(1):32–46, April 1985.

[13] Richard Hartley, Andrew Zisserman, Multiple View Geometry in Computer Vision. Cambridge University Press. 2004