高精度多視点カメラ群のキャリブレーション

沈 文裕† ウ 小軍††
若林 佳織†† 小池 秀樹††

近年、監視システムや多視点映像製作など、複数カメラ群を用いるビジョンシステムが多くなってきている。これらのシステム構築のコスト削減のため、簡単操作且つ高精度なキャリブレーションツールが欠かせない。本発表は、簡便さと高精度を両立できるキャリブレーションのツール開発についてまとめる。構成として、まず簡単操作で撮影状況に柔軟に対応できる対応点付けのアルゴリズムを紹介する。提案手法により、レンズひずみや撮影方向に影響されにくい対応点付けが簡単操作で実現できた。次に、外部キャリブレーションについて、カメラ群を複数のサブグループに分ける方法で、段階的にパラメータ推定を行うことで、キャリブレーション精度の維持が実現できた。また、キャリブレーション操作の各ステップにインタラクティブに誤差グラフの作成やパラメータ推定演算の並列実装などの工夫により、大量カメラ群のキャリブレーションタスクの煩雑さを軽減できた。

A High Accuracy Calibration Method for Wide-Area Distributed Multi-viewpoint Camera Group

WENYU SHEN ,† XIAOJUN WU ,†† KAORU WAKABAYASHI ††
and HIDEKI KOIKE††

Camera calibration, playing a key role in three-dimensional reconstruction, has been studied extensively in computer vision and photogrammetry for many years. With the trend of enlarging the scene and increasing the number of cameras, more calibration points need to be matched and besides, some calibration points may be partially captured by a camera, so that calibrating cameras accurately in a wide-area scene becomes a challenge. In order to solve these problems, a high accuracy calibration method for wide-area distributed multi-viewpoint camera group is put forward in this paper. On one hand, homography matrix is calculated iteratively based on the previous result in a proliferation manner to create conjugate pairs of feature points automatically. Meanwhile, a pre-prepared black and white chessboard template is used as the calibration object to improve the stability of the matching process. On the other hand, a multi-hierarchical tree-structural grouping strategy is presented, in which inner-group calibration is conducted on the lowest hierarchy first, after which the calibration results are combined to a unified coordination system in a parallel way through a series of iterative transformation of coordinates and optimization. Experimental results show that the iterative feature points matching algorithm makes feature points matching accurate and automatic even in the condition of a large calibration plane. Moreover, with the multi-hierarchical grouping strategy, the error of extrinsic parameters is tiny even in a wide-area scene with many cameras. In a word, the method works stably and accurately in calibrating distributed multi-viewpoint cameras in a wide-area scene.

Key words: Camera Calibration, Feature Points Matching, Multi-viewpoint Camera Group, 3D Reconstruction

1. Introduction

In real-time three-dimensional (3D) reconstruction systems, vision systems are used to capture thousands of points over a large spatial range, and based on which global surface information of an object is acquired. Therefore, in order to pursue reliable and precise surface information, the accuracy of a vision system is extremely essential[1),2)]. To realize this aim, camera calibration is usually required. It can be said that the precision of a vision system closely rests with the accuracy of camera calibration result.

† 中国・同済大学
Department of Computer Science and Technology, Tongji University, China
†† NTT サイバースペース研究所
NTT Cyber Space Laboratories

The goal of camera calibration is to determine the position and orientation of cameras in the scene and relating camera positions to scene coordinates. The parameters which determine the accuracy of calibration result can be divided into two categories[2),3)]: 1) extrinsic parameters, which determine the position and orientation of a camera in an absolute coordinate system from the projections of calibration points in the scene; 2) intrinsic parameters, which determine the internal geometry of a camera, including the camera constant, the location of the principal point, and corrections for lens distortions. On the other hand, the solutions to these parameters have been discussed in both computer vision and photogrammetry for many years, which can be classified into the following categories[1)]: 1) linear methods[3)~5)]; 2) non-linear methods[3),6)]; 3) multiple-stage methods[3),7),8)]. Multiple-stage methods have become more popular, since they mix both linear and direct nonlinear minimization methods[9)].

In 3D reconstruction problems, the scene becomes larger. Besides, more cameras are used in order to obtain high accuracy, both of which lead to the consequence that the calibration points that are needed increase in both the number and the area that they cover. Then the problems occur: 1) more calibration points need to be matched. All the calibration algorithms used to determine the transformation between coordinate systems assume that a set of conjugate pairs is available. The conjugate pairs are obtained by matching feature points between views. In fact, calibration algorithms can be ill conditioned due to those mismatched feature points; 2) some calibration points may be out of a camera view, owning to the larger scene and the limited camera view.

To solve these problems, the paper puts forward a high accuracy calibration method for wide-area distributed multi-viewpoint camera group. Firstly, in order to create accurate matching of feature points automatically in the case that not all the feature points can fill in a camera view, an iterative matching algorithm is put forward. In the meantime, a black and white chessboard template is introduced to provide with more stability in both interior and exterior calibration than that which directly matches feature points between camera images. In addition, the calibration method in this paper is based on the techniques mentioned above, but it introduces an idea of multi-hierarchical camera grouping[10], which firstly calibrates cameras in the lowest layer groups separately, and then using the calibrated results to calibrate cameras in the upper layer groups in an iterative way until it reaches the base (top) group, when all the cameras are in a unified coordinate.

2. Feature Points Matching Algorithm

Feature points matching problem is always the basis of camera calibration. Those mismatched feature points, to some extent, may cause a huge error in final calibration result, so to determine the matching relationship of feature points in advance is essential[3)]. However, with the enlargement of scene, the number of feature points is increasing dramatically, so that to manually decide the whole matching becomes time-consuming and sometimes impossible. On the other hand, under the situation when cameras are distributed in a wide area, some feature points may not fit in the view of a camera, which also brings the difficulties to the matching process.

As a practical solution, we employ a homography-based calibration method, where a plan with black and white chess board, as the calibration object, is used to hold calibration points. In this section, the algorithm and explanation is conducted in case of two pin-hold cameras.

Generally, any change in the position or orientation of an object is a rigid body transformation, since the object moves (changes position or orientation) but does not change size or shape[3)]. The simplest way to represent the transformation between p_1 in the coordinate system of the first viewpoint and p_2 in the coordinate system of the second viewpoint is:

$$p_2 = \mathbb{R}p_1 + \mathbb{T} \qquad (1)$$

where R is a 3×3 orthonormal matrix for rotation, and T is the vector of the amount and direction of translation.

Next, the transformation between scene coordinates (3D points) and pixels (2D points) can be represented by the following equation[10)]:

$$\begin{pmatrix} i \\ j \\ 1 \end{pmatrix} = \mathbb{K} \cdot \begin{pmatrix} & \mathbb{R} & & \mathbb{T} \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \qquad (2)$$

where (i, j) represents pixel coordinates, while (x, y, z) represents scene coordinates. $\mathbb{K}$ is a 3×3

matrix which represents intrinsic parameters. Define $\mathbb{P}$ as below:

$$\mathbb{P} = \mathbb{K} \cdot \begin{pmatrix} & \mathbb{R} & & \mathbb{T} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3)$$

$\mathbb{P}$ is a 3×4matrix which combines the rotation and translation factors into a single matrix.

Moreover, the source 3D point is constrained on a 2D plane in the case of homography, where the projection equation can be simplified as below:

$$\begin{pmatrix} i \\ j \\ 1 \end{pmatrix} = H_\pi \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} \quad (4)$$

Suppose that the point (x', y') on the plane π is projected to two screens in different directions, while (i, j) represents the pixel coordinates on the screen. H_π is a 3×3 matrix.

Let (i_1, j_1) and (i_2, j_2) denote the points on the two screens respectively and H_{π_1} and H_{π_2} be the matrices defined in Equation (4). The transformation between (i_1, j_1) and (i_2, j_2) can be defined as following[10]:

$$\begin{pmatrix} i_2 \\ j_2 \\ 1 \end{pmatrix} = H_{\pi_2} \cdot H_{\pi_1}^{-1} \cdot \begin{pmatrix} i_1 \\ j_1 \\ 1 \end{pmatrix} = H \cdot \begin{pmatrix} i_1 \\ j_1 \\ 1 \end{pmatrix} \quad (5)$$

where the regular matrix H is called as the homography matrix. At least four conjugate pairs are needed to estimate the matrix H.

Given the vector $\mathbb{N} = (p, q, -r)$ that is the normal vector of the plane π and r that is the Z-intercept of the plane π in the coordinate of the first camera, the homography matrix can be obtained from the following equation:

$$H = (\mathbb{I} + \mathbb{N} \cdot \mathbb{T}^T / r) \cdot \mathbb{R} \quad (6)$$

On the other hand, the rotation $\mathbb{R}$ and translation $\mathbb{T}$ between the two camera coordinates can also be calculated from H easily[10].

As mentioned above, the algorithm the paper puts forward is based on the fact that the calibration object used is a plane with a black and white chess board. Since at least four conjugate pairs are needed to calculate the homography matrix H , in fact, at least the corner points of one unit square can be used as the input to calculate H . However, the accuracy of estimation of H will increase with more input of conjugate pairs. Besides, another fact is that due to the similarity in position, it is always more accurate if homography matrix calculated is used to predict the matching relationship of the feature points around those, the matching relationship of which have been already determined, than those which locate far away. Based on these facts, the algorithm is represented below:

- Input: 1) Two images to be matched; 2) position coordinates of all the feature points that is going to be matched; 3) four conjugate pairs, which have been manually matched, with each point representing a corner of a unit square (Note: the unit square can be arbitrarily selected, but it is recommended to select the clearest square occupying the largest area in the image); 4) error threshold.
- Output: the homography matrix H between two 2D planes.
- Steps:
  (1) Record this initial conjugate pairs' matching relation.
  (2) Based on the initial conjugate pairs' matching relation, calculate an initial homography matrix and let it be the current homography matrix.
  (3) Push the initially selected square into a queue.
  (4) Pop a square out of the queue and let it be the current square.
  (5) Find all of its neighbor squares and push them to the queue one by one. If the square was once pushed in the queue, it can be neglected.
  (6) If the current square is the first square, the corner points of which were matched manually, go to (4). Otherwise, continue with (7).
  (7) Based on the currently used homography matrix, figure out four predicted positions of points that correspond to the four corner points of the current square. If the predicted position is out of the image scope, it should be omitted, since some of the calibration points may not fit in the view of the camera.
  (8) Try to search for the corresponding feature points which are closest to the estimated four positions of the points that have been calculated above. If the closest distance between the predicted position and real position is greater than the error threshold that was set in advance,

it can be omitted directly.

(9) Record this conjugate pairs' matching relation.

(10) Using all the recorded conjugate pairs' matching relations to calculate the new homography matrix and let the new matrix be the currently used homography matrix.

(11) Check if the queue is empty. If there are no elements any more, go to (12). Otherwise, go to (4) again.

(12) Output the final calculated homography matrix.

The finally calculated homography matrix represents the real matching relationship between two images, based on which, all the conjugate pairs have been found.

It should be noted that in the interior and exterior calibration in this method, a black and white chessboard template is introduced to improve stability. Instead of calculating the homography matrix between two camera images, the homography matrix between a camera image and a black and white chessboard template image is calculated here. Due to the internal errors in a camera, errors are introduced in a camera image as well, so if two camera images are used directly, the homography matrix that is calculated should not be ideal, maybe with a huge error. It is the template image that decreases this error. Meanwhile, it is also easy to obtain the homography matrix between two camera images just through a simple conversion, but with a more accurate matching.

3. Interior Calibration

Perfect pinhole camera model is employed in this method, so before cameras are grouped and exterior calibration is performed, it is better to compensate for errors in the construction of cameras so that the bundle of rays inside the cameras obeys the assumptions of perspective projection. The camera parameters are called the intrinsic parameters, which can be represented by a set of camera parameters (camera constant, principal point, lens distortion coefficients and scale factors), as opposed to the extrinsic parameters for the exterior calibration of the camera.

Some calibration algorithms solve the interior calibration problems and exterior calibration problems at the same time. The motivation for this is that

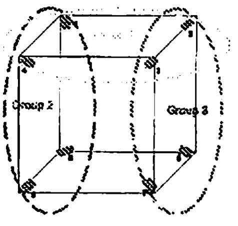

Fig. 1 Calibration Scene 1

the true location of the calibration points on the image plane cannot be known until the exterior orientation of the camera has been determined. However, the interior calibration problem can be solved by itself.

Lens distortions include two components (radial distortion and decentering). The radial distortion and decentering effects can be modeled as polynomials. The interior calibration algorithm estimates the coefficients of these polynomials.

Let (x', y') denote the true image coordinates and $(\tilde{x}, \tilde{y})$ denote the uncorrected image coordinates obtained from the pixel coordinates i and j using an estimate for the location of the principal point $(\hat{c}_x, \hat{c}_y)$:

$$\begin{aligned} \tilde{x} &= \ j - \hat{c}_x \\ \tilde{y} &= -(i - \hat{c}_y) \end{aligned} \tag{7}$$

The corrections $(\delta x, \delta y)$ will be added to the uncorrected coordinates to get the true image plane coordinates:

$$\begin{aligned} x' &= \tilde{x} + \delta x \\ y' &= \tilde{y} + \delta y \end{aligned} \tag{8}$$

The corrections for radial lens distortions that include tangential distortions due to effects such as lens decentering are modeled by a polynomial[3]:

$$\begin{aligned} \delta x &= (\tilde{x} - x_p)(\kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6) \\ \delta y &= (\tilde{y} - y_p)(\kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6) \end{aligned} \tag{9}$$

where (x_p, y_p) is the refinement to the location of the principal point:

$$r^2 = (\tilde{x} - x_p)^2 + (\tilde{y} - y_p)^2 \tag{10}$$

Generally, using the feature points matching results, the parameters of $\kappa_1, \kappa_2, \kappa_3, x_p, y_p$ can be estimated by some well known non-linear optimization methods.

4. Multi-hierarchical Camera Grouping Strategy

In 3D reconstruction problem, recently there is a trend in enlarging the scene, so that more objects

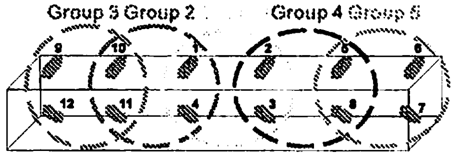

Fig. 2 Calibration Scene 2

can be included in it, which leads to the problem put forward above that the calibration object increases in size. This contributes to the problem that part of the calibration object is not in the camera view. To calibrate cameras under this situation, a multi-hierarchical camera grouping strategy is put forward in this paper.

To begin with, suppose the following two situations as shown as Fig.1 and Fig.2.

A black and white chessboard plane is used as the calibration object in these two situations. No matter where it is put, it can only be seen by some of those cameras. Besides, it is also possible that although seen by a camera, maybe only part of it is in the view. Therefore, a kind of grouping strategy is needed, so that all the cameras in the scene can be calibrated simultaneously and accurately. To illustrate the grouping strategy more clearly, some related definitions and constrains are given below.

The multi-hierarchical camera grouping strategy in this paper is an iterative process that deals with a tree-like multi-hierarchical structure, the nodes of which are called groups. The root of the tree, known as the base group, is relied on by other nodes. A group is called a child group as long as that the intersection between it and another group which is closer to the base group, known as the parent group, is not empty. One camera in the intersection is selected as the core camera of the group. In Fig.1, Group 1 can be the base group, while Group 2 and Group 3 can be the child groups of Group 1. Camera 1, 2, 3, 4 can be selected as the core camera of Group 1, while Camera 1, 4 and Camera 2, 3 can be selected as the core camera of Group 2, 3 respectively. Notably, there are 3 layers in the grouping structure in Fig.2. The graphical illustration of the hierarchy with group ids on the nodes and core camera ids on the relation arrows can be presented in Fig.3.

On the other hand, there should be one and only one base group. Besides, isolated nodes are not permitted. Furthermore, one camera can belong to two groups, only if the two groups have the parent-child relationship. At this time, the camera is the candidate for the core camera of the child node. Last but not least, although the strategy supports multiple hierarchies, in order to pursue accuracy, the grouping decision that the layer number is greater than three is not recommended. In a word, the less layer number, the more accurate calibration result. That is why although there are many grouping methods for Scene 1, the method illustrated in Fig.1 is the best. However, the grouping method for Scene 2 has three layers, since there are no better grouping solutions.

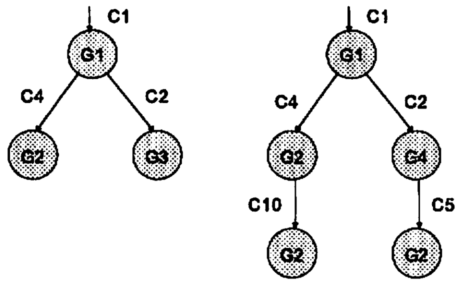

Fig. 3 Grouping hierarchies of Scene 1 and Scene 2

Based on the above definitions and constrains, the multi-hierarchical camera grouping strategy is proposed below:

- Step 1: According to the definitions and constrains above, divide all the cameras into several hierarchical groups and decide the base group and the core cameras of all the groups.
- Step 2: Within each group, calculate the homography matrices between the core camera's image and other cameras' images. Assume N ($N \geqq 2$) is the number of cameras in the group. The result set of homography matrices can be denoted as $\{H\} = \{H_{2,1}, H_{3,1}, \ldots, H_{N,1}\}$ with N representing the camera that is not the core camera and 1 representing the core camera. Since a black and white chessboard template is employed, suppose the template is the image of a virtual camera 0. The calculation can be further divided into the following sub-steps:
  - (1) Determine the position of the black and white chessboard calibration plane.
  - (2) Obtain images of the calibration plane from N cameras.
  - (3) For each pair of camera image and template image, using the feature points matching algorithm introduced in this paper to find all the M conjugate pairs. Let $C_i (i = 0, 1, 2, \ldots, N)$ denote the im-

age plane coordinate system of each camera, so $r_{j,i}(j = 1, 2, \ldots, M)$ denote the feature point J in C_i.

(4) Calculate the set of the homography matrices between the camera images and the virtual camera image. Denote these matrices as $\{H'\} = \{H_{1,0}, H_{2,0}, \ldots, H_{N,0}\}$

(5) Let $\{H\} = \{H_{1,0}, H_{2,1}, \ldots, H_{N,N-1}\}$. The coordinate of the j-th point in coordinate system C_i can be calculated through its corresponding coordinate in C_k by the following function

$$r'_{j,i} = \begin{cases} \left(\prod_{n=k}^{i} H_{n,n-1}\right) \cdot r_{j,k} & i > k \\ \left(\prod_{n=i}^{k} H^{-1}_{n,n-1}\right) \cdot r_{j,k} & i < k \end{cases} \quad (11)$$

Furthermore, $H_{n,n-1}$ can be calculated by the equation below:

$$H_{n,n-1} = H_{n,0} \cdot H^{-1}_{n-1,0} \quad (12)$$

(6) Define the error $e_{j,i}$ on j-th point in C_i as below:

$$e_{j,i} = |r'_{j,i} - r_{j,i}| \quad (13)$$

(7) Through performing the optimization for minimizing the sum of all error values, the more accurate homography set $\{\overline{H}\} = \{\overline{H_{1,0}}, \overline{H_{2,1}}, \ldots, \overline{H_{N,N-1}}\}$ is estimated[10].

(8) By decomposing the corresponding homography matrix, h) The rotation $\mathbb{R}_{l,1}$ and translation $\mathbb{T}_{l,1}$ from C_1 to $C_l (l = 2, 3, \ldots, N)$ can be calculated.

- Step 3: Transform the extrinsic parameters of cameras in the current group and the groups in the subtree which regards the current group as their root to the unified coordinate system of the parent group. The transformation is performed from the leaves (groups on the lowest layer) of the tree, so, to begin with, the group which is farthest from the base group in relation is firstly selected as the current group. Due the constrain that one camera can belong to two groups, only if the two groups have parent-child relationship, the process can start from each leaf simultaneously and in practice, threads are used owing to this parallelization. In fact, the unified coordinate system mentioned here can be the coordinate system of the core camera in the parent group. Besides, the transformation is conducted through the core camera in each group. Take Scene 1 as the example again. In order to transform the extrinsic parameters of Camera 8 to the coordinate system of Camera 1 (the core camera in the base group), they are firstly transformed to the coordinate system of Camera 4 (the core camera in the same group), after which they are further transformed to the coordinate system of Camera 1, since camera 4 either belongs to Group 2 and Group 1.
- Step 4: Optimize the extrinsic parameters of cameras in the current group, the groups in the subtree and the parent group to decrease the error caused by the above transformation.
- Step 5: Check whether the parent group of the current group is the base group or not. If it is, the thread of calibration process ends with the extrinsic parameters of the related cameras optimized. Otherwise, let the parent group be the current group and go back to Step 3. The process iterates until the parent group becomes the base group. If all the process threads end successfully, then the extrinsic parameters of all the cameras are optimized.

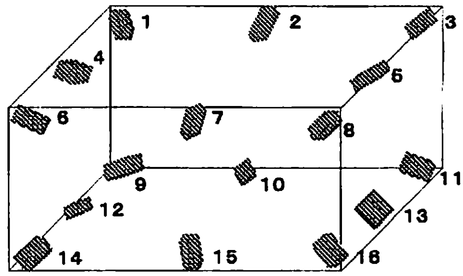

Fig. 4 The Layout of the Experimental Scene

5. Experimental Results

The experiment is conducted in a closed scene which is approximately 3 meters in height and 5 meters in length and width. Besides, 16 cameras are used to be calibrated with 8 cameras on the edge of the roof and 8 cameras on the edge of the bottom respectively. The layout of the scene with 16 cameras can be illustrated in Fig.4. In addition, all of the 16 cameras used in this experiment are SONY XCD-X710CR/X710 with output signal format of 1024×768 (horizontal/vertical).

In the first experiment, the feature points matching algorithm was tested. According to the al-

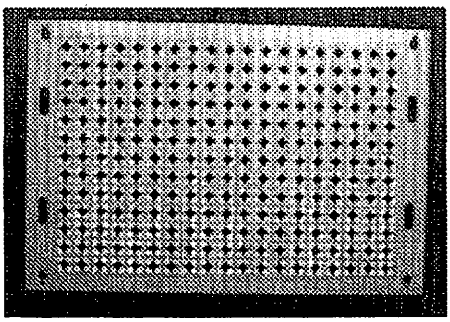

Fig. 5 Successful Matching Result

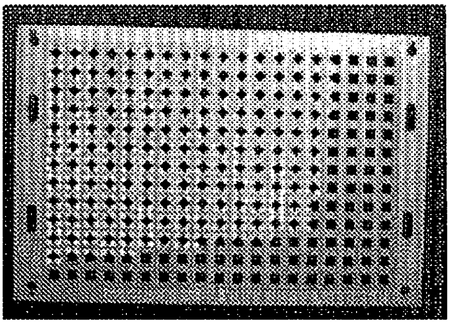

Fig. 6 Failure Matching Result

gorithm, manually matching 4 corner points of a square is needed at first. Since the matching is processed iteratively from the manually matched square in a proliferation manner, any complete square in the image can be selected. However, it is easy to be understood that the largest and clearest square should always be the best choice. This experiment started with the matching of the top-left square in the image. Fig.5 demonstrates that using the iterative matching algorithm in this paper, all the feature points are well matched, whereas although under the same condition, the matching process failed if non-iterative algorithm was conducted with only the knowledge of the manual matching relations, which can be demonstrated in Fig.6.

In the successful example, it can be seen from Fig.7 that after the iterative matching process, the distance between the real position and the estimated position decreased drastically with the greatest error less than 0.45 pixels and the average error around 0.1-0.5 pixels, compared with that during the process, in which the greatest error jumped to more than 2 and the average error fluctuated around 0.5, because more corrected matching pairs

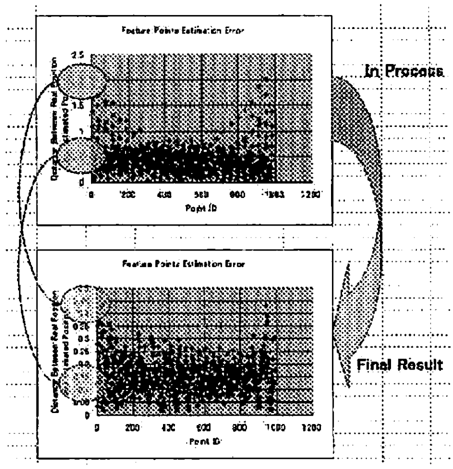

Fig. 7 Comparison of Estimation Errors between Final Result and Result in Process

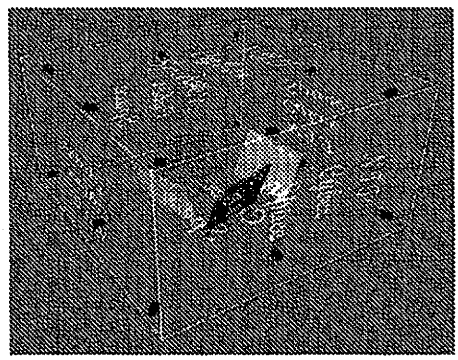

Fig. 8 Demonstration of Exterior Calibration Result

had been collected with the iterative process, which could provide more accurate homography relations between two images. The greater error occurred at the beginning of the matching process.

In the second experiment, all the cameras in the scene were divided into 5 groups. According to Fig. 4, Group 1 (the base group) : 1, 3, 6, 8; Group 2: 4, 6, 9, 12, 14; Group 3: 7, 8, 14, 15, 16; Group 4: 3, 5, 11, 13, 16; Group 5: 1, 2, 9, 10, 11. Using the grouping strategy in this paper, the demonstration of the final calibration result is illustrated in Fig.8, with the red plane corresponding to the calibration result of Group 2, the green to Group 3, the blue to Group 4 and the gray to Group 5 respectively.

According to the grouping strategy, there needs to be two stages of optimization in order to obtain the accurate extrinsic parameters: inner-group optimization and inter-group optimization. Take

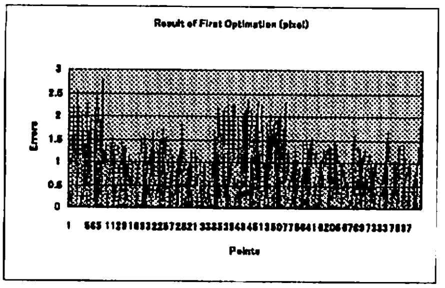

Fig. 9 Result of First Optimization

Group 3 as an example, after the inner-group optimization, the errors which are demonstrated in Fig. 9. The errors are regarded to be very small.

6. Conclusion

To meet the need of stable and accurate calibration for distributed multi-viewpoint cameras in a wide-area scene, a high accuracy calibration method is put forward in this paper. The method is conducted by introducing an iterative feature points matching algorithm and a multi-hierarchical camera grouping strategy. Finally the experimental result certifies that the method works well with only tiny errors and it makes the calibration of multiple cameras in a wide-area scene possible, easier and accurate.

References

1) Shen, T. S. and Menq, C.: Automatic camera calibration for a multiple-sensor integrated coordinate measurement system, *IEEE Trans. Robot. Automat.*, Vol. 17, No. 4, pp. 502 - 507 (2001).

2) Shen, T. S.: *Multiple-sensor integration for rapid and high-precision coordinate metrology*, PhD Thesis, Dep. Mech. Eng., Ohio State Univ., Columbus (2000).

3) Jain, R., Kasturi, R. and Schunck, B. G.: *Machine Vision*, McGraw-Hill, Inc. (1995).

4) Shih, S. W., Hung, Y. P. and Lin, W. S.: Accurate linear technique for camera calibration considering lens distortion by solving an eigenvalue problem, *Opt. Eng.*, Vol.32, pp.138 - 149 (1993).

5) Abdel-Aziz, Y. I. and Karara, H. M.: Direct linear transformation into object space coordinates in close-range photogrammetry, *Proc. Symp. on Close-Range Photogrammetry*, pp. 1 - 18 (1971).

6) Wei, G. Q. and Ma, S. D.: Implicit and explicit camera calibration: Theory and experiments, *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 16, pp. 469 - 480 (1994).

7) Gang, C. and Che, R.: Stereo vision sensor field calibration method based on volume template, *Optics and Precision Engineering*, Vol. 12, No. 6, pp. 626 -631 (2004).

8) Tsai, R. Y.: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses, *IEEE Trans. Robot. Automat.*; Vol. RA-3, pp. 323 - 344 (1987).

9) Xu, Q. Y.: Accurate Camera Calibration with New Minimizing Function, *IEEE int. Conf. Robot. Automat.*, Kunming, China (2006).

10) Wu, X.: *Parallel pipeline volume intersection for real-Time 3D shape reconstruction on a PC cluster*, PhD Thesis, Dep. Elec. Eng., Kyoto Univ., Japan (2005).