Multi-Camera Calibration with One-Dimensional Object under General Motions

L. Wang,   F. C. Wu and Z. Y. Hu

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, P. O. Box 2728, Beijing, P.R.China 100080
{wangliang, fcwu, huzy}@nlpr.ia.ac.cn

Abstract

It is well known that in order to calibrate a single camera with a one-dimensional (1D) calibration object, the object must undertake some constrained motions, in other words, it is impossible to calibrate a single camera if the object motion is of general one. For a multi-camera setup, i.e., when the number of camera is more than one, can the cameras be calibrated by a 1D object under general motions? In this work, we prove that all cameras can indeed be calibrated and a calibration algorithm is also proposed and experimentally tested. In contrast to other multi-camera calibration method, no one calibrated "base" camera is needed. In addition, we show that for such multi-camera cases, the minimum condition of calibration and critical motions are similar to those of calibrating a single camera with 1D calibration object.

1. Introduction

In 3D computer vision, camera calibration is a necessary step in order to extract metric information from 2D images. According to the dimension of the calibration objects, the camera calibration techniques can be roughly classified into four categories. In 3D reference object calibration an object with known geometry in 3D space is used [1, 15, 16]. In 2D plane based calibration planar patterns are used [8, 12, 18]. In 1D object based calibration a 1D segment with three or more markers is used [2, 17, 19]. In 0D approach or self-calibration only correspondences of image points or some special kinds of motion are used and it does not use any physical calibration objects [3, 6, 7, 10, 13].

In the above techniques, much work has been done expect for 1D object based calibration. Camera calibration using 1D object was proposed by Zhang in [19]. Here the 1D calibration object consists of a set of three or more collinear points with known distances. The motion of the object is constrained by one point being fixed. Hammarstedt *et al.* analyze the critical configuration of 1D calibration and provide simplified closed-form solutions in Zhang's setup [2].

Wu *et al*. prove that the rotating 1D object used in [19] is essentially equivalent to a familiar 2D planar object, and such equivalence still holds when 1D object undergoes a planar motion rather than the rotation around a fixed point [17].

The advantages of using 1D objects for calibration are: (1). 1D objects with known geometry are easy to construct. In practice, the 1D object can be constructed by marking three points on a stick. (2). In a multi-camera setup, all cameras can observe the whole calibration object simultaneously, which is a prerequisite for calibration and hard to satisfy with 3D or 2D calibration objects. A shortcoming of the 1D calibration is that the 1D object should be controlled to undertake some especial motions, such as rotations around a fixed point and planar motions. If the motions of 1D object are of general rigid motions, can cameras be calibrated? The problem will be discussed in this work. We know that camera calibration is impossible by using a single camera to observe the 1D object under general rigid motions [19]. However, the following result in this paper is proved: If in a multi-camera, all cameras synchronously observe the 1D object undergoing at least 6 times general motions (assuming these motions do not lie on a conic), then the intrinsic parameters and the pose relations of all these cameras can be calibrated. It is different with [5] that there is no need of one "base" camera being calibrated in advance.

The paper is organized as follows. Some preliminaries are introduced in Section 2. In Section 3 calibration algorithm for the camera set is presented. Then calibration experiments are reported in Section 4. Section 5 are some concluding remarks.

2. Preliminaries

2.1. Camera model

In this paper, a 2D point is denoted by $\mathbf{m} = [u, v]^T$, a 3D point by $\mathbf{M} = [X, Y, Z]^T$. The corresponding homogeneous vector is denoted respectively by $\tilde{\mathbf{m}} = [u, v, 1]^T$, $\tilde{\mathbf{M}} = [X, Y, Z, 1]^T$. With the standard pinhole camera model, the relationship between a 3D point $\tilde{\mathbf{M}}$ and its image
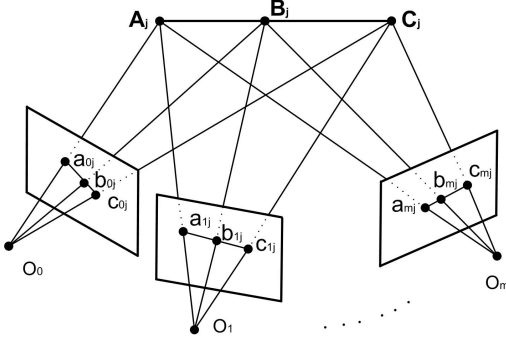
Figure 1. Illustration of 1D calibration objects.

point $\tilde{\mathbf{m}}$ (perspective projection) is given by

$$s\tilde{\mathbf{m}} = \mathbf{K}[\mathbf{R}|\mathbf{t}]\tilde{\mathbf{M}}, \quad \mathbf{K} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where s is a scale factor (projection depth of 3D point $\tilde{\mathbf{M}}$), $\mathbf{P} = \mathbf{K}[\mathbf{R}|\mathbf{t}]$ is called the camera matrix, $[\mathbf{R}|\mathbf{t}]$, called the extrinsic matrix, is the rotation and translation which relates the world coordinate system to the camera coordinate system. And $\mathbf{K}$ is called the camera intrinsic matrix, with α, β denotes the scale factors in the image u and v axes, γ the skew, and $[u_0, v_0]$ the principal point.

2.2. 1D calibration object

Assume 1D calibration object has three points say $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $||\mathbf{A} - \mathbf{C}|| = d_1$, $||\mathbf{B} - \mathbf{C}|| = d_2$. (Here we only consider the minimal configuration of 1D calibration object which consists of three collinear points.) For the convenience of statement, the 1D calibration object is also said the line-segment ($\mathbf{ABC}$). Moreover, the line defined by the line-segment ($\mathbf{ABC}$) is denoted by $\mathbf{L_{ABC}}$.

3. Calibration algorithm

Refer to Figure 1. Given the image points $\{\mathbf{a}_{ij}, \mathbf{b}_{ij}, \mathbf{c}_{ij}|j = 1, 2, ..., n, i = 0, 1, 2, ..., m\}$ of the line-segment ($\mathbf{ABC}$) under the jth rigid motion in the i^{th}camera, our goal is to compute the metric projection matrices under the 0th camera coordinate system:

$$\begin{cases} \mathbf{P}_0^{(e)} = \mathbf{K}_0[\mathbf{I}|\mathbf{0}] \\ \mathbf{P}_1^{(e)} = \mathbf{K}_1[\mathbf{R}_1|\mathbf{t}_1] \\ ... \\ \mathbf{P}_m^{(e)} = \mathbf{K}_m[\mathbf{R}_m|\mathbf{t}_m] \end{cases} . \quad (2)$$

The parameters of multi-camera system can be linearly determined. Firstly vanishing points of 1D calibration objects are computed. With these vanishing points, infinite homographies between cameras can be computed. Then the affine

projection matrices and the metric projection matrices can be gained. Unlike the traditional stratified calibration, our algorithm does not need any prior knowledge on the cameras. And unlike the existing 1D calibration methods, 1D calibration objects can undertake general rigid motions instead of especial motions [17, 19] and there is no need of one calibrated "base" camera [5].

3.1. Affine calibration

Let the 1D calibration object undertakes a series of general rigid motions in the field of multi-camera's view. The correspondence of image points $\{\mathbf{a}_{ij}, \mathbf{b}_{ij}, \mathbf{c}_{ij}|j = 1, 2, ..., n, i = 0, 1, 2, ..., m\}$ can be established.

Since the geometry of 1D calibration object is known, we can compute the vanishing point $\mathbf{v}_{ij}$ of the line $\mathbf{L}_{\mathbf{A}_j\mathbf{B}_j\mathbf{C}_j}$ in the ith camera. The simple ratio of the collinear points $\mathbf{A}_j$, $\mathbf{B}_j$ and $\mathbf{C}_j$ is

$$Simple(\mathbf{A}_j, \mathbf{B}_j; \mathbf{C}_j) = d_1/d_2 \quad (3)$$

Then the cross ratio of the collinear points $\{\mathbf{A}_j, \mathbf{B}_j; \mathbf{C}_j, \mathbf{V}_{j\infty}\}$ is also d_1/d_2, *i.e.*

$$Cross(\mathbf{A}_j, \mathbf{B}_j; \mathbf{C}_j, \mathbf{V}_{j\infty}) = d_1/d_2 \quad (4)$$

where $\mathbf{V}_{j\infty}$ is the infinite point of the line $\mathbf{L}_{\mathbf{A}_j\mathbf{B}_j\mathbf{C}_j}$. By the perspective transformation preserving the cross ratio, we can obtain the linear constraints on $\mathbf{v}_{ij}$:

$$Cross(\mathbf{a}_{ij}, \mathbf{b}_{ij}; \mathbf{c}_{ij}, \mathbf{v}_{ij}) = d_1/d_2. \quad (5)$$

Hence, we can obtain the vanishing points $\mathbf{v}_{ij}$.

Since $\{\mathbf{v}_{0j} \leftrightarrow \mathbf{v}_{ij}|j = 1, 2, ..., n\}$ are the image point correspondences of the infinite plane π_∞, the infinite homography between the 0th camera and the ith camera satisfy the following equations:

$$\mathbf{H}_{i\infty}\tilde{\mathbf{v}}_{0j} = \lambda_{ij}\tilde{\mathbf{v}}_{ij}, (i = 1, 2, ..., m; j = 1, 2, ..., n) \quad (6)$$

Eliminating unknown scale factors λ_{ij}, we can obtain linear constrained equations:

$$[\tilde{\mathbf{v}}_{ij}]_\times \mathbf{H}_{i\infty}\tilde{\mathbf{v}}_{0j} = \mathbf{0}, (i = 1, 2, ..., m; j = 1, 2, ..., n). \quad (7)$$

By solving the linear equations (7), we can determine the infinite homographies $\mathbf{H}_{i\infty}$.

With homographies $\mathbf{H}_{i\infty}$ and image points $\{\mathbf{a}_{ij}, \mathbf{b}_{ij}, \mathbf{c}_{ij}\}$, the projective reconstruction of points and cameras can be computed simultaneously with the technique of projective reconstruction using planes [11]. Considering that the plane inducing homographies $\mathbf{H}_{i\infty}$ is the infinite plane, the computed structure and motion are affine. So we have the affine camera matrices:

$$\begin{cases} \mathbf{P}_0^{(a)} = [\mathbf{I}|\mathbf{0}] \\ \mathbf{P}_1^{(a)} = [\mathbf{H}_{1\infty}|\mathbf{e}_1] \\ ... \\ \mathbf{P}_m^{(a)} = [\mathbf{H}_{m\infty}|\mathbf{e}_m] \end{cases} . \quad (8)$$

and the affine reconstructions of the space points $\{\mathbf{A}_j^{(a)}, \mathbf{B}_j^{(a)}, \mathbf{C}_j^{(a)}\}$.

3.2. Metric calibration

Based on the affine projection matrices showed in (8), the metric camera matrices must be of the following form [4]:

$$\mathbf{P}_i^{(e)} = \mathbf{P}_i^{(a)}\mathrm{diag}(\mathbf{K}_0, 1), (i = 0, 1, 2..., m) \quad (9)$$

and the metric reconstruction of the space points $\{\mathbf{A}_j^{(e)}, \mathbf{B}_j^{(e)}, \mathbf{C}_j^{(e)}\}$ satisfy the following equations:

$$\begin{cases} \mathbf{A}_j^{(e)} = \mathbf{K}_0^{-1}\mathbf{A}_j^{(a)} \\ \mathbf{B}_j^{(e)} = \mathbf{K}_0^{-1}\mathbf{B}_j^{(a)} \\ \mathbf{C}_j^{(e)} = \mathbf{K}_0^{-1}\mathbf{C}_j^{(a)} \end{cases}, (j = 1, 2, ..., n) . \quad (10)$$

Here $\mathbf{K}_0$ is the intrinsic parameter matrix of the 0th camera.

Since $\|\mathbf{A}_j^{(e)} - \mathbf{C}_j^{(e)}\| = d_1; \|\mathbf{B}_j^{(e)} - \mathbf{C}_j^{(e)}\| = d_2$, from equations (10) we have the linear constraints on $\mathbf{K}_0$:

$$\begin{cases} (\mathbf{C}_j^{(a)} - \mathbf{A}_j^{(a)})^T \varpi_0 (\mathbf{C}_j^{(a)} - \mathbf{A}_j^{(a)}) = d_1^2 \\ (\mathbf{C}_j^{(a)} - \mathbf{B}_j^{(a)})^T \varpi_0 (\mathbf{C}_j^{(a)} - \mathbf{B}_j^{(a)}) = d_2^2 \end{cases}, \quad (11)$$

where $\varpi_0 = \mathbf{K}_0^{-T}\mathbf{K}_0^{-1}$.

Remark 1. Since the affine transformations preserve the sample ratio of three collinear points, the two equations of (11) are not independent of each other for each j. Hence, as is the calibration method with 1D object under especial motions, in our method that 1D object moves at least 6 times is also necessary for calibration.

Remark 2. From (11), it is not difficult to see that the critical motions of the 1D object are similar to those of the calibration method with 1D object rotating around a known fixed point, *i.e.*, the motion is critical if and only if the infinite points $\{\tilde{\mathbf{V}}_j = \tilde{\mathbf{C}}_j - \tilde{\mathbf{A}}_j | j = 1, 2, \ldots, n\}$ determined by the 1D object lie on a conic.

From (11), we obtain the linear solution of ϖ_0, and obtain the intrinsic parameter matrix, $\mathbf{K}_0$, using Cholesky decomposition of ϖ_0^{-1}. Hence, the metric projection matrices are:

$$\begin{cases} \mathbf{P}_0^{(e)} = [\mathbf{K}_0|\mathbf{0}] \\ \mathbf{P}_1^{(e)} = [\mathbf{H}_{1\infty}\mathbf{K}_0|\mathbf{e}_1] \\ ... \\ \mathbf{P}_m^{(e)} = [\mathbf{H}_{m\infty}\mathbf{K}_0|\mathbf{e}_m] \end{cases} \quad (12)$$

and the metric reconstructions of the space points are:

$$\begin{cases} \mathbf{A}_j^{(e)} = \mathbf{K}_0^{-1}\mathbf{A}_j^{(a)} \\ \mathbf{B}_j^{(e)} = \mathbf{K}_0^{-1}\mathbf{B}_j^{(a)} \\ \mathbf{C}_j^{(e)} = \mathbf{K}_0^{-1}\mathbf{C}_j^{(a)} \end{cases}. \quad (13)$$

Using QR decomposition, we can extract the intrinsic parameter matrix, $\mathbf{K}_i$, and the motion parameters, $\mathbf{R}_i$ and $\mathbf{t}_i$, of the ith camera from the metric projection matrix,

$$\mathbf{P}_i^{(e)} = \mathbf{K}_i[\mathbf{R}_i|\mathbf{t}_i], (i = 0, 1, 2..., m). \quad (14)$$

3.3. Bundle adjustment

The above solution is obtained through minimizing an algebraic distance which is not physically meaningful. We can refine it through bundle adjustment.

Bundle adjustment is a nonlinear procedure involving the projection matrices and space points, attempting to maximize the likelihood of the reconstruction, being equivalent to minimizing the reprojection error when the noise on measured image points has an identical and independent Gaussian distribution. Here bundle adjustment is to solve the following nonlinear minimization:

$$\min_{\hat{\mathbf{P}}_i, \hat{\mathbf{M}}_j} \sum_{ij} d(\mathbf{m}_{ij}, \hat{\mathbf{P}}_i[\hat{\mathbf{M}}_j, 1]^T)^2, \quad (15)$$

where $d(\cdot, \cdot)$ is the geometric distance between two points, $\hat{\mathbf{P}}_i$ and $\hat{\mathbf{M}}_j \in \{\hat{\mathbf{A}}_j, \hat{\mathbf{B}}_j, \hat{\mathbf{C}}_j\}$ are estimated projection matrices and 3D points, $\mathbf{m}_{ij}$ is the detected (measured) image point of 3D point $\mathbf{M}_j$ in the ith camera. Here $\mathbf{A}_j$, $\mathbf{B}_j$ and $\mathbf{C}_j$ are collinear, so they are not independent. Since the direction of the line $\mathbf{L}_{\mathbf{A}_j\mathbf{B}_j\mathbf{C}_j}$ can be expressed as:

$$\mathbf{n}_j = \frac{\mathbf{C}_j - \mathbf{A}_j}{\|\mathbf{C}_j - \mathbf{A}_j\|} \equiv \begin{bmatrix} \sin\phi_j \cos\theta_j \\ \sin\phi_j \sin\theta_j \\ \cos\phi_j \end{bmatrix}, \quad (16)$$

the point $\mathbf{B}_j$ and $\mathbf{C}_j$ are given by

$$\mathbf{B}_j = \mathbf{A}_j + \|\mathbf{A}_j - \mathbf{B}_j\|\mathbf{n}_j = \mathbf{A}_j + (d_1 - d_2)\mathbf{n}_j \quad (17)$$

and

$$\mathbf{C}_j = \mathbf{A}_j + \|\mathbf{A}_j - \mathbf{C}_j\|\mathbf{n}_j = \mathbf{A}_j + d_1\mathbf{n}_j. \quad (18)$$

Let $\hat{\mathbf{a}}_{ij}$ ($\hat{\mathbf{b}}_{ij}$, $\hat{\mathbf{c}}_{ij}$) be the reprojection of $\mathbf{A}_j$ ($\mathbf{B}_j$, $\mathbf{C}_j$). Then the minimization problem (15) can be rewritten as:

$$\min_{\hat{\mathbf{P}}_i, \hat{\mathbf{A}}_j, \hat{\phi}_j, \hat{\theta}_j} \sum_{i=0}^{m} \sum_{j=1}^{n} (\|\mathbf{a}_{ij} - \hat{\mathbf{a}}_{ij}\|^2 + \|\mathbf{b}_{ij} - \hat{\mathbf{b}}_{ij}\|^2$$
$$+ \|\mathbf{c}_{ij} - \hat{\mathbf{c}}_{ij}\|^2) \quad (19)$$

Regarding the metric reconstruction in the above section as an initial value, the nonlinear minimization can be done using the Levenberg-Marquardt algorithm [9].

4. Experiments

The proposed algorithm has been tested on computer simulated data and real image data.
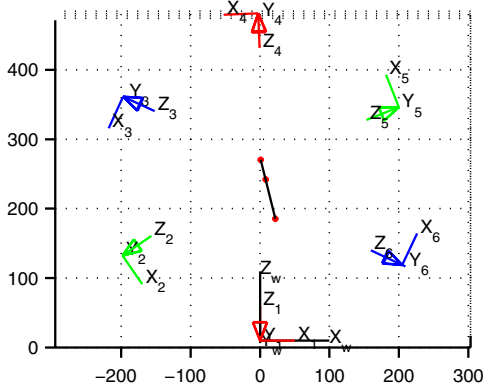
Figure 2. The planform of synthetic experimental setup in one trial.

camera	1st	2nd	3rd	4th	5th	6th
α	1200	1000	1050	1100	1100	1200
β	1000	1200	1050	1000	1100	1050
γ	0	1.0	2.0	0	-1.0	-2.0

Table 1. Intrinsic parameters of six simulated cameras.

4.1. Simulated data

We perform a lot of simulations with two cameras, three cameras, six cameras and so on. Due to the limitation of space, we only report results of simulation with six cameras.

In the simulation, the six simulated cameras' intrinsic parameters are shown in Table 1. All the principal points are [512, 384] and the image resolutions are of 1024×768. The first camera locates at one vertex of an regular hexagon whose side is of 250. And the optical axis coincides with the line joined the vertex to the common center of the regular hexagon. Each of the other five cameras is inside a cube of 20, and the cube's center is at one of the other five vertexes of the regular hexagon. And for each of the other five cameras, the angle between the optical axis and the line joined the corresponding vertex to the common center of the regular hexagon is of a random value between -5 and +5 degrees. The length of the simulated line-segment (ABC) is 90, and point B is the trisection point of (ABC) and $\|\mathbf{AB}\| = 30$. So we have $d_1 = 90$ and $d_2 = 60$. Let the line-segment (ABC) undertake 20 times of general motions inside the cube of 120 whose center coincides with the common center of the regular hexagon, and insure the image of line-segment (ABC) inside 1024×768. Figure 2 shows the planform of cameras' setup and 1D object's position in one trial.

Add Gaussian noise with mean 0 and standard derivations σ to image points. Here we use two methods, the linear 1D calibration algorithm and the bundle adjustment algorithm, to calibrate the camera set. The estimated camera
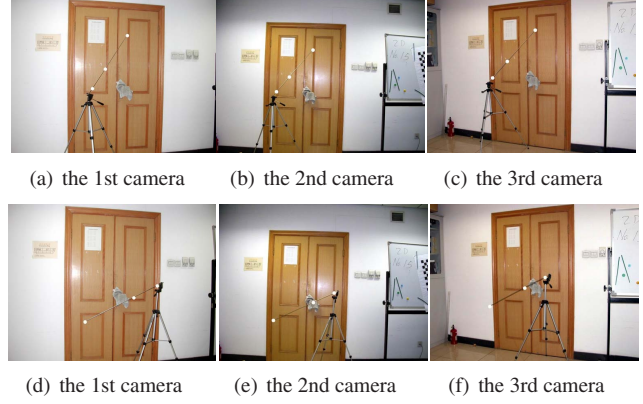

(a) the 1st camera   (b) the 2nd camera   (c) the 3rd camera

(d) the 1st camera   (e) the 2nd camera   (f) the 3rd camera

Figure 5. Sample images of camera set for 1D calibration.

(a) the 1st camera   (b) the 2nd camera   (c) the 3rd camera
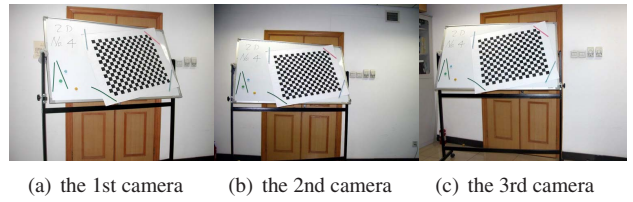
Figure 6. Sample images of camera set for 2D calibration.

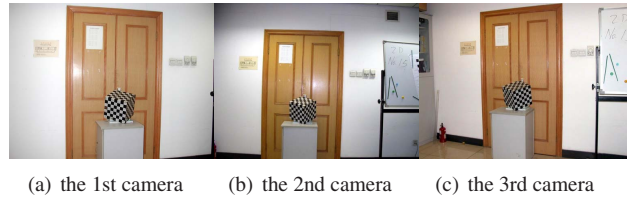(a) the 1st camera   (b) the 2nd camera   (c) the 3rd camera

Figure 7. The image triplet of 3D object for reconstruction.

parameters are compared with the ground truth, and RMS errors are measured. Vary the noise level σ from 0.2 pixels to 2.0 pixels in steps of 0.2pixels. At each noise level, 500 independent trials are performed, and the results are shown in Figure 3 and 4.

Figure 3 displays the relative errors of intrinsic parameters. Here we measure the relative errors with respect to α, as proposed by Triggs in [14]. Errors increase almost linearly with the noise level. The bundle adjustment can produce significantly better results than the linear algorithm. At the 2.0 pixels noise level, the errors for the linear algorithm are about 10%, while those for the bundle adjustment are about 2%. Figure 4(a), 4(b), 4(c), 4(d) and 4(e) display the errors of the other cameras' Euler angles relative to the first camera. Figure 4(f) displays the errors of the other cameras' translation relative to the first camera. These errors also increase almost linearly with the noise level. And the bundle adjustment can significantly refine the results of the linear algorithm.
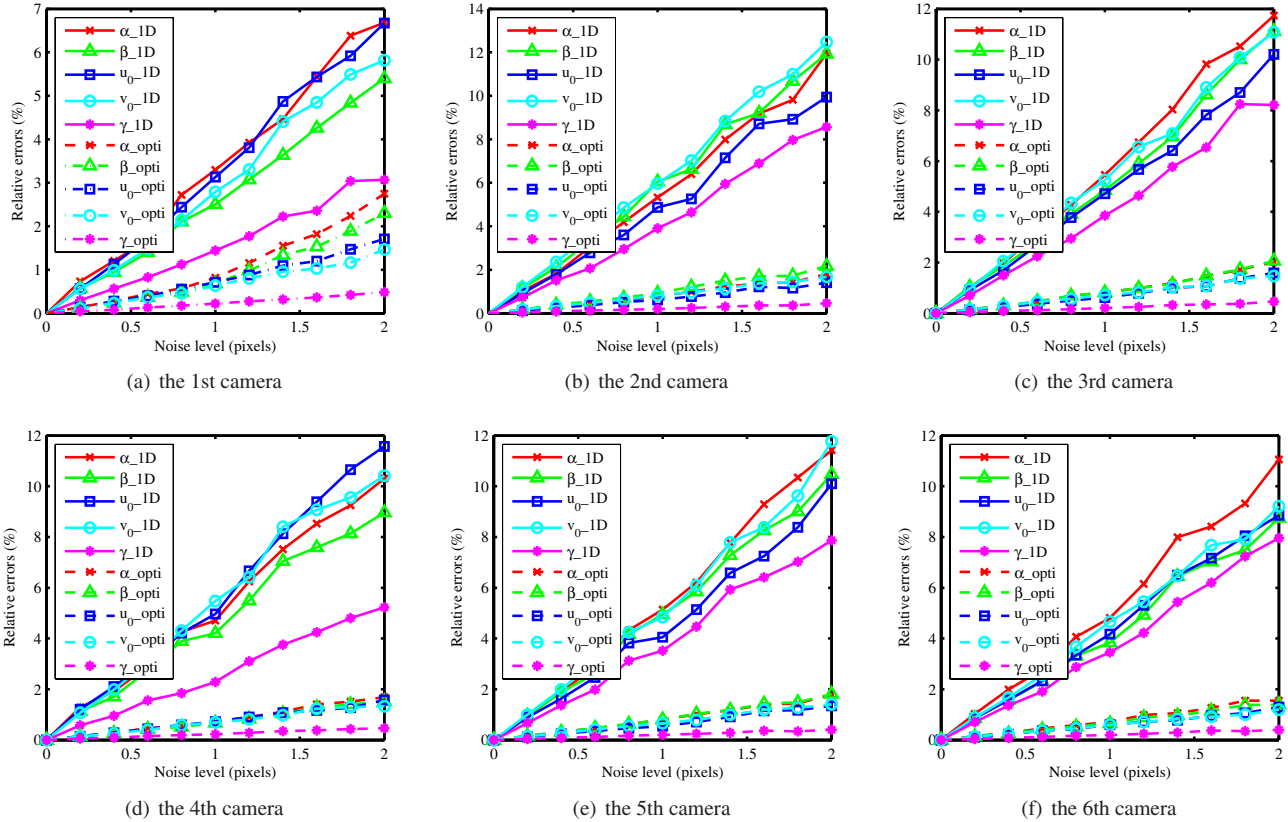
(a) the 1st camera		(b) the 2nd camera		(c) the 3rd camera	
(d) the 4th camera		(e) the 5th camera		(f) the 6th camera	

Figure 3. The relative errors of six simulated cameras' intrinsic parameters.

4.2. Real Images

For the experiment with real data, three cameras are used. The image resolution of three cameras is of 2048×1536, 1024×768 and 1024×768 respectively. We used three toy beads and strung them together with a stick. The distance between the first and the second bead is 30cm, and that of the second and the third is 60cm. Put one end of the stick on a tripod, let it undertake general rigid motion. Fifteen triplets of images are taken, two of them are shown in Figure 5. Figure 5(a), 5(b) and 5(c), corresponding to the 1st camera, the 2nd camera and the 3rd camera respectively, is a triplet corresponding to the 11th motion. Figure 5(d), 5(e) and 5(f) is another triplet corresponding to the 12th motion. Here white circle marks are used to give prominence to beads in images.

The beads are extracted from images manually. The cameras' parameters are calibrated with the proposed 1D algorithm with bundle adjustment. For comparison, we also used the 2D plane-based calibration technique described in [4] to calibrate the same cameras. Fifteen triplets of images are taken, and the 4th of them are shown in Figure 6. Figure 6(a), 6(b) and 6(c) respectively correspond to the 1st camera, the 2nd camera and the 3rd camera.

camera		α	β	u_0	v_0	γ
1	1D	2586.24	2559.96	1106.53	629.34	2.80
	2D	2569.89	2580.73	1098.45	640.36	0.20
2	1D	1170.90	1100.07	451.50	357.10	1.34
	2D	1129.35	1134.64	485.25	344.73	-3.52
3	1D	1148.45	1132.17	449.68	308.48	-1.42
	2D	1141.35	1151.63	492.50	324.72	-1.66

Table 2. Calibration results of three cameras.

method	θ_{12}	θ_{13}	θ_{23}	d	σ_d
1D	89.32	86.92	90.52	4.05	0.47
2D	88.47	87.14	88.77	4.06	0.48

Table 3. Reconstruction results with calibration results.

Three cameras' intrinsic parameters with 1D and 2D calibration methods are shown in Table 2. For each camera the first row shows the estimation from the 1D calibration algorithm, while the second row shows the result of the 2D plane-based method. We can see that there is little difference between two methods' results. We also perform reconstruction with calibration results to compare two methods.

| (a) the 2nd camera's rotation | (b) the 3rd camera's rotation | (c) the 4th camera's rotation |

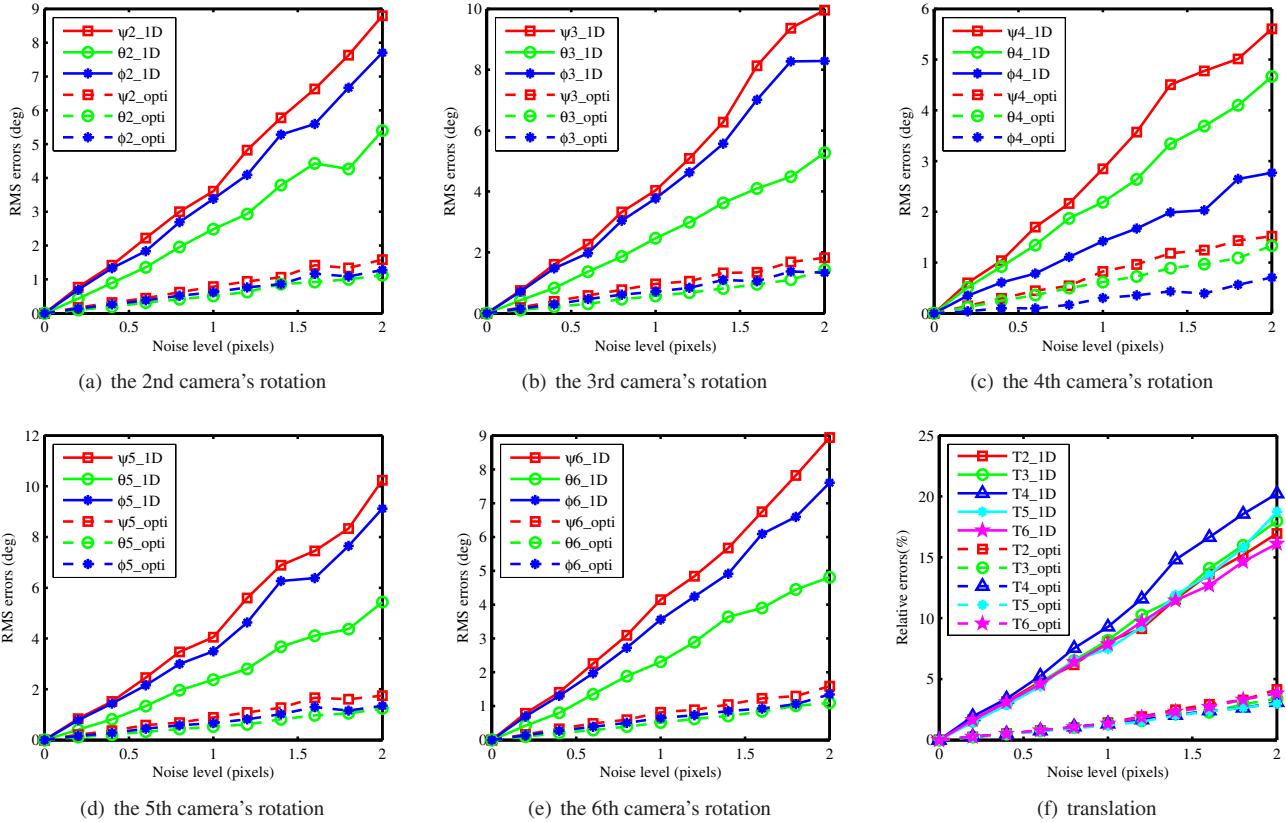| (d) the 5th camera's rotation | (e) the 6th camera's rotation | (f) translation |

Figure 4. The relative errors of six simulated cameras' extrinsic parameters.

The image triplet of 3D calibration object for performing reconstruction is shown in Figure 7. The reconstruction results and the pose relations of three cameras are shown in Figure 8. Three planes of the 3D calibration object are fitted with reconstructed 3D points, and angles between two of them, θ_{12}, θ_{13} and θ_{23} (whose ground truth is 90 degrees), are computed, as shown in Table 3. The ground truth of distance between each pair of neighbor corner points is 4.0cm. The mean $\bar{d}$ and standard deviation σ_d of these distances computed by the reconstructed 3D points are also shown in Table 3. We can see that the two methods are comparable. In some cases, the 1D calibration can outperform the 2D calibration. It may be chiefly because: 1) our 2D pattern, a printed paper on a whiteboard, is not accurate enough; 2) the 1D object can be freely placed on the interested scene volume, which can increase the calibration accuracy.

There is little difference between the reconstruction results of 1D calibration and the ground truth. The difference may come from several sources. One is the image noise and inaccuracy of the extracted data points. Another is our current rudimentary experimental setup: There was eccentricity between the holes made manually and the real axis of the bead, which inevitably results in errors of the extracted data points. Besides, the positioning of the beads was done

with a ruler with barely eye inspection.

Although there are so many error sources, the proposed 1D calibration algorithm is comparable with the 2D plane-based method. This demonstrates the applicability of proposed 1D calibration algorithm in practice.

5. Conclusions

In this paper, we have investigated the possibility of multi-camera calibration using 1D objects undertaking general rigid motion. A linear algorithm for multi-camera calibration is proposed, followed by a bundle adjustment to refine the results. Both the computer simulated and real image data have been used to test the proposed algorithm. It shows that the proposed algorithm is valid and robust.

In addition, for multiple cameras mounted apart from each other, all cameras must observe the whole calibration object simultaneously, which is a prerequisite for calibration and hard to satisfy with 3D or 2D calibration objects. This is not a problem for 1D object. Most importantly, the proposed calibration algorithm is very easier and more practical due to 1D object performing general rigid motion instead of rotation around a fixed point or planar motion and no need of one calibrated "base" camera.
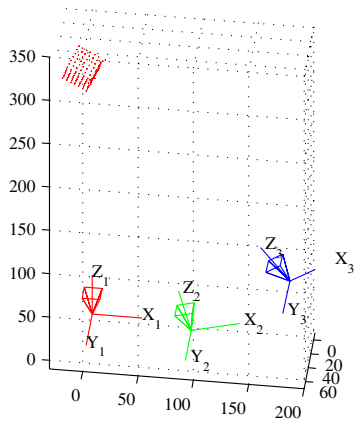
(a) with 1D calibration results
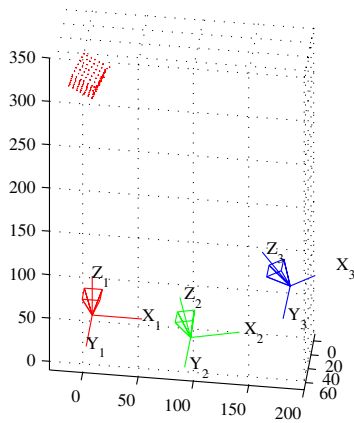
(b) with 2D calibration results

Figure 8. Reconstruction results of 3D object.

References

[1] Y. I. Abdel-Aziz and H. M. Karara. Direct linear transformation from comparator coordinates into object space coordinates, 1971. ASP Symposium on Colse-Range Photogrammetry, Virginia, USA.

[2] P. Hammarstedt, P. Sturm, and A. Heyden. Degenerate cases and closed-form solutions for camera calibration with one-dimensional objects, 2005. In Proceedings of the ICCV'05, Beijing, China. pages 317-324.

[3] R. Hartley. Estimation of relative camera positions for un-calibrated cameras, 1992. In Proceedings of the ECCV'92, Genova, Italy. pages 579-587.

[4] R. Hartley and A. Zisserman. Mmultiple view geometry in computer vision, 2000. Cambridge: Cambridge University Press.

[5] Y. Kojima, T. Fujii, and M. Tanimoto. New multiple camera calibration method for a large number of cameras, 2004. Proceedings of the SPIE, Volume 5665, pp 156-163.

[6] Q. T. Luong and O. D. Faugeras. Self-calibration of a moving camera from point correspondence and fundamental matrices. *Int. J. Comput. Vision*, 22(3):261–289, 1997.

[7] S. J. Maybank and O. D. Faugeras. A theory of self-calibration of a moving camera. *Int. J. Comput. Vision*, 8(2):123–152, 1992.

[8] X. Q. Meng, H. Li, and Z. Y. Hu. A new easy camera calibration technique based on circular points, 2000. In Proceedings of the BMVC'2000, Bristol, UK. pages 496–505.

[9] J. J. More. The levenberg-marquardt algorithm, implementation and theory, 1977. Numerical Analysis G.A. Watson, ed. Springer Verlag.

[10] M. Pollefeys, L. V. Gool, and O. A. The modulus constraint: a new constraint for self-calibration, 1996. In Proceedings of the ICPR'96, Vienna, Austria. pages 31-42.

[11] C. Rother. Multi-view reconstruction and camera recovery using a real or virtual reference plane, 2003. PhD thesis, Computational and Active Perception Laboratory, Kungl Tekniska Högskolan.

[12] P. Sturm and S. J. Maybank. On plane-based camera calibration: a general algorithm, singularities, applications, 1999. In Proceedings of the CVPR'99, Colorado, USA. pages 432–437.

[13] B. Triggs. Auto-calibration and the absolute quadric, 1997. In Proceedings of the CVPR'97, Puerto Rico, USA. pages 609-614.

[14] B. Triggs. Autocalibration from planar scenes, 1998. In Proceedings of the ECCV'98. pages 89-105.

[15] R. Tsai. An efficient and accurate camera calibration technique for 3d machine vision, 1986. In Proceedings of the CVPR'86, Miami Beach, USA. pages 364–374.

[16] J. Y. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion model and accuracy evaluation. *IEEE Trans. Pattern Anal. Mach. Intell*, 14(14):965–980, 1992.

[17] F. C. Wu, Z. Y. Hu, and H. J. Zhu. Camera calibration with moving one-dimensional objects. *Pattern Recognition*, 38(5):755–765, 2005.

[18] Z. Y. Zhang. Flexible camera calibration by viewing a plane from unknown orientations, 1999. In Proceedings of the ICCV'99, Kerkya, Greece. pages 666–673.

[19] Z. Y. Zhang. Camera calibration with one-dimensional objects. *IEEE Trans. Pattern Anal. Mach. Intell*, 26(7):892–899, 2004.