

once per second which is acceptable for welding process control although further development of hardware and algorithm optimization would significantly improve the speed. Hence, the proposed algorithm appears to have the potential to be used in online control of metal transfer process.

## V. CONCLUSION

- The bilinear interpolation is effective for image enhancement toward better edge detection.
- The proposed brightness-based selection and edge-based separation algorithm can detect droplets from the image and try to detect adequate edge information from interpolated images.
- The proposed model for droplet edge gives an effective method to estimate the size of the droplet robustly and accurately and the proposed model validation assures that the model used meets a minimal accuracy requirement.
- The speed of the image processing appears to meet the minimal requirement for real-time control.

It should be mentioned that certain algorithm parameters and equations are ad hoc for the well-defined and constrained problem under investigation. In case welding conditions change, appropriate modifications may become necessary.

## REFERENCES

- [1] Y. M. Zhang and P. J. Li, "Modified active control of metal transfer and pulsed GMAW of titanium," *Welding J.*, vol. 80, pp. 54S–61S, 2001.
- [2] L. A. Jones, T. W. Eagar, and J. H. Lang, "A dynamic model of drops detaching from a gas metal arc welding electrode," *J. Phys. D-Appl. Phys.*, vol. 31, pp. 107–123, 1998.
- [3] G. Wang, P. G. Huang, and Y. M. Zhang, "Numerical analysis of metal transfer in gas metal arc welding under modified pulsed current conditions," *Metallurgical and Materials Trans. B-Process Metallurgy and Materials Processing Science*, vol. 35, pp. 857–866, 2004.
- [4] S. Chakraborty, "Analytical investigations on breakup of viscous liquid droplets on surface tension modulation during welding metal transfer," *Appl. Phys. Lett.*, vol. 86, pp. 1–3, 2005.
- [5] C. S. Wu, M. A. Chen, and Y. F. Lu, "Effect of current waveforms on metal transfer in pulsed gas metal arc welding," *Measure. Sci. Technol.*, vol. 16, pp. 2459–2465, 2005.
- [6] B. Y. B. Yudodibroto, M. J. M. Hermans, and Y. Hirata *et al.*, "Pendant droplet oscillation during GMAW," *Sci. Technol. Welding*, vol. 11, pp. 308–314, 2006.
- [7] S. Rhee and A. Kannatey, "Observation of metal transfer during gas metal arc-welding," *Welding J.*, vol. 71, pp. 381S–386S, 1992.
- [8] K. H. Li, J. S. Chen, and Y. M. Zhang, "Double-electrode GMAW process and control," *Welding J.*, vol. 86, no. 8, pp. 231S–237S, 2007.
- [9] Z. Wang, K. R. Rao, and J. Ben-Arie, "Optimal ramp edge detection using expansion matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 11, pp. 1092–1097, 1996.
- [10] N. Saito and M. A. Cunningham, "Generalized e-filter and its application to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 8, pp. 814–817, 1990.
- [11] R. J. Qian and T. S. Huang, "Optimal edge detection in two-dimensional images," *IEEE Trans. Image Process.*, vol. 5, no. 7, pp. 1215–1220, 1996.
- [12] M. Petrou and J. Kittler, "Optimal edge detector for ramp edges," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 5, pp. 483–491, 1991.
- [13] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall.
- [14] F. Luo, S. J. Wu, L. C. Jiao, and L. R. Zhang, "Implementation of de-noise DWT chip based on adaptive soft-threshold," in *Proc. Int. Conf., Signal Process.*, 2000, vol. 1, pp. 614–618.
- [15] M. Tommiska, M. Loukola, and T. Koskivirta, "An FPGA-based simulation and implementation of AAL type 2 receiver," *J. Commun. Netw.*, pp. 63–67, 1999.

## An Efficient Approach for Dynamic Calibration of Multiple Cameras

I.-Hsien Chen and Sheng-Jyh Wang

**Abstract**—In this paper, we propose a new algorithm for dynamic calibration of multiple cameras. Based on the mapping between a horizontal plane in the 3-D space and the 2-D image plane on a panned and tilted camera, we utilize the displacement of feature points and the epipolar-plane constraint among multiple cameras to infer the changes of pan and tilt angles for each camera. This algorithm does not require a complicated correspondence of feature points. It can be applied to surveillance systems with wide-range coverage. It also allows the presence of moving objects in the captured scenes while performing dynamic calibration. The sensitivity analysis of our algorithm with respect to measurement errors and fluctuations in previous estimations is also discussed. The efficiency and feasibility of this approach has been demonstrated in some experiments over real scenery.

**Note to Practitioners**—For a surveillance system with multiple cameras, the poses of cameras may be changed from time to time to acquire different views of the monitored scene. Whenever the poses of cameras are changed, the relative positioning and orientation among cameras need to be recalibrated. In this paper, we demonstrate a new and efficient approach to calibrate multiple cameras dynamically. The concept of our approach originated from the observation that people usually can identify the directions of the pan and tilt angles, and even make a rough estimate about the changes of pan and tilt angles, simply based on some clues revealed in the captured images.

In our approach, a set of cameras are first calibrated based on a static calibration method. As cameras begin to pan or tilt, the images of these cameras change accordingly. We keep extracting and tracking a few feature points from these images. Based on the displacement of these feature points in consecutive images, the pan and tilt angle changes of the cameras can be automatically estimated via the proposed approach. There is no need to place calibration patterns or landmarks while performing dynamic calibration. In addition, there is no need to perform the complicated correspondence of feature points among cameras. The proposed approach is practical for a wide-range surveillance system with multiple cameras and is applicable for complicated environments. In the future, we will combine the proposed approach with an object tracking system to develop an efficient active surveillance system with multiple cameras.

**Index Terms**—Dynamic camera calibration, multiple cameras.

## I. INTRODUCTION

For a surveillance system with multiple cameras, cameras may pan or tilt from time to time to acquire different views of the monitored scene. However, when a camera pans or tilts, its extrinsic parameters change accordingly. For this type of surveillance systems, how to accurately and efficiently recalibrate the extrinsic parameters of multiple cameras has become an important issue.

Up to now, various kinds of approaches have been developed to calibrate static camera's intrinsic and extrinsic parameters, such as the

Manuscript received April 23, 2007; revised April 23, 2007. First published June 10, 2008; current version published December 30, 2008. This paper was recommended for publication by Associate Editor Y. F. Li and Editor M. Wang upon evaluation of the reviewers' comments. This work was supported in part by the Ministry of Economic Affairs of the Republic of China under Grant 96-EC-17-A-02-S1-032.

The authors are with the Department of Electronics Engineering, Institute of Electronics, National Chiao Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: kace.ee89g@nctu.edu.tw; shengjih@cc.nctu.edu.tw).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TASE.2008.918128

techniques proposed in [1]–[6]. Nevertheless, it is impractical to repeatedly perform these elaborate calibration processes over a camera when the camera is under panning or tilting all the time. On the other hand, [7]–[9] have proposed plane-based calibration methods specially designed for the calibration of multiple cameras. However, for a wide-range surveillance system with multiple active cameras, these planar calibration objects may not be properly observed by all cameras when cameras are under movement.

For dynamic camera calibration, [10] proposed an offline method, where they tried to find the relationship between the realized rotation angle and the requested angle. In [11], the pose of a calibrated camera was estimated from a planar target. However, both [10] and [11] only demonstrated the dynamic calibration of a single camera, but not the calibration among multiple cameras. In [12], the authors utilized the marks and width of parallel lanes to calibrate PTZ cameras. Although this method is practical for traffic monitoring, it is not general enough for other types of surveillance systems. In [13], a dynamic camera calibration with narrow-range coverage was proposed. For a pair of cameras, this method performed the correspondence of feature points on the image pair and used coplanar geometry for camera calibration. In [14], the relative pose between a calibrated camera and a projector was determined via plane-based homography. This approach requires the correspondence of feature points. However, for surveillance systems with wide-range coverage, the matching of feature points is usually a difficult problem.

In this paper, we propose a new algorithm for dynamic calibration of multiple cameras. This algorithm does not require a complicated correspondence of feature points. Our algorithm also allows the presence of moving objects in the captured scenes, while performing dynamic calibration. As cameras begin to pan or tilt, we keep extracting and tracking feature points based on the Kanade–Lucas–Tomasi (KLT) algorithm [15]. Next, we utilize the displacement of feature points and the epipolar-plane constraint among multiple cameras to infer the changes of pan and tilt angles for each camera. Compared with [13], we only need the correspondence of epipolar lines but not the exact matching of feature points. The use of epipolar lines greatly simplifies the correspondence process and makes our approach suitable for complicated surveillance environments.

This paper is organized as follows. First, in Section II, we explain how we utilize the displacement of feature points and the epipolar-plane constraint to infer the changes of pan angle and tilt angle. We also describe how to filter out undesired feature points when moving objects are present. The sensitivity analysis with respect to measurement errors and the fluctuations of previous estimations will be addressed in Section III. Some experimental results over real data are demonstrated in Section IV. Finally, in Section V, the conclusion is drawn.

## II. DYNAMIC CALIBRATION OF MULTIPLE CAMERAS

In this section, we explain how we perform dynamic calibration process based on temporal and 3-D spatial information. We will first introduce how to calibrate a dynamic camera based on the displacement of feature points in the temporal domain. After that, we will apply the epipolar-plane constraint over each pair of cameras to obtain more robust calibration. Moreover, since people or moving objects may enter or leave the scene while cameras are capturing images, we need to filter out their interferences to avoid the degradation of calibration accuracy. In Fig. 1, we show an overall picture of the proposed dynamic calibration algorithm.

### A. Dynamic Calibration of a Single Camera

The basic camera model and related formulae can be found in our previous work [17]. When a camera with a rotation radius  $r$  has a tilt angle  $\phi$  and a pan angle  $\theta$ , we may deduce (1), to express the back projection function  $B(p, \theta, \phi, h, \Omega)$  from the image coordinates  $p =$

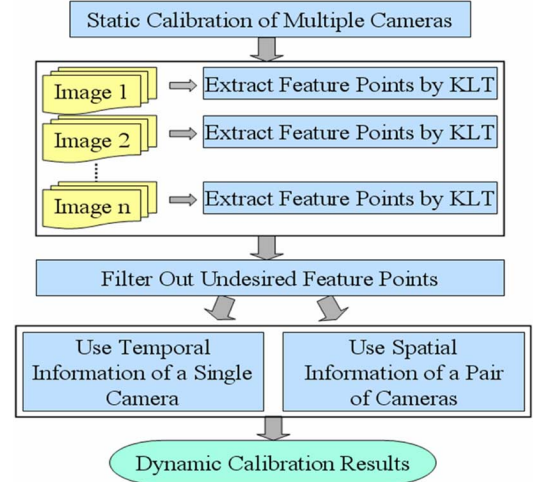


Fig. 1. Flowchart of the proposed dynamic calibration algorithm.

$(x, y)$  onto a 3-D point  $(X, Y, Z)$  lying on a horizontal plane with  $Y = -h$  [17]

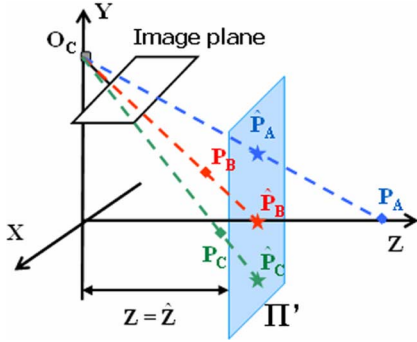
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{\beta x C_\theta (r S_\phi - h) + \alpha y S_\theta (r - h S_\phi) - \alpha \beta h C_\phi S_\theta}{\alpha (y C_\phi - \beta S_\phi)} \\ -h \\ \frac{\beta x S_\theta (h - r S_\phi) + \alpha y C_\theta (r - h S_\phi) - \alpha \beta h C_\phi C_\theta}{\alpha (y C_\phi - \beta S_\phi)} - r \end{bmatrix} \equiv B(p, \theta, \phi, h, \Omega). \quad (1)$$

Here,  $C_\theta$ ,  $S_\theta$ ,  $C_\phi$ , and  $S_\phi$  represent  $\cos(\theta)$ ,  $\sin(\theta)$ ,  $\cos(\phi)$  and  $\sin(\phi)$ , respectively. Additionally,  $\Omega$  represents the set of intrinsic parameters of the camera.

Assume we have a set of cameras. At the beginning, we calibrate the 3-D pose of each camera via the static calibration method proposed in [17]. As a camera starts to pan or tilt, its image content changes. To recalibrate the new pose of the camera, we check the temporal displacement of a few feature points in the image. Here, we use the KLT method [15] to extract and track feature points in consecutive images. We also assume all extracted feature points correspond to some unknown static points in the 3-D space.

Typically, we may assume the rotation radius  $r$  is far smaller than the distances between these 3-D points and the camera. We also assume the changes of pan angle and tilt angle are very small during the capturing of two successive images. With these two assumptions, the projection center of the camera can be thought to be fixed with respect to the 3-D points, while the camera is panning or tilting. In other words, the projection lines, which connect the projection center to each of these observed 3-D points, are fixed in the 3-D space, as long as these 3-D points stay static during the capture of images. By using these projection lines as a reference, we may recalibrate the new pose of the camera. Moreover, as illustrated in Fig. 2, if three 3-D points,  $\mathbf{P}_A$ ,  $\mathbf{P}_B$ , and  $\mathbf{P}_C$ , are replaced by another three points,  $\hat{\mathbf{P}}_A$ ,  $\hat{\mathbf{P}}_B$ , and  $\hat{\mathbf{P}}_C$  on their projection lines, there is no influence on the projected points on the image plane. Hence, even if we do not actually know the real locations of these 3-D points, we may simply back project all feature points in the 2-D image onto a 3-D pseudo plane with a constant  $Z$  coordinate, as shown in Fig. 2.

In our approach, based on a few feature points on a pair of successive images  $I_{t-1}$  and  $I_t$ , we first back project these feature points in  $I_{t-1}$  onto a 3-D pseudo plane with a constant  $Z$ . Then, we try to find a new pose of the camera that can map the corresponding feature points in  $I_t$  onto the same 3-D pseudo points. That is, if we assume the camera has the pan angle  $\theta_{t-1}$  and the tilt angle  $\phi_{t-1}$ , while capturing  $I_{t-1}$ , and has the pan angle  $\theta_{t-1} + \Delta\theta_t$  and the tilt angle  $\phi_{t-1} + \Delta\phi_t$ , while


 Fig. 2. Illustration of a pseudoplane  $\Pi'$ .

capturing  $I_t$ , we try to find  $\Delta\theta_t$  and  $\Delta\phi_t$  that minimize the following formula:

$$D = \sum_{k=1}^K \|\hat{B}(\hat{p}_k, \theta_{t-1} + \Delta\theta_t, \phi_{t-1} + \Delta\phi_t) - \hat{B}(p_k, \theta_{t-1}, \phi_{t-1})\|^2. \quad (2)$$

In (2),  $\hat{B}$  represents the back projection function of an image feature point onto a pseudo 3-D plane  $\Pi'$ . Here, we especially use “hat” to denote that the back-projection is restricted to a vertical pseudo plane  $\Pi'$ . Besides,  $p_k$  denotes a feature point in  $I_{t-1}$  and  $\hat{p}_k$  denotes the same feature point in  $I_t$ .  $K$  is the total number of image feature points for calibration. Note that in (2), we ignore the altitude parameter  $h$  of these back-projected points. This is because the altitude  $h$  can be deduced from (1) once if the  $Z$  coordinate is fixed. We also ignore the intrinsic parameters  $\Omega$  since they are not changed when the camera pans and tilts.

### B. Dynamic Calibration of Multiple Cameras Based on Epipolar-Plane Constraint

In the previous section, we assume the projection center of a single camera is fixed during panning and tilting. The projection lines are then used as a reference to calibrate the new pose of that camera. To further increase the accuracy of calibration, we add on the 3-D spatial relationship among cameras.

In Fig. 3, we show the epipolar geometry for a pair of cameras [16]. For these two cameras, their projection centers,  $O_{C1}$  and  $O_{C2}$ , together with a 3-D point  $P_A$ , determine an epipolar plane  $\Pi$ . This epipolar plane  $\Pi$  intersects the image planes of the cameras to form two epipolar lines  $l_1$  and  $l_2$ . If  $p_A^1$  and  $p_A^2$  are the projected points of  $P_A$  on the image planes, they must lie on  $l_1$  and  $l_2$ , respectively. This epipolar constraint implies that  $O_{C1}$ ,  $O_{C2}$ ,  $p_A^1$ , and  $p_A^2$  are coplanar and the epipolar plane  $\Pi$  can be expressed as

$$\begin{aligned} & \pi(O_{C1}, O_{C2}, p_A^1, \theta^1, \phi^1) \\ & \equiv \overline{O_{C2}O_{C1}} \times \overline{O_{C1}B(p_A^1, \theta^1, \phi^1)} \\ \text{or } & \pi(O_{C1}, O_{C2}, p_A^2, \theta^2, \phi^2) \\ & \equiv \overline{O_{C1}O_{C2}} \times \overline{O_{C2}B(p_A^2, \theta^2, \phi^2)}. \end{aligned} \quad (3)$$

In (3), we use the  $B(\cdot)$  function defined in (1). Note that we ignore the altitude parameter  $h$  because the formation of epipolar plane is actually independent of  $h$ . That is, no matter what value  $h$  is, the epipolar plane is still the same.

As illustrated in Fig. 3, we assume a pair of cameras has initially been calibrated via some kind of calibration algorithm. We assume a few features, like  $p_A^1$ ,  $p_B^1$ ,  $p_C^1$ ,  $p_A^2$ ,  $p_D^2$ , and  $p_E^2$ , are located on a pair

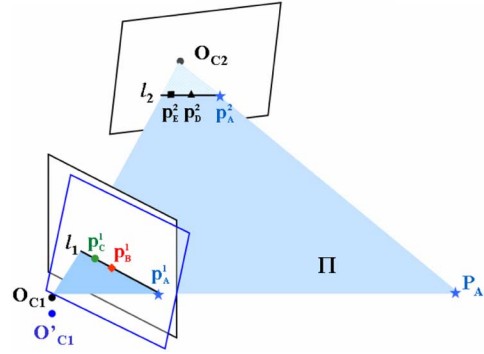


Fig. 3. Illustration of epipolar-plane constraint.

of corresponding epipolar lines. Without performing pointwise correspondence, we do not actually know where these feature points are projected from. However, we are still confident of the fact that these 3-D points must be “somewhere” on the epipolar plane. As long as these 3-D points remain static in the 3-D space, this epipolar plane is fixed. Hence, the epipolar planes that have been identified at the previous moment can be used as a reference for the calibration of cameras at the current moment.

In Fig. 3, we assume a pair of cameras has been calibrated at the time instant  $t-1$  and an epipolar plane  $\Pi$  has been identified. Assume at that time instant  $t-1$ , the pan and tilt angles of Camera-1 are  $\theta_{t-1}^1$  and  $\phi_{t-1}^1$ , while the pan and tilt angles of Camera-2 are  $\theta_{t-1}^2$  and  $\phi_{t-1}^2$ . Camera-1 captures the image  $I_{t-1}^1$ , while Camera-2 captures  $I_{t-1}^2$ . On the other hand, at the time instant  $t$ , Camera-1 rotates to a new pan angle ( $\theta_{t-1}^1 + \Delta\theta_t^1$ ) and a new tilt angle ( $\phi_{t-1}^1 + \Delta\phi_t^1$ ), while Camera-2 rotates to ( $\theta_{t-1}^2 + \Delta\theta_t^2$ ) and ( $\phi_{t-1}^2 + \Delta\phi_t^2$ ). Here, we only discuss the calibration of Camera-1. The calibration of Camera-2 can be implemented in a similar way.

For Camera-1, assume a prominent feature point  $p_A^1$  has been extracted from  $I_{t-1}^1$ . This feature moves to  $\hat{p}_A^1$  in  $I_t^1$ . Based on  $p_A^1$ ,  $\theta_{t-1}^1$ , and  $\phi_{t-1}^1$ , we may form an epipolar plane  $\Pi$ . At the time instant  $t$ , we then seek to find the angles ( $\theta_{t-1}^1 + \Delta\theta_t^1$ ) and ( $\phi_{t-1}^1 + \Delta\phi_t^1$ ) such that  $\hat{p}_A^1$  is still located on the same epipolar plane. That is, we seek to find  $\Delta\theta_t^1$  and  $\Delta\phi_t^1$  such that

$$B(\hat{p}_A^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_A^1, \theta_{t-1}^1, \phi_{t-1}^1) = 0. \quad (4)$$

Similarly, for  $p_B^1$  and  $p_C^1$  that share the same epipolar line with  $p_A^1$ , we have

$$B(\hat{p}_B^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_A^1, \theta_{t-1}^1, \phi_{t-1}^1)$$

and

$$B(\hat{p}_C^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_A^1, \theta_{t-1}^1, \phi_{t-1}^1). \quad (5)$$

Note that in (4) and (5), the projection center  $O_{C2}$  may have a slight movement when Camera-2 rotates. That movement can be taken into account to achieve more accurate calibration. Here, we simply ignore that part to simplify the formulation.

For Camera-1, assume we have extracted  $m$  epipolar lines. Moreover, on the  $j$ th epipolar line, where  $j = 1, 2, \dots, m$ , we have extracted  $n_j$  feature points  $\{p_{j,1}^1, p_{j,2}^1, \dots, p_{j,n_j}^1\}$  on  $I_{t-1}^1$ . These  $n_j$  feature points move to  $\{\hat{p}_{j,1}^1, \hat{p}_{j,2}^1, \dots, \hat{p}_{j,n_j}^1\}$  on  $I_t^1$ . Besides, we assume  $p_j^1$  denotes one of the feature points in  $\{p_{j,1}^1, p_{j,2}^1, \dots, p_{j,n_j}^1\}$ .

Based on the epipolar-plane constraint, we can estimate  $\Delta\theta_t^1$  and  $\Delta\phi_t^1$  that minimize

$$G_t^1 = \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| B(\hat{p}_{j,i}^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_j^1, \theta_{t-1}^1, \phi_{t-1}^1) \right\|^2. \quad (6)$$

Furthermore, by integrating (2) and (6), the changes of pan angle and tilt angle of Camera-1 can be estimated by minimizing the following formula:

$$F_t^1 = \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| \hat{B}(\hat{p}_{j,i}^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) - \hat{B}(p_{j,i}^1, \theta_{t-1}^1, \phi_{t-1}^1) \right\|^2 + \lambda \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| B(\hat{p}_{j,i}^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_j^1, \theta_{t-1}^1, \phi_{t-1}^1) \right\|^2. \quad (7)$$

Similarly, the changes of pan angle and tilt angle of Camera-2 can be estimated by minimizing

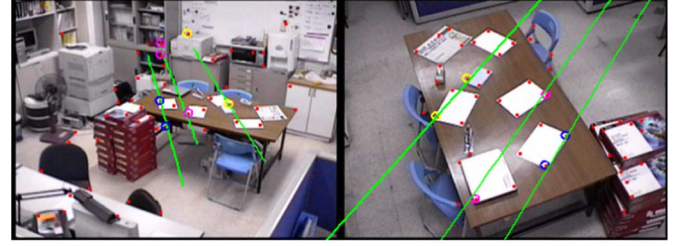
$$F_t^2 = \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| \hat{B}(\hat{p}_{j,i}^2, \theta_{t-1}^2 + \Delta\theta_t^2, \phi_{t-1}^2 + \Delta\phi_t^2) - \hat{B}(p_{j,i}^2, \theta_{t-1}^2, \phi_{t-1}^2) \right\|^2 + \lambda \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| B(\hat{p}_{j,i}^2, \theta_{t-1}^2 + \Delta\theta_t^2, \phi_{t-1}^2 + \Delta\phi_t^2) \cdot \pi(O_{C1}, O_{C2}, p_j^2, \theta_{t-1}^2, \phi_{t-1}^2) \right\|^2. \quad (8)$$

Here,  $\lambda$  is a parameter to weight the contributions of temporal clues and 3-D spatial clues. In our experiments, we simply set  $\lambda = 1$ . In theory, for each camera, one feature point is sufficient for the first right term of (7) or (8) to solve  $\Delta\theta_t$  and  $\Delta\phi_t$ . Whenever a pair of epipolar lines can be determined, any feature point on the epipolar lines can be used for the second right term of (7) or (8) to make the estimation more robust.

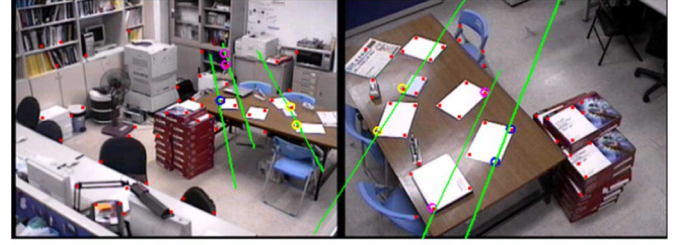
To deduce  $\Delta\theta_t^1$ ,  $\Delta\phi_t^1$ ,  $\Delta\theta_t^2$ , and  $\Delta\phi_t^2$ , we adopt the Levenberg–Marquardt (LM) algorithm. In our experiments, the initial guesses of pan/tilt angle changes are set to be  $0^\circ$ . Note that for a pair of corresponding epipolar lines, Camera-1 and Camera-2 may have very different numbers of feature points. That is, the  $n_j$  in (7) may be different from the  $n_j$  in (8). This is because we do not actually seek to perform the correspondence of feature points. Instead, we seek for a consistent matching of epipolar lines between  $I_{t-1}$  and  $I_t$ . This strategy greatly simplifies the correspondence problem. Moreover, formulae (7) and (8) can also be merged together into a single formula in the optimization process.

In summary, for the proposed dynamic calibration algorithm, we perform the following steps.

- Step 1) We perform static camera calibration based on the method proposed in [17]. After that, cameras are allowed to pan and tilt freely.
- Step 2) On each image, a few feature points are extracted and tracked based on the KLT algorithm [15]. Feature points moving out of the image are removed, while new feature points entering the image are added.
- Step 3) For each pair of cameras, based on the previous calibration results, we generate pairs of epipolar lines that pass through these extracted feature points. Actually, as long as a feature point is within a predefined distance from an epipolar line, we say that the feature point is passed through by the epipolar line. In our experiments, the predefined distance is set to be 3 pixels.



(a)



(b)

Fig. 4. Image pairs captured at two different time instants. Green lines indicate three pairs of corresponding epipolar lines.



(a)

(b)

Fig. 5. (a) Image captured by a camera with  $55.1^\circ$  tilt angle. (b) Image captured by a camera with  $54.6^\circ$  tilt angle. Red crosses represent feature points extracted by the KLT algorithm.

- Step 4) Based on the extracted feature points and the information of epipolar lines, we calibrate the new pan angle and tilt angle for each pair of cameras by minimizing (7) and (8). After that, go back to Step 2.

The above procedure is repeated to acquire the new poses of all cameras. In Fig. 4(a) and (b), we show images captured by two different cameras at two different time instants, overlapped by three pairs of epipolar lines. Note that even though the feature points on these epipolar lines come from different 3-D points, we may still be able to achieve reliable dynamic calibration based on the matching of epipolar lines.

### C. Dynamic Calibration With Presence Of Moving Objects

So far, we have assumed all the feature points used for calibration correspond to some fixed 3-D points in the scene. However, in real applications, such as object tracking or 3-D positioning, some moving objects may be present. To guarantee accurate calibration, we need to get rid of these feature points related to moving objects.

In Fig. 5, we show two successive image frames where the camera tilts up by  $0.5^\circ$ . For each feature point, we calculate its spatial displacement  $(dx, dy)$ . The distribution of  $(dx, dy)$  is plotted in Fig. 6, where most displacements cluster around  $(0, -4)$ . These clustered displacements correspond to the movements of static feature points caused by camera rotation. On the other hand, there exist some outlier displacements which correspond to the movement of feature points lying on the moving person.

However, the displacement of feature points depends not only on the pose of camera but also on the contents inside the 3-D scene. Theoretically,

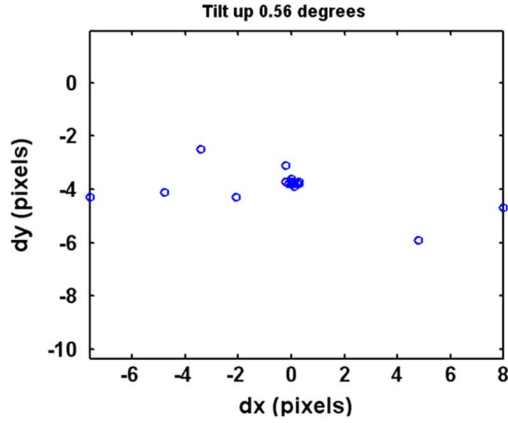


Fig. 6. The distribution of spatial displacement for the extracted feature points in Fig. 5.

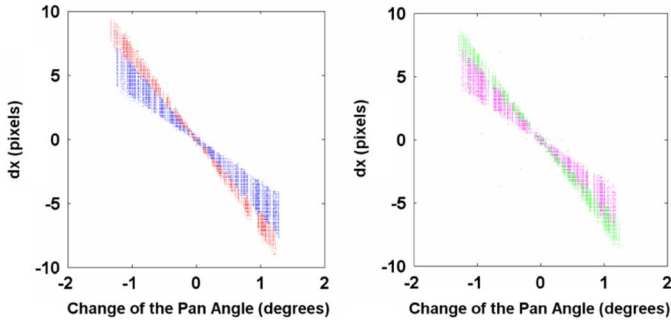


Fig. 7. The  $x$ -component displacement of feature points with respect to the changes of pan angle for four different cameras, without the presence of moving objects. The relationships for Camera-1, Camera-2, Camera-3, and Camera-4 are plotted in red, blue, green, and magenta, respectively.

ically, by taking the partial derivative of the image coordinates with respect to the pan angle  $\theta$ , we have

$$\begin{aligned} \frac{\partial x}{\partial \theta} &= -\frac{\alpha C_\phi (XC_\theta - ZS_\theta - rS_\theta)^2}{(XC_\phi S_\theta - YS_\phi + ZC_\phi C_\theta + r(C_\phi C_\theta - 1))^2} \\ &\quad - \frac{\alpha (XS_\theta + ZC_\theta + rC_\theta)}{XC_\phi S_\theta - YS_\phi + ZC_\phi C_\theta + r(C_\phi C_\theta - 1)} \\ \frac{\partial y}{\partial \theta} &= -\beta (XC_\phi C_\theta - ZC_\phi S_\theta - rC_\phi S_\theta) \\ &\quad + \frac{(XS_\phi S_\theta + YC_\phi + ZS_\phi C_\theta + rS_\phi C_\theta)}{(XC_\phi S_\theta - YS_\phi + ZC_\phi C_\theta + r(C_\phi C_\theta - 1))^2} \\ &\quad + \frac{\beta (XS_\phi C_\theta - ZS_\phi S_\theta - rS_\phi S_\theta)}{XC_\phi S_\theta - YS_\phi + ZC_\phi C_\theta + r(C_\phi C_\theta - 1)} \end{aligned} \quad (9)$$

which indicates how the location of a feature point varies with respect to the change of pan angle. To simplify the formula, we assume  $\phi = 0$  to ignore the influence of tilt angle. The simplified formula is expressed in (10), shown at the bottom of the page. Similarly, by ignoring the effect of pan angle, (11), shown at the bottom of the page, indicates how

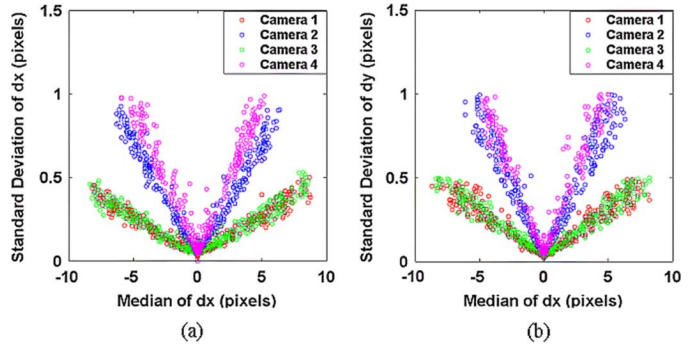


Fig. 8. (a) Standard deviation of  $dx$  with respect to the median of  $dx$  when cameras are under panning. (b) Standard deviation of  $dy$  with respect to the median of  $dx$  when cameras are under panning.

the location of a feature point varies with respect to the change of tilt angle. Both (10) and (11) indicate the crucial role of the 3-D location  $(X, Y, Z)$  in the displacement of feature points. Hence, for different scenes, we expect different degrees of feature point displacements. On the other hand, if the same scene observed by the same camera but with two different pan-angle changes, not only the displacement magnitudes but also the distributions of displacement are different. The distribution with a smaller pan angle change is more compact.

Since the distribution of the displacement highly depends on the observed scene and the magnitude of angle change, we obtain the characteristics of displacement via a learning process for each camera. In the learning stage, we intentionally pan and tilt each camera to capture a sequence of images, without the presence of moving objects. In our experiments, four cameras are used and Fig. 9 shows an example of images captured by these four cameras. In Fig. 7, we show the  $x$ -component displacement of feature points with respect to the change of pan angle for each of our four cameras. It can be observed that Camera-1 and Camera-3 have roughly the same statistical behaviors, while Camera-2 and Camera-4 have similar behaviors. In Fig. 8(a), we further plot the relationship between the standard deviation of  $dx$  and the median of  $dx$  when cameras are under panning. Again, Camera-1 and Camera-3 have roughly the same statistical behaviors, while Camera-2 and Camera-4 have similar behaviors. Even though different cameras may have very different statistical behaviors, the relationship between the standard deviations of  $dx$  and the median of  $dx$  is roughly linear for each camera. Similarly, Fig. 8(b) shows the statistical relationship between the standard deviation of  $dy$  and the median of  $dx$ . On the other hand, for the tilting case, we also observed similar statistical behaviors between the standard deviation of  $dx$  (or  $dy$ ) and the median of  $dy$ . All these statistical relationships offer useful knowledge about the displacement of feature points when the 3-D scene is stationary.

When moving objects are present, these feature points caused by the moving objects usually have very different statistical behaviors. Hence, in the dynamic calibration process, we may calculate the median of displacements for all feature points. Based on the median, we estimate the standard deviation of displacement according to these already learned statistical relationships. When the displacement of a feature point is away from the median by three standard deviations, that feature point

$$\begin{bmatrix} \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial \theta} \end{bmatrix} = \begin{bmatrix} -\alpha \frac{(XC_\theta - ZS_\theta - rS_\theta)^2}{(XS_\theta + ZC_\theta + r(C_\theta - 1))^2} - \alpha \left( 1 + \frac{r}{XS_\theta + ZC_\theta + r(C_\theta - 1)} \right) \\ \beta Y \frac{(XC_\theta - ZS_\theta - rS_\theta)}{(XS_\theta + ZC_\theta + r(C_\theta - 1))^2} \\ \alpha X \frac{(YS_\phi + ZC_\phi + rS_\phi)}{(YS_\phi + ZC_\phi + r(C_\phi - 1))^2} \end{bmatrix} \quad (10)$$

$$\begin{bmatrix} \frac{\partial x}{\partial \phi} \\ \frac{\partial y}{\partial \phi} \end{bmatrix} = \begin{bmatrix} \beta \frac{(YC_\phi + ZS_\phi + rS_\phi)^2}{(-YS_\phi + ZC_\phi + r(C_\phi - 1))^2} + \beta \left( 1 + \frac{r}{-YS_\phi + ZC_\phi + r(C_\phi - 1)} \right) \end{bmatrix} \quad (11)$$

is treated as an undesired feature point and is discarded in the dynamic calibration process.

### III. SENSITIVITY ANALYSIS

Based on (7) and (8), we can dynamically estimate the changes of pan angle and tilt angle while a camera is rotating. In this section, we will analyze how sensitive our algorithm is with respect to the calibration errors at the previous time instant and the measurement errors at the current time instant. Here, we assume there could be some errors in the calibration results at the previous time instant  $t - 1$ . Moreover, there could be some errors in the extraction of feature points, including tracking errors and the departure of feature points from the epipolar lines.

Without loss of generality, we only discuss the sensitivity of our algorithm in the dynamic calibration of Camera-1. In theory, for the estimation of  $\Delta\theta_t^1$  and  $\Delta\phi_t^1$ , the optimization of (7) conforms to  $f_1 \equiv \partial F_t^1 / \partial(\Delta\theta_t^1) = 0$  and  $f_2 \equiv \partial F_t^1 / \partial(\Delta\phi_t^1) = 0$ . Note that in (7), the projection center  $O_{C2}$  actually has a slight movement when Camera-2 rotates. This is because the rotation center is not exactly the same as the projection center. To simplify the formulation of (7), we intentionally ignored that part in Section II. However, in the implementation of our algorithm, we actually had taken this fact into account to achieve more accurate calibration. Hence, in the following analyses,  $f_1$  and  $f_2$  depend not only on  $\theta_{t-1}^1$  and  $\phi_{t-1}^1$ , but also on  $\theta_{t-1}^2$  and  $\phi_{t-1}^2$ . On the other hand,  $f_1$  and  $f_2$  also depend on the measurement errors of  $\{\hat{p}_{j,1}^k, \hat{p}_{j,2}^k, \dots, \hat{p}_{j,n_j}^k\}$ , where  $j = 1, 2, \dots, m$  and  $k = 1$  or  $2$ . Here,  $m$  denotes the number of epipolar-lines used for dynamic calibration.

To find how  $\Delta\theta_t^1$  and  $\Delta\phi_t^1$  deviate with respect to the fluctuations of  $\theta_{t-1}^k$ , where  $k = 1$  or  $2$ , we may apply the implicit function theorem over  $f_1$  and  $f_2$  to get

$$\begin{bmatrix} \frac{\partial(\Delta\theta_t^1)}{\partial(\theta_{t-1}^k)} \\ \frac{\partial(\Delta\phi_t^1)}{\partial(\theta_{t-1}^k)} \end{bmatrix} = - \begin{bmatrix} \frac{\partial f_1}{\partial(\Delta\theta_t^1)} & \frac{\partial f_1}{\partial(\Delta\phi_t^1)} \\ \frac{\partial f_2}{\partial(\Delta\theta_t^1)} & \frac{\partial f_2}{\partial(\Delta\phi_t^1)} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial f_1}{\partial(\theta_{t-1}^k)} \\ \frac{\partial f_2}{\partial(\theta_{t-1}^k)} \end{bmatrix} \quad (12)$$

Similarly, we can deduce the formulae for  $\partial(\Delta\theta_t^1) / \partial(\phi_{t-1}^k)$ ,  $\partial(\Delta\phi_t^1) / \partial(\phi_{t-1}^k)$ ,  $\partial(\Delta\theta_t^1) / \partial(\hat{p}_{j,i}^k)$ , and  $\partial(\Delta\phi_t^1) / \partial(\hat{p}_{j,i}^k)$ .

If we assume the total variations of  $\Delta\theta_t^1$  and  $\Delta\phi_t^1$  are the combination of individual variation with respect to the fluctuations in  $\theta_{t-1}^k$  and  $\phi_{t-1}^k$  and the measurement errors in  $\{\hat{p}_{j,1}^k, \hat{p}_{j,2}^k, \dots, \hat{p}_{j,n_j}^k\}$ , we have

$$\begin{aligned} \delta(\Delta\theta_t^1) \approx & \sum_{k=1}^2 \left\{ \frac{\partial(\Delta\theta_t^1)}{\partial(\theta_{t-1}^k)} \delta(\theta_{t-1}^k) + \frac{\partial(\Delta\theta_t^1)}{\partial(\phi_{t-1}^k)} \delta(\phi_{t-1}^k) \right. \\ & \left. + \sum_{j=1}^m \sum_{i=1}^{n_j} \frac{\partial(\Delta\theta_t^1)}{\partial(\hat{p}_{j,i}^k)} \delta(\hat{p}_{j,i}^k) \right\} \end{aligned} \quad (13)$$

and

$$\begin{aligned} \delta(\Delta\phi_t^1) \approx & \sum_{k=1}^2 \left\{ \frac{\partial(\Delta\phi_t^1)}{\partial(\theta_{t-1}^k)} \delta(\theta_{t-1}^k) + \frac{\partial(\Delta\phi_t^1)}{\partial(\phi_{t-1}^k)} \delta(\phi_{t-1}^k) \right. \\ & \left. + \sum_{j=1}^m \sum_{i=1}^{n_j} \frac{\partial(\Delta\phi_t^1)}{\partial(\hat{p}_{j,i}^k)} \delta(\hat{p}_{j,i}^k) \right\}. \end{aligned} \quad (14)$$

To verify the above formulae, we perform the following simulations. Here, two cameras are assumed to have been accurately calibrated. Camera-1 is hung at a height 2.06 m. If Camera-1 is translated by  $-0.69, -0.13$ , and  $6.25$  m along  $X, Y$ , and  $Z$  axes, respectively, and then rotated by  $-143.64^\circ$  about  $Y$  axis, Camera-1 will coincide with Camera-2. At first, Camera-1 has the pan angle  $\theta_0^1 = 0$  and tilt angle  $\phi_0^1 = 20$ , while Camera-2 has the pan angle  $\theta_0^2 = 0$  and tilt angle  $\phi_0^2 = 40$ . Moreover, based on the rectified world coordinate system of Camera-1, we assume there is an epipolar plane  $\Pi$  with the homogeneous coordinates  $\pi = [0.63, 0.77, 0.09, 0.01]$ . Based on this plane  $\Pi$ , we deduce the corresponding epipolar lines on the image planes of

TABLE I  
VARIATIONS OF ESTIMATION RESULTS WITH RESPECT TO PREVIOUS ESTIMATION ERRORS AND MEASUREMENT ERRORS

$(\theta_0^1, \phi_0^1) = (0^\circ, 20^\circ), (\Delta\theta_0^1, \Delta\phi_0^1) = (0^\circ, 0.5^\circ)$						
	$\theta_0^1 - 2^\circ$	$\theta_0^1 - 1^\circ$	$\theta_0^1 + 1^\circ$	$\theta_0^1 + 2^\circ$		
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.52, 0.60	-0.26, 0.30	0.27, -0.30	0.53, -0.60		
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	-0.49, 0.60	-0.25, 0.30	0.25, -0.30	0.49, -0.59		
$\theta_0^1 - 2^\circ$ $\theta_0^1 - 1^\circ$ $\theta_0^1 + 1^\circ$ $\theta_0^1 + 2^\circ$						
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.39, -0.44	0.19, -0.22	-0.19, 0.22	-0.38, 0.44		
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.39, -0.47	0.20, -0.23	-0.20, 0.23	-0.39, 0.47		
$\hat{x}_{1,-2}$ pixels $\hat{x}_{1,-1}$ pixels $\hat{x}_{1,+1}$ pixels $\hat{x}_{1,+2}$ pixels						
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.08, -0.01	0.04, -0.01	-0.04, 0.00	-0.08, 0.01		
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.08, -0.02	0.04, -0.01	-0.04, 0.01	-0.08, 0.02		
$\hat{y}_{1,-2}$ pixels $\hat{y}_{1,-1}$ pixels $\hat{y}_{1,+1}$ pixels $\hat{y}_{1,+2}$ pixels						
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.01, -0.09	0.01, -0.04	-0.01, 0.04	-0.01, 0.09		
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.01, -0.09	0.00, -0.04	-0.00, 0.04	-0.01, 0.09		
$\theta_0^2 - 2^\circ$ $\theta_0^2 - 1^\circ$ $\theta_0^2 + 1^\circ$ $\theta_0^2 + 2^\circ$						
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.49, -0.59	0.25, -0.29	-0.25, 0.28	-0.51, 0.54		
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.49, -0.58	0.23, -0.28	-0.23, 0.28	-0.44, 0.53		
$\theta_0^2 - 2^\circ$ $\theta_0^2 - 1^\circ$ $\theta_0^2 + 1^\circ$ $\theta_0^2 + 2^\circ$						
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.27, -0.32	0.14, -0.16	-0.15, 0.16	-0.30, 0.33		
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	0.26, -0.32	0.13, 0.16	-0.13, 0.16	-0.27, 0.32		

TABLE II  
VARIATIONS OF ESTIMATION RESULTS WITH RESPECT TO DISTANCE FLUCTUATIONS IN EPIPOLAR LINES

$(\theta_0^1, \phi_0^1) = (0^\circ, 20^\circ), (\Delta\theta_0^1, \Delta\phi_0^1) = (0^\circ, 0.5^\circ)$						
d (pixels)	-3	-2	-1	+1	+2	+3
simulated $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	-0.05, 0.06	-0.04, 0.04	-0.02, 0.02	0.02, -0.02	0.04, -0.04	0.05, -0.05
deduced $\Delta(\Delta\theta_0^1), \Delta(\Delta\phi_0^1)$	-0.04, 0.05	-0.03, 0.04	-0.02, 0.02	0.02, -0.02	0.04, -0.04	0.05, -0.05

these two cameras. On each of these two epipolar lines, we randomly choose three image points  $\{p_{1,1}^k, p_{1,2}^k, p_{1,3}^k\}$  as the feature points, with  $k = 1$  or  $2$ . After that, the tilt angle of Camera-1 is changed to  $20.5^\circ$  so that the feature points on the image plane of Camera-1 will move to the new positions  $\{\hat{p}_{1,1}^1, \hat{p}_{1,2}^1, \hat{p}_{1,3}^1\}$ . Besides, the intrinsic parameters  $\{\alpha_1, \beta_1, \alpha_2, \beta_2\}$  are set to be  $\{392, 388, 392.3, 385\}$ .

In the simulation, we change individually the initial pan and tilt angles  $\{\theta_0^1, \phi_0^1, \theta_0^2, \phi_0^2\}$  of Camera-1 and Camera-2 to see how the estimated values of  $\Delta\theta_0^1$  and  $\Delta\phi_0^1$  vary. Moreover, we also change the measurement  $\hat{p}_{1,1}^1$  whose coordinates are defined as  $(\hat{x}_{1,1}^1, \hat{y}_{1,1}^1)$  to see how  $\Delta\theta_0^1$  and  $\Delta\phi_0^1$  vary. Here, the LM algorithm is applied to (7) for the estimation of  $\Delta\theta_0^1$  and  $\Delta\phi_0^1$ . The variations of these estimation results, together with the variations deduced by (13) and (14) are listed in Table I. Besides, we also show in Table II how  $\Delta\theta_0^1$  and  $\Delta\phi_0^1$  vary with respect to the distance fluctuation  $d$  in epipolar lines. In our simulation, we change the measurement  $\hat{p}_{1,1}^1$  to be away from its epipolar line. The deduced variations can be expressed as

$$\delta(\Delta\theta_0^1) \approx \frac{\partial(\Delta\theta_0^1)}{\partial(\hat{x}_{1,1}^1)} \frac{\partial(\hat{x}_{1,1}^1)}{\partial(d)} \delta(d) + \frac{\partial(\Delta\theta_0^1)}{\partial(\hat{y}_{1,1}^1)} \frac{\partial(\hat{y}_{1,1}^1)}{\partial(d)} \delta(d) \quad (15)$$

and

$$\delta(\Delta\phi_0^1) \approx \frac{\partial(\Delta\phi_0^1)}{\partial(\hat{x}_{1,1}^1)} \frac{\partial(\hat{x}_{1,1}^1)}{\partial(d)} \delta(d) + \frac{\partial(\Delta\phi_0^1)}{\partial(\hat{y}_{1,1}^1)} \frac{\partial(\hat{y}_{1,1}^1)}{\partial(d)} \delta(d). \quad (16)$$

It can be seen that the all deduced variations in Tables I and II well approximate the simulation results.

Additionally, when the number of epipolar line pair doubles, the errors of estimated  $\Delta\theta_0^1$  and  $\Delta\phi_0^1$  caused by the fluctuations of the feature points are roughly halved. On the other hand, the errors of estimated  $\Delta\theta_0^1$  and  $\Delta\phi_0^1$  caused by the fluctuations of  $\{\theta_0^1, \phi_0^1, \theta_0^2, \phi_0^2\}$  have no apparent changes. Besides, if we change the value of  $\Delta\phi_0^1$  up to  $5^\circ$ ,



Fig. 9. Test images with the presence of landmarks. The images captured by Camera-1, Camera-2, Camera-3, and Camera-4 are arranged in the left-to-right, top-to-bottom order.

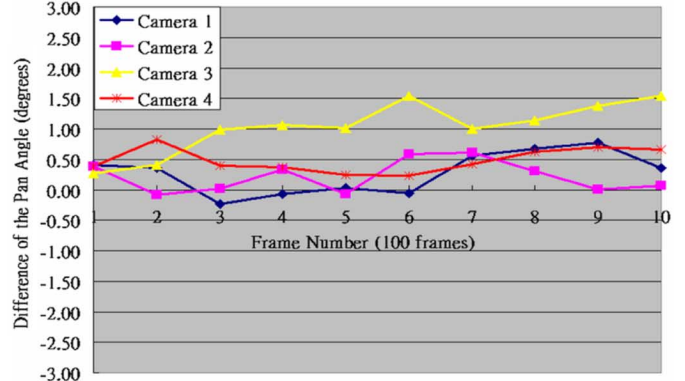
the variations of estimated  $\Delta\theta_1^1$  and  $\Delta\phi_1^1$  caused by the fluctuations of  $\{\theta_0^1, \phi_0^1, \theta_0^2, \phi_0^2\}$  and  $\{\hat{p}_{1,1}^1, \hat{p}_{1,2}^1, \hat{p}_{1,3}^1\}$  still confirm to that in Tables I and II. Finally, we also change the tilt angle  $\phi_0^1$  from  $20^\circ$  to  $80^\circ$  with a  $20^\circ$  step, and repeat the simulation. The variations of the simulation results also confirm to that in Tables I and II. In practice, the initial static calibration is usually accurate enough so that the fluctuations of  $\{\theta_0^1, \phi_0^1, \theta_0^2, \phi_0^2\}$  are usually less than  $0.5^\circ$ . Moreover, the measurement errors of  $\{\hat{p}_{1,1}^1, \hat{p}_{1,2}^1, \hat{p}_{1,3}^1\}$  are likely to be less than 2 pixels. Hence, the estimation errors of  $\Delta\theta_1^1$  and  $\Delta\phi_1^1$  are expected to be acceptable in real cases.

#### IV. EXPERIMENTS

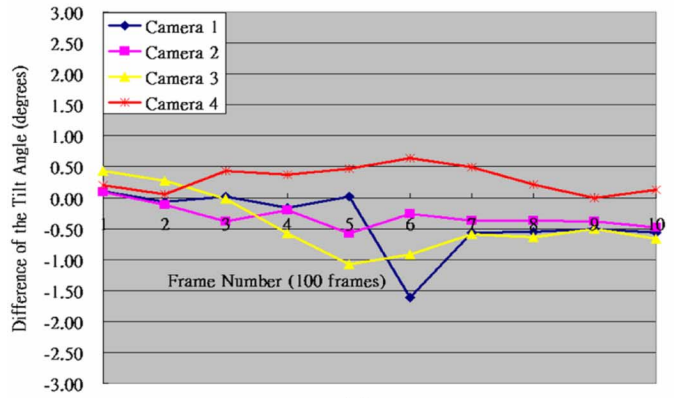
To verify the effectiveness of our dynamic calibration algorithm, we performed the following experiments over real scenes. In the first experiment, test images were captured by four cameras mounted on the ceiling. These four cameras kept panning and tilting while capturing images. In total, each camera captured 1000 test images, with the resolution of  $320 \times 240$  pixels. Besides, in order to evaluate the calibration results, we placed test landmarks in the scene with a 100-frame interval. That is, we capture 100 image frames; stop and place some landmarks in the scene; capture an image with the presence of landmarks; stop and remove these landmarks; and then resume image capturing for another 100 frames. This procedure was repeated till we captured all 1000 images for every camera. Fig. 9 shows an example of captured images by these four cameras, with the presence of landmarks.

At the beginning of the experiment, the static calibration proposed in [17] was applied to calibrate the initial setup of these four cameras. The static calibration results are listed in Table III. The left part of Table III lists for each camera the estimated tilt angle and its altitude above the brown table in the scene. The right part of Table III lists for each camera the estimated position and orientation with respect to Camera-2. In addition, we also calculated the 3-D coordinates of the test landmarks and used them as a ground truth for the evaluation of our dynamic calibration algorithm.

As cameras began to pan and tilt, we extracted 50 prominent feature points from each of these four initial images and tracked these feature points by the KLT method. In our experiment, we fixed one of the four cameras. Based on (7) and (8), we performed dynamic calibration for every image pair.



(a)



(b)

Fig. 10. (a) Differences of the pan angles and (b) differences of the tilt angles between the dynamic calibration results and the static calibration results, with one of the cameras being fixed all of the time.



Fig. 11. Evaluations of dynamical calibration at the 1000th frame.

To evaluate the results of dynamic calibration, we performed static calibration at the period of every 100 frames, based on these images with the presence of landmarks. The result was verified by projecting the aforementioned 3-D landmarks onto the image plane of each camera. Fig. 10 shows the differences of the estimated pan angles and tilt angles between the dynamic calibration results and the static calibration results. Note that the static calibration results are performed based on the 3-D landmarks that have been well calibrated at the



Fig. 12. One sample of the test sequence with the presence of a moving person.



Fig. 13. Evaluated corresponding relationship of the 1000th frame in the test sequence with a moving person.

TABLE III  
RESULTS OF THE STATIC CALIBRATION

	Tilt Angle & Altitude		Relative Position & Orientation			
	$\phi$ (degrees)	h (m)	X (m)	Y (m)	Z (m)	$\varphi$ (degrees)
Camera 1	29.0	2.14	2.92	0.11	5.04	139.5
Camera 2	51.2	2.03	0.00	0.00	0.00	0.0
Camera 3	32.5	2.25	4.61	0.22	1.68	84.8
Camera 4	49.0	1.91	-1.24	-0.12	3.58	-148.0

beginning of the experiment. In Fig. 10, it shows that the deviation at the 1000th frame is still acceptable and is within the range of  $\pm 1.5^\circ$ . Besides, the differences do not gradually increase. Moreover, based on the results of dynamic calibration, we may also directly pick up a few landmark points in the image captured by Camera-2 and project them onto the other three images, as shown in Fig. 11.

We also test the situation when a moving object is present during the dynamic calibration process. Limited by our camera control system, we cannot simultaneously control four cameras in real time. Hence, we only allow two cameras to pan and tilt in this experiment. Again, we captured 1000 frames for each camera and Fig. 12 shows a sample of the captured sequence. In Fig. 13, we show the corresponding relationship of the 1000th frame based on our dynamic calibration result. This reasonable correspondence demonstrates the effectiveness and feasibility of our dynamic calibration algorithm.

## V. CONCLUSION

In this paper, a dynamic calibration method is proposed that can be applied to a wide-range surveillance system with multiple cameras. This method does not require specific calibration patterns or complicated correspondence of feature points. It also allows the presence of

moving objects in the captured scenes, while performing dynamic calibration. This kind of dynamic calibration process can be very useful for applications related to active video surveillance.

## REFERENCES

- [1] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robot. Autom.*, vol. 3, no. 4, pp. 323–344, Aug. 1987.
- [2] M. Agrawal and L. S. Davis, "Camera calibration using spheres: A semi-definite programming approach," in *Proc. 9th IEEE Int. Conf. Comput. Vision*, Oct. 2003, vol. 2, pp. 782–789.
- [3] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," in *Proc. 7th IEEE Int. Conf. Comput. Vision*, Sep. 1999, vol. 1, pp. 666–673.
- [4] G.-Q. Wei and S. D. Ma, "A complete two-plane camera calibration method and experimental comparisons," in *Proc. 4th Int. Conf. Comput. Vision*, May 1993, pp. 439–446.
- [5] P. F. Sturm and S. J. Maybank, "On plane-based camera calibration: a general algorithm, singularities, applications," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recogn.*, Jun. 1999, vol. 1, pp. 432–437.
- [6] Z. Zhang, "Camera calibration with one-dimensional objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 7, pp. 892–899, Jul. 2004.
- [7] P. Sturm, "Algorithms for plane-based pose estimation," in *Proc. IEEE Conf. Comput. Vision Pattern Recogn.*, June 2000, vol. 1, pp. 706–711.
- [8] T. Ueshiba and F. Tomita, "Plane-based calibration algorithm for multi-camera systems via factorization of homography matrices," in *Proc. 9th IEEE Int. Conf. Comput. Vision*, Oct. 2003, vol. 2, pp. 966–973.
- [9] C. Wiles and A. Davison, "Calibrating a multi-camera system for 3D modelling," in *Proc. IEEE Workshop on Multi-View Modeling Anal. Visual Scenes*, Jun. 1999, pp. 29–36.
- [10] A. Jain, D. Kopell, K. Kakligian, and Y.-F. Wang, "Using stationary-dynamic camera assemblies for wide-area video surveillance and selective attention," in *Proc. IEEE Conf. Comput. Vision Pattern Recogn.*, Jun. 2006, vol. 1, pp. 537–544.
- [11] G. Schweighofer and A. Pinz, "Robust pose estimation from a planar target," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2024–2030, Dec. 2006.
- [12] K.-T. Song and J.-C. Tai, "Dynamic calibration of pan-tilt-zoom cameras for traffic monitoring," *IEEE Trans. Syst., Man, Cybern., Part B*, vol. 36, no. 5, pp. 1091–1103, Oct. 2006.
- [13] C. T. Huang and O. R. Mitchell, "Dynamic camera calibration," in *Proc. Int. Symp. Comput. Vision*, Nov. 1995, pp. 169–174.
- [14] B. Zhang, Y. F. Li, and Y. H. Wu, "Self-recalibration of a structured light system via plane-based homography," *Pattern Recogn.*, vol. 40, no. 4, pp. 1368–1377, Apr. 2007.
- [15] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vision Pattern Recogn.*, Jun. 1994, pp. 593–600.
- [16] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Englewood Cliffs, NJ: Prentice-Hall, 2003, pp. 216–217.
- [17] I.-H. Chen and S.-J. Wang, "An efficient approach for the calibration of multiple PTZ cameras," *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 2, pp. 286–293, Apr. 2007.