

國立交通大學

電子工程學系電子研究所

博士論文

多台攝影機之靜態與動態校正技術



Static and Dynamic Calibration
of Multiple Cameras

研 究 生：陳宜賢

指 導 教 授：王聖智 博士

中 華 民 國 九 十 七 年 一 月



多台攝影機之靜態與動態校正技術

Static and Dynamic Calibration of Multiple Cameras

研究生：陳宜賢

Student: I-Hsien Chen

指導教授：王聖智 博士

Advisor: Dr. Sheng-Jyh Wang

國立交通大學
電子工程學系 電子研究所博士班
博士論文



A Dissertation Submitted to
Department of Electronics Engineering & Institute of Electronics
College of Electrical Engineering and Computer Science
National Chiao Tung University
in Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in
Electronics Engineering
January 2008
Hsinchu, Taiwan, Republic of China

中華民國九十七年一月



多台攝影機之靜態與動態校正技術

研究生：陳宜賢

指導教授：王聖智博士

國立交通大學電子工程學系電子研究所 博士班

摘要

在本論文中，我們提出兩個新穎且有效的攝影機校正技術。第一個是多台攝影機的靜態校正方法，其以同一平面上的簡單物體之影像，投影回空間為基礎；另一個則是多台攝影機的動態校正技術，其以單一攝影機所拍之畫面在時間軸上的變化，以及多台攝影機之間的相對方位為基礎。我們所採用的系統模型，能普遍適用於大部分配有多台攝影機的監控系統，不論是靜態或動態校正方法，都不需要特別的系統架設或使用特定的校正樣本。值得一提的是，我們的動態校正技術不用作複雜的特徵點對應技術。

此論文所提的多台攝影機之靜態校正，是指尋找攝影機之間的相對位置和方向。一開始我們先推導在攝影機傾斜角度的變化下，三度空間和攝影機影像的座標轉換關係，此對應關係建立之後，再透過攝影機觀察水平面上的簡單物體，來估測此攝影機的擺放高度及其傾斜角度。接著，依據每台攝影機所估測的傾斜角度和高度，我們將各攝影機所觀測到的同一向量，投影回三度空間中，藉著比較此向量的空間座標，各攝影機之間的相對方位即可很容易地被估測出來。就某個方面來看，我們的方法可被視為將 homography 矩陣的運算，拆解成兩個簡單的校正過程，因此可以減少多台攝影機校正的運算量。除此之外，不需要使用到座

標化的校正樣本，而我們的校正結果可以提供較直接的幾何感覺。本論文亦討論關於參數波動和測量誤差的敏感性分析，我們將數學分析的結果和電腦模擬的結果都呈現出來以驗證我們的分析，實際影像的實驗結果也展現此方法的功效和可行性。

至於動態校正的問題，我們推測多台攝影機的左右轉動角度（pan angle）和傾斜角度（tilt angle）的變化情況。在這個部分中，我們將左右轉動角度的因素考慮進來，並且重新建立在攝影機左右轉動、以及上下轉動的情況下，三維空間中水平面和二維影像的對應關係，以此關係為基礎，利用影像特徵點的位移情形，和多台攝影機之間所形成的 epipolar 平面的約束，來估測各台攝影機左右轉動角度和傾斜角度的變化。此方法不需要複雜的特徵點對應技術，而且也允許移動物體出現在校正場景中，這樣的動態校正過程，對於主動式之視訊監控的相關應用將會非常地有用。此外，我們也從數學上去探討了關於測量誤差和前次估測誤差的敏感性分析，從模擬的結果，證明了左右轉動角度和傾斜角度的變化之估計誤差，在實例中是可被接受的。而此方法的功效和可行性也在實際場景的實驗中展現出來。

Static and Dynamic Calibration of Multiple Cameras

Student : I-Hsien Chen

Advisor : Dr. Sheng-Jyh Wang

Department of Electronics Engineering & Institute of Electronics
National Chiao Tung University

Abstract

In this dissertation, we present two new and efficient camera calibration techniques. The first one is the static calibration for multiple cameras, which is based on the back-projections of simple objects lying on the same plane. The other one is the dynamic calibration for multiple cameras, which is based on the temporal information on a single camera and the relative space information among multiple cameras. We adopt a system model that is general enough to fit for a large class of surveillance systems with multiple cameras. Both our static and dynamic calibration methods do not require particular system setup or specific calibration patterns. It is worthwhile to mention that, for our dynamic calibration, no complicated correspondence of feature points is needed. Hence, our calibration methods can be well applied to a wide-range surveillance system with multiple cameras.

In the problem of static calibration for multiple cameras, we infer the relative positioning and orientation among multiple cameras. The 3D-to-2D coordinate transformation in terms of the tilt angle of a camera is deduced first. After having established the 3D-to-2D transformation, the tilt angle and altitude of each camera are

estimated based on the observation of some simple objects lying on a horizontal plane. With the estimated tilt angles and altitudes, the relative orientations among multiple cameras can be easily obtained by comparing the back-projected world coordinates of some common vectors in the 3-D space. In some sense, our approach can be thought to have decomposed the computation of homography matrix into two simple calibration processes so that the computational load becomes lighter for the calibration of multiple cameras. Additionally, no coordinated calibration pattern is needed and our calibration results can offer direct geometric sense. In this dissertation, we also discuss the sensitivity analysis with respect to parameter fluctuations and measurement errors. Both mathematical analysis and computer simulation results are shown to verify our analysis. Experiment results over real images have demonstrated the efficiency and feasibility of this approach.

In the problem of dynamic calibration, we infer the changes of pan and tilt angles for multiple cameras. In this part of the thesis, we take the pan angle factor into account and re-build the mapping between a horizontal plane in the 3-D space and the 2-D image plane on a panned and tilted camera. Based on this mapping, we utilize the displacement of feature points and the epipolar-plane constraint among multiple cameras to estimate the pan-angle and tilt-angle changes for each camera. This algorithm does not require a 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. Besides, the sensitivity analysis of our dynamic calibration algorithm with respect to measurement errors and fluctuations in previous estimations is also discussed mathematically. From the simulation results, the estimation errors of pan and tilt angle changes are proved to be acceptable in real cases. The efficiency and feasibility of this approach has been

demonstrated in some experiments over real scenery.





誌 謝

有幸得王聖智教授的殷切指導，使我的人生有了一條很不一樣的路。在漫長的研究過程中，非常感謝王老師很費心思地跟我討論問題、督促我，訓練我獨立研究的精神，在辯證問題的過程中，有時候還得容忍我說話的任性，真的很感謝他，我才能在艱辛的學位求取過程走到最後。

在此，我要將這份得之不易的榮耀，獻給我最摯愛的家人，特別感謝我的父親、母親，在我遇到挫折時，適時地提供鼓勵與支持，並堅定我的信念，也謝謝妹妹宜妙，由於曾經是實驗室的一員，更能體會研究和攻讀博士所承受的壓力，因此常成為我傾吐宣洩苦悶的對象。還有謝謝正民的陪伴，在這同樣的人生階段，我們一起互相勉勵。

另外，感謝實驗室學長陳信嘉、陳俊宇適時的鼓勵和開導，還有學弟們在求學上、生活上的相互支持，同時為我的研究生活增添色彩。也要謝謝師母的關心和照顧，讓我在研究之餘，能注意健康和保養身體。一路走來，很幸運地有大家的關心和扶持，真的很謝謝你們。

最後，對於口試委員們陳稔博士、黃仲陵博士、宋開泰博士、張隆紋博士、洪一平博士、陳永昇博士、陳祝嵩博士以及王老師，感謝你們的建議和指導，尤其是陳永昇老師，曾經不吝於提供我研究上的解惑，在此致上最誠摯的敬意。

陳宜賢 2008年1月



Contents

摘要	i
Abstract	iii
誌謝	vii
Contents	ix
List of Tables	xiii
List of Figures	xiv
List of Notations	xviii
Introduction	1
1.1 Dissertation Overview	1
1.2 Organization and Contribution	5
Backgrounds	7
2.1 Projective Geometry	7
2.1.1 Perspective Projection	8
2.1.2 Epipolar Geometry	10
2.1.3 Homography	12
2.2 Camera Calibration	15

2.2.1 Calibration with Three-Dimensional Objects	15
2.2.2 Calibration with Planar Objects.....	16
2.2.3 Calibration with One-Dimensional Objects	19
2.2.4 Self-Calibration	21
2.3 Dynamic Camera Calibration.....	24
Static Calibration of Multiple Cameras.....	27
3.1 Introduction of Our Camera Model Syatem	28
3.1.1 System Overview	28
3.1.2 Camera Setup Model	29
3.2 Pose Estimation of a Single Camera.....	31
3.2.1 Coordinate Mapping on a Tilted Camera.....	31
3.2.2 Constrained Coordinate Mapping	32
3.2.3 Pose Estimation Based on the Back-Projections	32
3.2.3.1 Back-Projected Angle w.r.t. Guessed Tilt Angle	34
3.2.3.2 Back-Projected Length w.r.t. Guessed Tilt Angle	37
3.3 Calibration of Multiple Static Cameras.....	40
3.3.1 Static Calibration Method of Multiple Cameras	40
3.3.2 Discussion of Pan Angle.....	42
3.4 Sensitivity Analysis.....	44
3.4.1 Mathematical Analysis of Sensitivity	47
3.4.2 Sensitivity Analysis via Computer Simulations.....	47
3.4.2.1 Sensitivity w.r.t. u_0 and v_0	49
3.4.2.2 Sensitivity w.r.t. α and β	49
3.4.2.3 Sensitivity w.r.t. $\Delta x_i'$ and $\Delta y_i'$	49
3.4.2.4 Sensitivity w.r.t. Different Choices of Tilt Angle	50
3.5 Experiments over Real Images	55

3.5.1 Calibration Results	55
3.5.2 Discussion and Comparison with the Homography Technique.....	60
Dynamic Calibration of Multiple Cameras	65
4.1 Dynamic Calibration of Multiple Cameras	66
4.1.1 Coordinate Mapping on a Tilted and Panned Camera	67
4.1.2 Dynamic Calibration of a Single Camera Based on Temporal Information	68
4.1.3 Dynamic Calibration of Multiple Camera Based on Epipolar-Plane Constraint	70
4.2 Dynamic Calibration with Presence of Moving Objects	77
4.3 Sensitivity Analysis.....	86
4.4 Experiments over Real Scenes	91
Conclusions	99
Bibliography	103
Curriculum Vita.....	107





List of Tables

Table 3.1: Variations of Tilt Angle and Altitude with respect to Different Parameter Fluctuations and Measurement Errors.....	49
Table3.2: Variations of Tilt Angle and Altitude with respect to Different Choices of Tilt Angle.....	53
Table 3.3: Upper Table: Estimation of Tilt Angle and Altitude. Lower Table: Spatial Relationship among Cameras.....	56
Table 3.4: Mean Absolute Distance and Standard Deviation of the Point-wise Correspondence.....	62
Table 4.1: Variations of Estimation Results with respect to Previous Estimation Errors and Measurement Errors.....	88
Table 4.2: Variations of Estimation Results with respect to Distance Fluctuations in Epipolar Lines.....	88
Table 4.3: Results of the Static Calibration.....	92

List of Figures

Fig. 1.1	An example of images captured by a camera mounted on the ceiling.....	3
Fig. 1.2	An image pair with two different views. Green lines indicate three pairs of corresponding epipolar lines.....	4
Fig. 2.1	Pinhole imaging model [3, p. 4].....	9
Fig. 2.2	Perspective projection coordinate system [3, p. 28].....	9
Fig. 2.3	Illustration of epipolar-plane constraint.....	11
Fig. 2.4	A homography between two views [44, p. 325].....	12
Fig. 2.5	A homography compatible with the epipolar geometry [44, pp. 328].....	13
Fig. 2.6	The fundamental matrix can be represented by $\mathbf{F} = [\mathbf{e}]_{\times} H_{\Pi}$, where H_{Π} is the projective transform from the second to the first camera, and $[\mathbf{e}]_{\times}$ represents the fundamental matrix of the translation [44, p. 250].....	14
Fig. 2.7	A 3-D calibration pattern with regularly arranged rectangles.....	16
Fig. 2.8	Camera imaging system of a one-dimensional object [12].....	20
Fig. 2.9	An example of the calibration operation by using a 1-D object with a fixed point [12].....	20
Fig. 2.10	The epipolar tangency to the absolute conic images [18].....	22
Fig. 3.1	Flowchart of the proposed static calibration procedure.....	28

Fig. 3.2	Model of camera setup.....	29
Fig. 3.3	Geometry of a horizontal plane Π with respect to a tilted camera.....	31
Fig. 3.4	(a) Rectangular corner captured by a tilted camera (b) Illustration of back-projection onto a horizontal plane on for different choices of tilt angles.....	35
Fig. 3.5	Back-projected angle with respect to guessed tilt angles.....	36
Fig. 3.6	Back-projected length with respect to guessed tilt angle. Each curve is generated by placing a line segment on some place of a horizontal plane.....	38
Fig. 3.7	(a) Top view of two cameras and a vector in the space (b) The world coordinates of the vector with respect to these two cameras.....	40
Fig. 3.8	(a) Three points marked in the image captured by a PTZ camera (b) Top view of the back-projected corners and the optical axes with respect to different guessed pan angles.....	42
Fig. 3.9	(a) Top view of line segments placed on a horizontal plane (b) Top view of corners placed on a horizontal plane.....	47
Fig. 3.10	Variations of the (a) L-v.s.- ϕ curves and (b) ψ -v.s.- ϕ curves with respect to the variation of v_0	50
Fig. 3.11	Variations of the (a) L-v.s.- ϕ curves and (b) ψ -v.s.- ϕ curves with respect to the variation of β	51
Fig. 3.12	Sensitivity w.r.t. fluctuations of x_i' and y_i'	52
Fig. 3.13	(a) Test image (b) Deduced ψ -v.s.- ϕ curves (c) Deduced L-v.s.- ϕ curves....	55
Fig. 3.14	(a) Test image captured by four cameras (b) Top view of the relative positions between four cameras.....	58
Fig. 3.15	Evaluation of calibration results.....	59
Fig. 3.16	Test images with a rectangular calibration pattern.....	60
Fig. 3.17	Evaluation of calibration results by using five points. (a) Point correspondence based on the homography technique. (b) Point correspondence based on the proposed method.....	62
Fig. 4.1	Flowchart of the proposed dynamic calibration algorithm.....	65

Fig. 4.2	Illustration of a pseudo plane Π'	69
Fig. 4.3	Illustration of epipolar-plane constraint.....	70
Fig. 4.4	Image pairs captured at two different time instants. Green lines indicate three pairs of corresponding epipolar lines.....	74
Fig. 4.5	(a) Image captured by a camera with 55.1° tilt angle. (b) Image captured by a camera with 54.6° tilt angle. Red crosses represent feature points extracted by the KLT algorithm.....	77
Fig. 4.6	The distribution of spatial displacement for the extracted feature points in Fig. 4.5.....	78
Fig. 4.7	Illustration of the coordinate system when camera is panning. If r is far smaller than Z' , we may simply dismiss r	80
Fig. 4.8	(a) The displacements of feature points observed by two different cameras. Both cameras are under a 1-degree pan-angle change, while their tilt angles are fixed at 34.8° . (b) The displacements of feature points observed by the same camera but with different pan-angle changes. (Blue: 0.6-degree pan-angle change. Red: 1-degree pan-angle change.).....	81
Fig. 4.9	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 statistical relationships for Camera-1, Camera-2, Camera-3, and Camera-4 are plotted in red, blue, green, and magenta, respectively....	83
Fig. 4.10	(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.....	84
Fig. 4.11	(a) Test images captured by four cameras. (b) 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.....	91
Fig. 4.12	Evaluation of initial calibration.....	92
Fig. 4.13	(a) Differences of the pan angles between the dynamic calibration results and the static calibration results. (b) Differences of the tilt angles between the dynamic calibration results and the static calibration results.....	94
Fig. 4.14	Evaluations of dynamical calibration at (a) the 300th frame, (b) the 600th frame, and (c) the 1000th frame.....	95

Fig. 4.15 (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.....96

Fig. 4.16 One sample of the test sequence with the presence of a moving perso.....97

Fig. 4.17 Evaluated corresponding relationship of the 1000th frame in the test sequence with a moving person.....97




List of Notations

$B(\cdot)$	The back projection function
$\ell(\phi)$	The length of a back-projected segment in terms of tilt angle
α	The scale parameter relates a distance level to a pixel level in the x axis
β	The scale parameter relates a distance level to a pixel level in the y axis
u_0	The x coordinate of the principal point
v_0	The y coordinate of the principal point

CHAPTER 1

Introduction

1.1 Dissertation Overview



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 may need to be recalibrated. In practice, the rotatory encoders of most conventional cameras are not sufficiently accurate, while cameras with high accuracy encoders are rather expensive. For example, for the cameras we use in our experiments, a request of 1-degree rotation may cause a 0.1-degree error in panning or a 0.25-degree error in tilting. Even though we may correct this error via an off-line training, we may still face a synchronization problem. This synchronization problem is caused by the fact that a camera keeps capturing images when it is under panning or tilting. That is, during the period of one rotation request, the camera may have captured tens of image frames. Even if we may correct the angle error for each rotation request, we still have difficulty in estimating the camera pose for each frame

unless we know the exact timing of camera's movement and the sampling instant of each image frame. With multiple cameras, the synchronization problem becomes even more complicated. In this case, the use of pre-encoder may not offer instantaneous multi-view geometry information at any time instant. Hence, instead of the rotatory encoder, we seek to recalibrate a set of multiple cameras based on the feedback of visual information in the captured images.

Up to now, various kinds of approaches have been developed to calibrate static camera's intrinsic and/or extrinsic parameters, such as the techniques proposed in [1]-[43]. 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, [9]-[11] 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, some methods have been proposed in the literature [38]-[43]. However, they are not general enough to be applied in a wide-range surveillance system with multiple active cameras. Besides, [42] and [43] require the correspondence of feature points on the image pair. For surveillance systems with wide-range coverage, the matching of feature points is usually a difficult problem.

In this thesis, we first demonstrate a new and efficient approach to calibrate multiple cameras without movement. For the static calibration, we estimate the tilt angle and altitude of each camera as a starting point. The concept of our approach originated from the observation that people could usually make a rough estimate about the tilt angle of the camera simply based on some clues revealed in the captured images. Based on our approach, once a set of cameras are settled, we can simply place a few simple patterns on a horizontal plane. These patterns can be A4 papers, books,

boxes, etc.; and the horizontal plane can be a tabletop or the ground plane. The whole procedure does not need specially designed calibration patterns. For example, with the image shown in Fig. 1.1, with the shape of the tabletop and these A4 papers on the table, we can easily infer that the camera has a pretty large tilt angle, which is expected to be larger than 45 degrees. Once the tilt angle and altitude of each camera are estimated, we will show that the relative positions and orientations among these cameras can be easily calibrated, without the need to calculate the homography matrix. It is worthwhile to mention that, in some sense, our approach can be thought to have decomposed the computation of homography matrix into two simple calibration processes so that the computational load becomes lighter for the calibration of multiple PTZ cameras.



Fig. 1.1 An example of images captured by a camera mounted on the ceiling.

So far as we know, most calibration algorithms require corresponding feature points, special calibration patterns, or known landmarks in the three dimensional space. To dynamically calibrate multiple cameras, calibration patterns and landmarks are not always applicable since they may get occluded or even move out of the

captured scenes when cameras pan or tilt. On the other hand, if using the correspondence of feature points, we need to keep updating the correspondence of feature points when cameras rotate. For a wide-range surveillance system with many cameras, the correspondence of feature points cannot be easily solved. Take Fig. 1.2 as an example, the captured scenes of two cameras are quite different. We show three pairs of corresponding epipolar lines on these two images. It can be observed that feature points on each pair of corresponding epipolar lines may not originate from the same 3-D points. For this kind of image pair, the matching of feature points is not a simple task. Hence, after the static calibration of multiple cameras, we seek to recalibrate multiple cameras without specific calibration patterns and without complicated correspondence techniques.

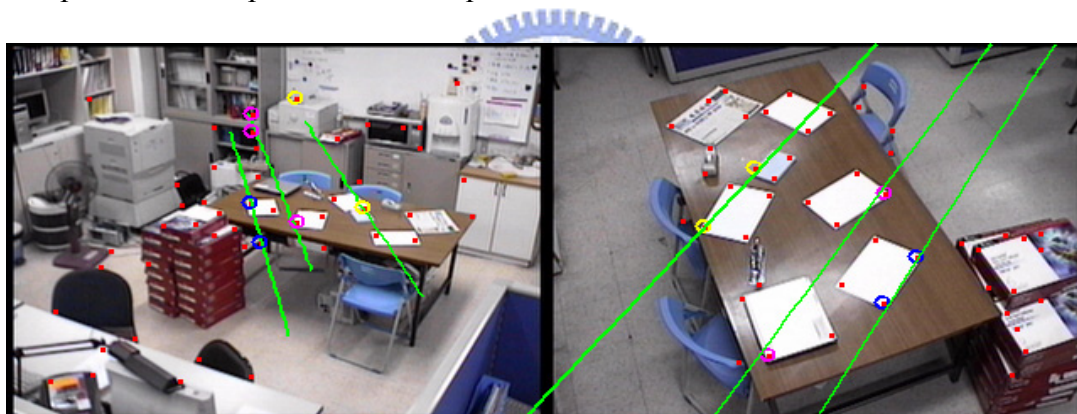


Fig. 1.2 An image pair with two different views. Green lines indicate three pairs of corresponding epipolar lines.

Based on the result of our static calibration of multiple cameras, we begin to perform dynamic calibration when the cameras are under movement. The concept of our approach originated from the observation that people can usually 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. The major advantage of our dynamic calibration algorithm is that it does not

require a complicated correspondence of feature points. As cameras begin to pan or tilt, we keep extracting and tracking feature points based on the Kanade-Lucas-Tomasi (KLT) algorithm [45]. 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 [42], 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. Our algorithm also allows the presence of moving objects in the captured scenes while performing dynamic calibration. This property makes our approach practical for general surveillance systems.

1.2 Organization



The following chapters in this dissertation are organized as follows.

- ◆ In Chapter 2, we first introduce the basic camera projection geometry, including the perspective projection, the epipolar geometry and the homography concept. Next, a few literatures are briefly reviewed.
- ◆ In Section 3.1, the camera model of our surveillance system is first described. Next, in Section 3.2, we develop the mapping between the 3-D space and the 2-D image plane in terms of tilt angle, under the constraint that all observed points are lying on a horizontal plane. Based on the back projection formula, the tilt angle and altitude of a camera can thus be estimated by viewing some simple patterns on a horizontal plane. Then, we will introduce how to utilize the estimation results to achieve the calibration of multiple cameras in Section 3.3. In addition, the sensitivity analysis with respect to parameter fluctuations and measurement errors will be discussed in Section 3.4. In Section 3.5, some

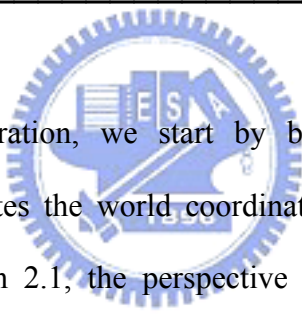
experimental results over real data are demonstrated to illustrate the feasibility of the proposed static calibration method.

- ◆ In Section 4.1, based on the results of our static calibration, 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. Then, in Section 4.2, we describe how to filter out undesired feature points when moving objects are present. After that, the sensitivity analysis with respect to measurement errors and the fluctuations of previous estimations will be addressed in Section 4.3. In Section 4.4, the efficiency and feasibility of this dynamic calibration approach are demonstrated in some experiments over real scenery.
- ◆ Finally, conclusions are drawn in Chapter 5.



CHAPTER 2

Backgrounds



To understand camera calibration, we start by briefly introducing the camera projection geometry that relates the world coordinate system with camera's image coordinate system. In Section 2.1, the perspective projection commonly used for camera calibration will be introduced first. Next, we will show the geometric relationship of two views — the epipolar geometry and the homography. Then, in Section 2.2, we roughly classify some existing calibration methods [1]-[43] based on the usage of the calibration objects. Additionally, some dynamic calibration approaches are briefly introduced in Section 2.3.

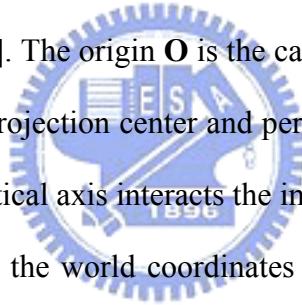
2.1 Projective Geometry

The commonly used camera model for camera calibration is the pinhole camera model, which is also called the perspective projection model. Although real cameras are usually equipped with lenses, the perspective projection model often approximates well enough to an acceptable camera projection process. In Section 2.1.1, we first

introduce the perspective projection and the camera parameters that relate the world coordinate system with the image coordinate system. Then, for multiple cameras, we extend our discussion to the two-view geometry, the epipolar geometry and the homography in Section 2.1.2 and Section 2.1.

2.1.1 Perspective Projection

The pinhole perspective projection model was first proposed by Brunelleschi at the early 15th century [3, pp. 3-6], as illustrated in Fig. 2.1. For the sake of convenience, we consider a virtual image in front of the pinhole, instead of the inverted image behind the pinhole. The distances from this virtual image to the pinhole and from the pinhole to the actual image are the same. Figure 2.2 illustrates the perspective projection system [3, p. 28-30]. The origin \mathbf{O} is the camera projection center (pinhole). The ray passing through the projection center and perpendicular to the image plane is called the optical axis. The optical axis intersects the image plane at the image center \mathbf{C} . Assume $P=[X, Y, Z]^T$ denotes the world coordinates of a 3-D point \mathbf{P} and its image coordinates are denoted as $p=[x, y]^T$. Under perspective projection, we have



$$\begin{cases} x = kf \frac{X}{Z} = \alpha \frac{X}{Z}, \\ y = lf \frac{Y}{Z} = \beta \frac{Y}{Z}. \end{cases} \quad (2.1)$$

Here, the image point is expressed in pixel units. The scale parameters k and l relate from a distance level to a pixel level. To simplify the equations, we replace kf and lf with α and β , respectively.

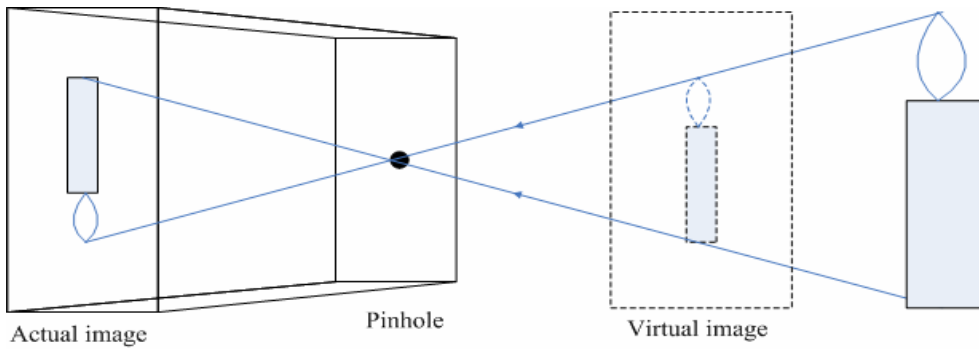


Fig. 2.1 Pinhole imaging model [3, p. 4].

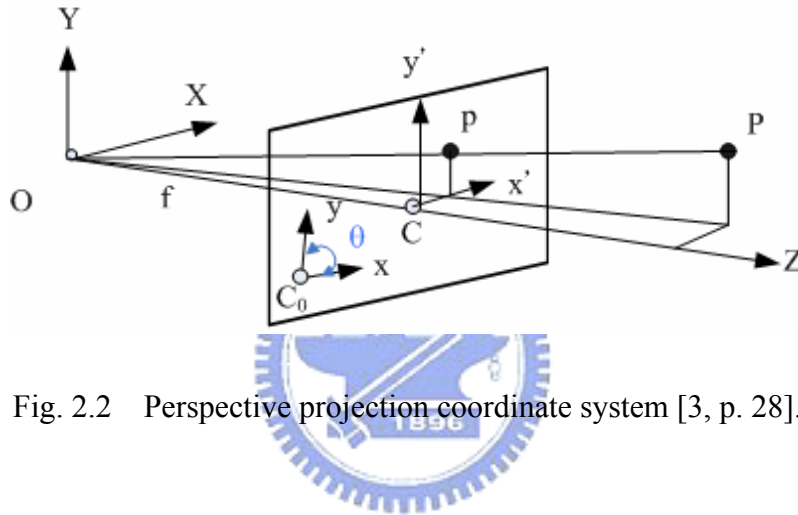


Fig. 2.2 Perspective projection coordinate system [3, p. 28].

Generally, the origin of the image coordinate system is not at the image center \mathbf{C} but at the lower-left or upper-left corner \mathbf{C}_0 . Hence, we add (u_0, v_0) in (2.2) to represent the principal point \mathbf{C} in pixel units.

$$\begin{cases} x = \alpha \frac{X}{Z} + u_0 \\ y = \beta \frac{Y}{Z} + v_0 \end{cases} \quad (2.2)$$

Moreover, due to the manufacturing errors, the two image axes may have an angle θ which is not equal to 90 degrees. This makes (2.2) to be

$$\begin{cases} x = \alpha \frac{X}{Z} - \alpha \cot \theta \frac{Y}{Z} + u_0 \\ y = \frac{\beta}{\sin \theta} \frac{Y}{Z} + v_0 \end{cases} \quad (2.3)$$

These parameters α , β , u_0 , v_0 , and θ are called the intrinsic parameters of a camera.

Usually, the perspective image plane will be moved to the front of the pinhole with a unit distance. For such a normalized coordinate system, the perspective projection can be expressed by

$$p = \frac{1}{z} [K \ \mathbf{0}] P, \text{ where } K = \begin{bmatrix} \alpha & -\alpha \cot \theta & u_0 \\ 0 & \frac{\beta}{\sin \theta} & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.4)$$

In (2.4), We reassume $P = [X, Y, Z, 1]^T$ denotes the homogeneous world coordinates of \mathbf{P} and $p = [x, y, 1]^T$ denotes the homogeneous image coordinates of \mathbf{p} 's perspective projection. Additionally, if the world coordinate system of \mathbf{P} does not coincide with that in the camera projection system, the mapping between the image plane and the 3-D space becomes

$$p = \frac{1}{z} K [R \ t] P. \quad (2.5)$$

Here, R is a rotation matrix and t is a translation vector. They are called extrinsic parameters.

2.1.2 Epipolar Geometry

Now considering a more complicated situation, we introduce the geometric relationship between two views of the same scene [3, p. 216-219]. We assume two cameras are observing the scenery. For these two cameras, their projection centers, \mathbf{O} and \mathbf{O}' , together with a 3-D point \mathbf{P} , determine an epipolar plane Π , as shown in Fig. 2.3. This epipolar plane Π intersects the image planes of the cameras to form two epipolar lines l and l' . The epipolar line l passes through the epipole e while l' passes through e' . The epipole e is the projection of \mathbf{O}' observed by the first camera, while e'

is the projection of \mathbf{O} observed by the second camera. If \mathbf{p} and \mathbf{p}' are the projected points of \mathbf{P} on these two image planes, they must lie on l and l' , respectively. This epipolar constraint implies that \mathbf{O} , \mathbf{O}' , \mathbf{p} , and \mathbf{p}' are coplanar. For calibrated cameras with known intrinsic parameters, this constraint can be expressed as

$$\overline{Op} \cdot [\overline{OO'} \times \overline{O'p'}] = 0. \quad (2.6)$$

If we choose the first camera coordinate system as the reference coordinate system and consider the coordinate transformation of the second one, (2.6) can be rewritten as

$$p \cdot [t \times (Rp')] \equiv p^T Ep' = 0. \quad (2.7)$$

Here, p and p' are homogeneous image coordinate vectors, t is the translation vector $\overline{OO'}$, and R is the rotation matrix. If a vector has the coordinates v' in the second camera coordinate system, from the view of the first one, this vector has the coordinates $v = Rv'$. Moreover, $E = t \times R$ is called the *essential matrix* [3, p. 217]. By (2.7), Ep' and $E^T p$ can be interpreted as the homogeneous coordinates of the epipolar lines l and l' in terms of the image points \mathbf{p} and \mathbf{p}' , respectively.

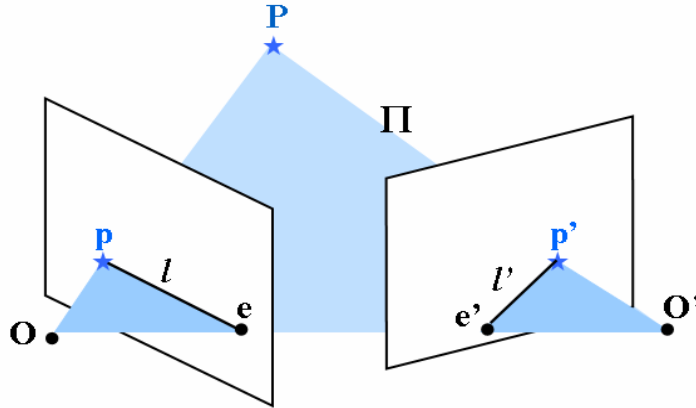


Fig. 2.3 Illustration of epipolar-plane constraint.

Furthermore, when we consider the intrinsic parameters of these two cameras, based on (2.4), the world coordinates of \mathbf{P} observed by the first camera (second

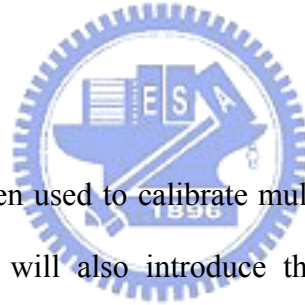
camera) can be represented by $K^{-1}p$ ($K'^{-1}p'$) up to a scale. In this way, (2.7) is rewritten as

$$p^T K^{-T} E K'^{-1} p' \equiv p^T F p' = 0. \quad (2.8)$$

Equation (2.8) is called the Longuet-Higgins equation and F is called the *fundamental matrix* [3, pp. 218-219]. Similar to E , Fp' and $F^T p$ can be interpreted as the epipolar line l and l' , respectively. Therefore, F can be considered as the mapping from an image point on one view to the epipolar line on the other view.

The epipolar constraint plays an important role and is often used in the camera calibration. Based on the information of the point correspondence among multiple views of image frames, we may extract the intrinsic parameter K or/and the extrinsic parameters R and t .

2.1.3 Homography



Because a homography is often used to calibrate multiple views of cameras, we also briefly describe it here. We will also introduce the combination of the epipolar constraint and homography [44, pp. 325-343].

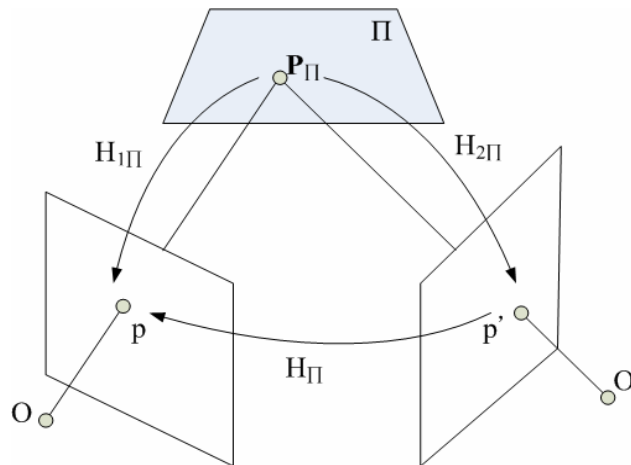


Fig. 2.4 A homography between two views [44, p. 325].

As shown in Fig. 2.4, a homography H_{Π} that is induced by a plane Π in the 3-D space can map the image points \mathbf{p} and \mathbf{p}' between two views. To be more apprehensible, through the transformation $H_{2\Pi}$, we first back-project the image point \mathbf{p}' on the second frame to the space point \mathbf{P}_{Π} on the plane Π . Then, \mathbf{P}_{Π} is projected to the image point \mathbf{p} on the first frame by the transformation $H_{1\Pi}$. This procedure can be expressed as

$$p = H_{1\Pi} H_{2\Pi}^{-1} p' = H_{\Pi} p'. \quad (2.9)$$

In theory, the 3×3 matrix H_{Π} can be obtained by four image point correspondences between two views. However, a homography needs to conform to the epipolar constraint so that the mapping of the two image planes can obey the projective geometry.

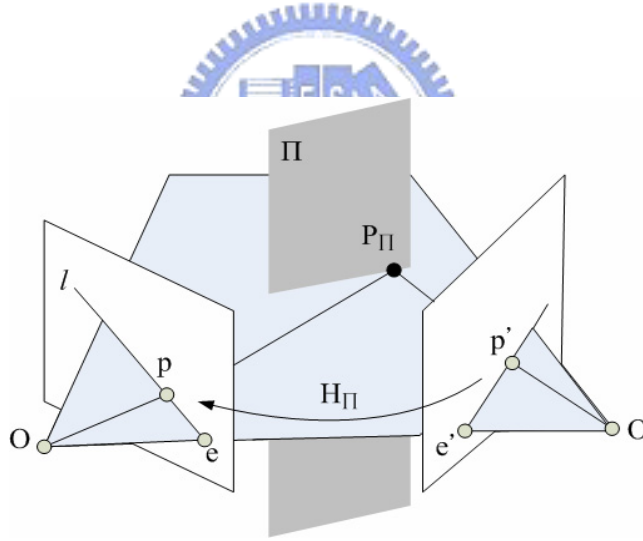


Fig. 2.5 A homography compatible with the epipolar geometry [44, pp. 328].

Figure 2.5 shows the projective geometry combining a homography induced by a plane Π and the epipolar plane constraint. In the homogeneous forms, the epipolar line l can be represented by

$$l = e \times p = [e]_{\times} (H_{\Pi} p'). \quad (2.10)$$

As mentioned in Section 2.1.2, the epipolar constraint related with one image point on

one view and one epipolar line on the other view can be expressed as $l = Fp'$.

Combining this constraint with (2.10), we can obtain

$$\mathbf{F} = [\mathbf{e}]_{\times} H_{\Pi}. \quad (2.11)$$

This formula is illustrated in Fig. 2.6.

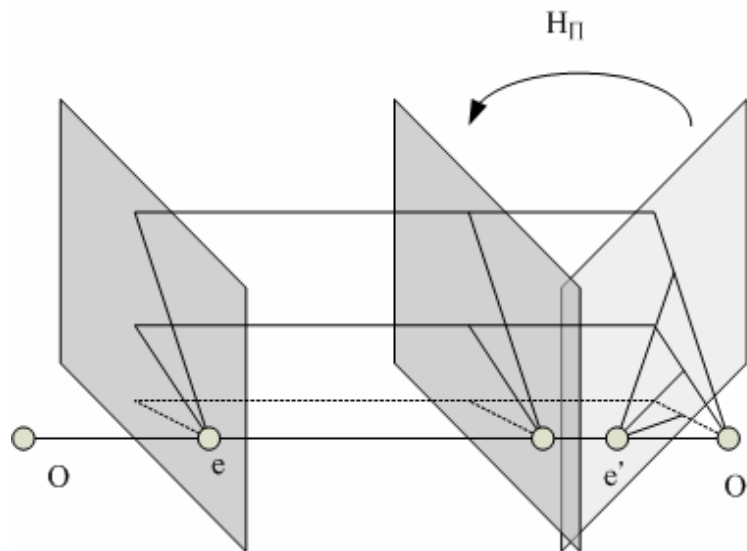


Fig. 2.6 The fundamental matrix can be represented by $\mathbf{F} = [\mathbf{e}]_{\times} H_{\Pi}$, where H_{Π} is the projective transform from the second to the first camera, and $[\mathbf{e}]_{\times}$ represents the fundamental matrix of the translation [44, p. 250].

Here, we simply mention the concept of a homography and add the epipolar constraint on it. Some papers [7], [10], [11], [14]-[36] have developed their camera calibration methods based on this compatibility constraint.

2.2 Camera Calibration

Up to now, plenty of camera calibration methods have already been developed in the literature [1]-[36]. According to their calibration objects, these methods can be roughly classified into four categories: calibration with three-dimensional objects, calibration with planar objects, calibration with one-dimensional objects, and self-calibration (with no specific objects). These methods will be briefly introduced in Section 2.2.1- 2.2.4.

2.2.1 Calibration with Three-Dimensional Objects

This type of calibration methods [1]-[3, pp. 38–53] uses 3-D objects or 3-D reference points with known world coordinates to calibrate cameras. Among these methods, O. Faugeras [2] proposed an approach that uses the calibration pattern as shown in Fig. 2.7. Such a calibration object usually contains two or three planes orthogonal to each other so that the object forms a reference world coordinate system. Some regularly arranged rectangles are on these planes. In this way, the coordinates of the corner on these rectangles are exactly known. Based on these reference correspondences between the world coordinate system and the image coordinate system, the projective map M which is called the camera matrix can be obtained by minimizing the geometric distance errors as follows:

$$\sum_i d(p_i, \hat{p}_i)^2 . \quad (2.12)$$

In (2.12), i is the number of corresponding points, and $\hat{p}_i = MP_i$. Finally, the intrinsic and extrinsic parameters can be estimated by decomposing M . Beside this approach, [1] uses different sets of 3-D reference points for calibration, while [3] offers some other optimization processes to estimate the intrinsic and extrinsic parameters.

Basically, these approaches tried to build the mapping between the 3-D coordinate system and the image coordinate system.

In summary, these methods based on 3-D reference points usually need special set-ups. Moreover, they can hardly be applied to the calibration of multiple cameras since the 3-D calibration object has to be in the view of all cameras. Especially, for dynamic calibration, it is even more difficult to calibrate cameras based on these methods

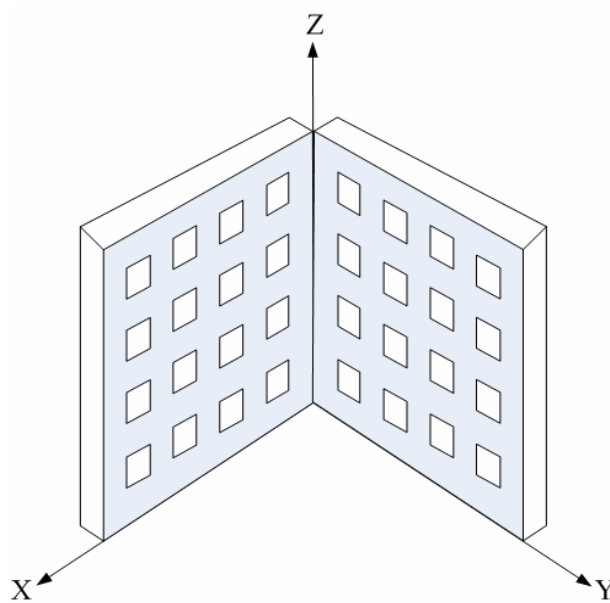


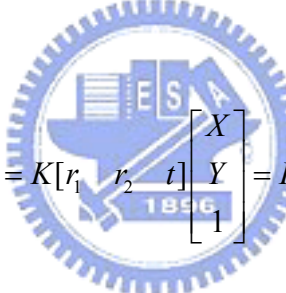
Fig. 2.7 A 3-D calibration pattern with regularly arranged rectangles.

2.2.2 Calibration with Planar Objects

Since planar objects can usually be observed in the scene and are easier to be patched with some specific geometric features, some calibration methods [4]-[11] have been proposed by using planar calibration objects. Compared with the 3-D calibration objects, planar objects are more suitable for the calibration of multiple cameras [8]-[11].

In principle, the majority of these plane-based calibration methods built the homography between a viewed plane in the 3-D space and its projection on the image plane. Based on sufficient point correspondences, this homography can be estimated. Furthermore, by changing the rotation and translation of this calibration plane several times, we have several homographies, where the same camera intrinsic parameters are embedded. From these homographies, the intrinsic parameters can be extracted by applying some constraints. The differences among different methods lie on the adopted conditions, such as a known structure of planar features or a known external motion of the calibration plane.

We take [4] as an example, due to its flexibility and easier implementation. Under perspective projection, based on (2.5), the principle procedure mentioned above can be formulized by



$$\lambda p = K [R \ t] P = K [r_1 \ r_2 \ t] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = H \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}, \quad (2.13)$$

where λ is an arbitrary scale factor; and r_1 and r_2 are the first two column vectors of the rotation matrix R . Note that because \mathbf{P} is on a plane, its coordinates can be simplified to be $[X, Y, 1]^T$ without loss of generality. In addition, from the fact that r_1 and r_2 are orthogonal to each other, the other two constraints of the homography and the intrinsic parameters can be obtained as

$$h_1^T K^{-T} K^{-1} h_2 = 0 \quad (2.14)$$

and

$$h_1^T K^{-T} K^{-1} h_1 = h_2^T K^{-T} K^{-1} h_2. \quad (2.15)$$

In (2.14) and (2.15), h_1 , h_2 , and h_3 are the three column vectors of H . However, there are 6 extrinsic parameters, 3 for rotation and 3 for translation; while a homography has 8 degrees of freedom. Having one homography provides only two constraints on

the intrinsic parameters. Hence, we need at least three different views to solve 6 intrinsic parameters. In [4], an additional parameter, lens distortion, was considered.

Other plane-based methods [8]-[11] have been specially designed for the calibration of multiple cameras. The work in [8] calibrates the intrinsic parameters with a planar grid first. Then, relative to a reference grid on the floor, the position of each camera is estimated. However, this method needs to take several image views to complete the calibration task. In comparison, [9] needs fewer views. In [9], the proposed process is like an integration of static camera calibration and “moving” camera calibration. It needs a multi-camera rig to change the specific orientation of cameras to capture two or more views of a calibration grid. With such a known condition of camera motion, this approach needs at least two views to recover the fixed intrinsic parameters and the extrinsic parameters of cameras. On the other hand, the approaches proposed in [10] and [11] belong to factorization-based methods. In [10], the cameras are assumed to be well calibrated beforehand. The author recovered the poses of multiple planes and multiple views relative to a global 3-D world reference frame by using coplanar points with known Euclidean structure. The method in [11] is an extension of [7]. Both intrinsic and extrinsic parameters can be estimated via factorization of homography matrices.

However, for surveillance systems with multiple cameras, these elaborate processes and the adopted constraints do not seem to be practical choices. Even though such 2-D calibration objects are simpler than the 3-D calibration object mentioned in Section 2.2.1, specific planar calibration objects with known structure are still needed to achieve the calibration task. As the number of cameras increases, or for wide-range multi-camera systems, a planar object may not be simultaneously observed by all cameras. Thereafter, more calibration objects are needed and more image frames need be captured to complete the calibration process. Besides, for

dynamic calibration, it is not an efficient way to repeatedly adopt these methods to recalibrate cameras.

2.2.3 Calibration with One-Dimensional Objects

Recently, Zhang [12] proposed a camera calibration method that used a one-dimensional object with three points on it. The length of this object L and the relative positions between these points are known in advance. In addition, one of these points is fixed in the 3-D space. The camera imaging system of these collinear space points \mathbf{A} , \mathbf{B} , and \mathbf{C} is illustrated in Fig. 2.8. With these constraints, some equations are deduced as follows.

$$\|B - A\|^2 = L^2 \quad (2.16)$$

$$C = \lambda_A A + \lambda_B B \quad (2.17)$$

Based on (2.5), when $[R \ t]$ were chosen as $[I \ 0]$, the following equation is obtained.

$$A = z_A K^{-1} \mathbf{a}, B = z_B K^{-1} \mathbf{b}, \text{ and } C = z_C K^{-1} \mathbf{c} \quad (2.18)$$

In (2.18), z_A , z_B , and z_C are the unknown depths of \mathbf{A} , \mathbf{B} , and \mathbf{C} , respectively. Based on (2.18), (2.17) is rewritten as

$$z_C \mathbf{c} = z_A \lambda_A \mathbf{a} + z_B \lambda_B \mathbf{b} \quad (2.19)$$

By applying cross-product with \mathbf{c} on (2.19), (2.20) is obtained.

$$z_A \lambda_A (\mathbf{a} \times \mathbf{c}) + z_B \lambda_B (\mathbf{b} \times \mathbf{c}) = 0 \quad (2.20)$$

Finally, based on (2.16), (2.18) and (2.20), a basic camera calibration constraint by using a 1-D object is obtained as follows.

$$z_A^2 h^T K^{-T} K^{-1} h = L^2. \quad (2.21)$$

In (2.21), $h = \mathbf{a} + \frac{\lambda_A (\mathbf{a} \times \mathbf{c}) \cdot (\mathbf{b} \times \mathbf{c})}{\lambda_B (\mathbf{b} \times \mathbf{c}) \cdot (\mathbf{b} \times \mathbf{c})} \mathbf{b}$. Hence, with the known length of the calibration

bar, the known position of the point \mathbf{C} with respect to \mathbf{A} and \mathbf{B} , and a fixed point \mathbf{A} ,

the 5 intrinsic parameters of the camera together with z_A can be estimated by using at least six different views of the calibration bar. Such a calibration operation with a fixed space point is shown in Fig. 2.9. A more detailed discussion of this calibration method can be found in [13].

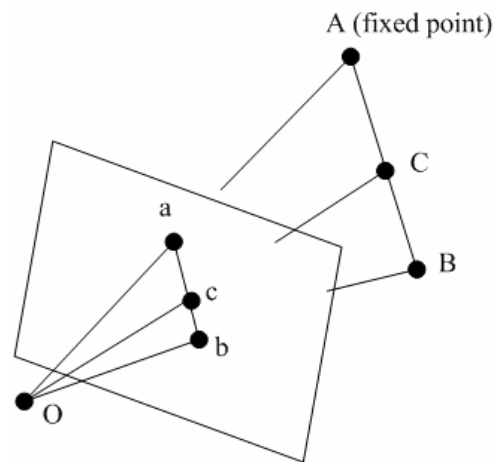


Fig. 2.8 Camera imaging system of a one-dimensional object [12].

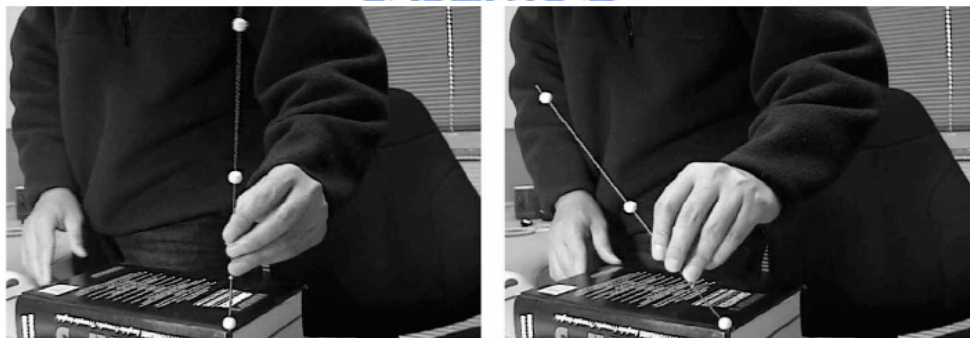


Fig. 2.9 An example of the calibration operation by using a 1-D object with a fixed point [12].

Since a 1-D object with known geometry is easy to be constructed and is more likely to be observed by multiple cameras at the same time, this calibration method seems to be potentially suitable for the calibration of multiple cameras. However, during the calibration, it still needs manual operations. That is, we need to fix a 3-D

point and change the direction of the calibration stick. Otherwise, a special calibration pattern will be required. For active cameras that may change their poses from time to time, such a technique does not seem to be a practical choice either.

2.2.4 Self-Calibration

Several research works about self-calibration [14]-[37] have been done in the last decade. The theory of self-calibration was first introduced by Maybank and Faugeras [14]. Very different from the aforementioned calibration methods, self-calibration methods do not require either calibration objects with known structure or the motion information of a camera. Based on the epipolar constraint produced by the displacement of an uncalibrated camera, the camera can be calibrated via the absolute conic Ω .

Here, we briefly describe the major kind of self-calibration techniques. The absolute conic is defined to be a conic of purely imaginary points on the plane at infinity. It can be expressed as

$$\left. \begin{array}{l} X_1^2 + X_2^2 + X_3^2 \\ X_4 \end{array} \right\} = 0,$$

$$(X_1, X_2, X_3)\Omega(X_1, X_2, X_3)^T = 0, \quad (2.22)$$

where $\Omega = \mathbf{I}$.

The absolute conic has an important property that its image ω is invariant under rigid motions of a camera. Under perspective projection, the dual matrix of ω can be represented by $\omega^* = KK^T$. Figure 2.10 shows the epipolar constraints of ω between two image frames. The first camera constraint is that the epipolar line $l = e \times p$ is tangent to ω if and only if

$$(e \times p)^T \omega^* (e \times p) = 0. \quad (2.23)$$

The second camera constraint is that the epipolar line $l' = Fp$ represented by the

corresponding point p on the first image and the fundamental matrix F is tangent to ω' if and only if

$$p^T F^T \omega'^* F p = 0. \quad (2.24)$$

Equations (2.23) and (2.24) are the so-called Kruppa equations [37]. If the intrinsic parameters are constant, (2.23) and (2.24) can be further combined into (2.25).

$$[e]_{\times} \omega^* [e]_{\times} = F^T \omega'^* F. \quad (2.25)$$

From at least three different views where each F can be obtained based on point correspondences between two views, the intrinsic parameters of the camera can be extracted.

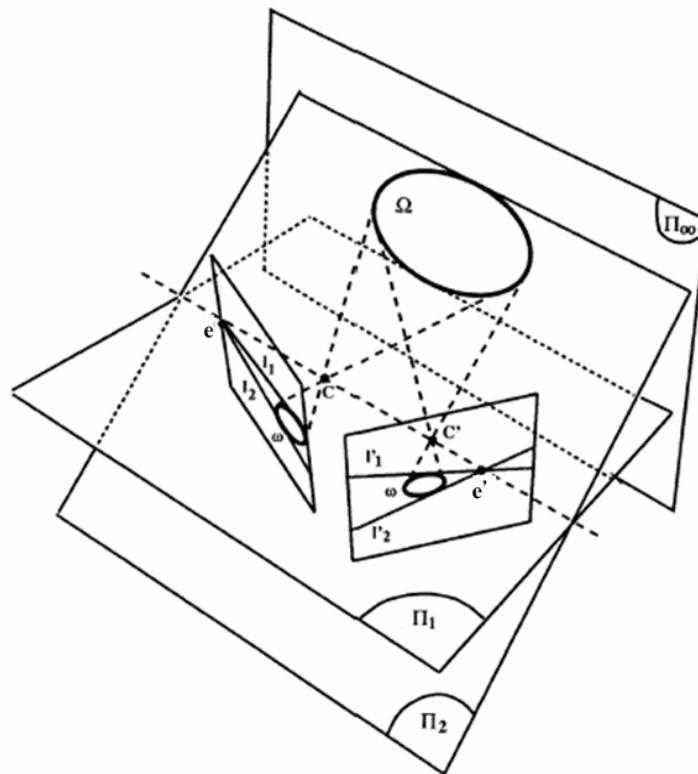


Fig. 2.10 The epipolar tangency to the absolute conic images [18].

In fact, self-calibration techniques are mainly concerned with the intrinsic parameters of cameras. Most self-calibration approaches [14]-[26] were proposed

concerning constant intrinsic parameters. Among these methods, [22]-[26] solve their problem with additional camera motion restrictions. On the other hand, much extended research [27]-[35] has been developed to solve varying intrinsic parameters. Some of them [31]-[35] additionally utilize camera motion constraints to achieve calibration work. The further detail discussions of camera self-calibration approaches can be found in [36].

Although self-calibration has no or fewer assumptions about the camera motion information and doesn't require specific calibration objects, the computational load is heavy and the calibration work is too elaborate to be applied to the dynamic calibration of multiple cameras.



2.3 Dynamic Camera Calibration

Up to now, very few research works [38]-[43] have been proposed for dynamic camera calibration. Most existing dynamic calibration techniques concern with extrinsic parameters of cameras. Jain et al [38] proposed an off-line method, where they tried to find the relationship between the realized rotation angle and the requested angle. In [39], the pose of a calibrated camera is estimated from a planar target. However, both [38] and [39] only demonstrate the dynamic calibration of a single camera, but not the calibration among multiple cameras. In [40], the authors utilize the marks and width of parallel lanes to calibrate PTZ cameras. In [41], the focal length and two external rotations are dynamically estimated by using parallel lanes. Although these two methods are practical for traffic monitoring, it is not general enough for other types of surveillance systems. In [42], a dynamic camera calibration with narrow-range coverage was proposed. For a pair of cameras, this method performs the correspondence of feature points on the image pair and uses coplanar geometry for camera calibration. In [43], the relative pose between a calibrated camera and a projector is determined via plane-based homography. The authors took two steps to recalibrate the pose parameters. They first estimated the translation vector and then found the rotation matrix. They also offered analytic solutions. Nevertheless, this approach requires the correspondence of feature points.

So far as we know, most calibration algorithms require corresponding feature points, special calibration patterns (coplanar points with known structure or parallel lines), or known landmarks in the three dimensional space. However, to dynamically calibrate multiple cameras, calibration patterns and landmarks are not always applicable since they may get occluded or even are out of the captured scenes when cameras pan or tilt. On the other hand, in the correspondence of feature points, we

need to keep updating the correspondence of feature points when cameras rotate. For surveillance systems with a wide-range coverage, the matching of feature points is usually a difficult problem. Hence, in this thesis, we develop a new algorithm for the dynamic calibration of multiple cameras, without the need of a complicated correspondence of feature points.





CHAPTER 3

Static Calibration of Multiple Cameras



In this chapter, we introduce how to efficiently calibrate the extrinsic parameters of multiple static cameras. In Section 3.1, the camera model of our surveillance system is first described. Next, in Section 3.2, we will deduce the 3D-to-2D coordinate transformation in terms of the tilt angle of a camera. In [46], a similar scene model based on pan angle and tilt angle has also been established. In this paper, however, we will deduce a more complete formula that takes into account not only the translation effect but also the rotation effect when a camera is under a tilt movement. After having established the 3D-to-2D transformation, the tilt angle and altitude of a camera can thus be estimated based on the observation of some simple objects lying on a horizontal plane. Then, we will introduce how to utilize the estimation results to achieve the calibration of multiple cameras in Section 3.3. In addition, the sensitivity analysis with respect to parameter fluctuations and measurement errors will be

discussed in Section 3.4. In Section 3.5, some experimental results over real data are demonstrated to illustrate the feasibility of the proposed static calibration method.

3.1 Introduction of Our Camera Model System

In this section, we give a sketch of our system overview, camera setup model and the basic camera projection model. Although the camera model is built based on our surveillance environment, this model is general enough to fit for a large class of surveillance scenes, which are equipped with multiple cameras.

3.1.1 System Overview

In the setup of our indoor surveillance system, four PTZ cameras are mounted on the four corners of the ceiling in our lab, about 3 meters above the ground plane. The lab is full of desks, chairs, PC computers, and monitors. All the tablespots are roughly parallel to the ground plane. These cameras are allowed to pan or tilt while they are monitoring the activities in the room. Figure 3.14(a) shows four images captured by these four cameras. We will first estimate the tilt angle and altitude of each camera based on the captured images of some prominent features, such as corners or line segments, on a horizontal plane. Once the tilt angles and altitudes of these four cameras are individually estimated, we will perform the calibration of multiple cameras. Figure 3.1 shows the flowchart of the proposed static calibration procedure.

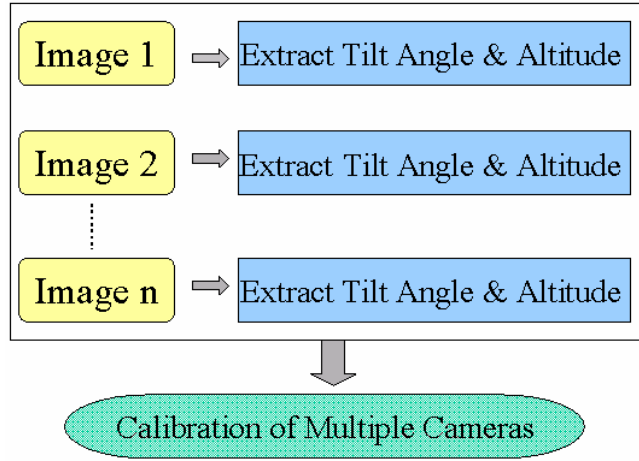


Fig. 3.1 Flowchart of the proposed static calibration procedure.

3.1.2 Camera Setup Model

Figure 3.2 illustrates the modeling of our camera setup. Here, we assume the observed objects are located on a horizontal plane Π , while the camera lies above Π with a height h . The camera may pan or tilt with respect to the rotation center \mathbf{O}_R . Moreover, we assume the projection center of the camera, denoted as \mathbf{O}_C , is away from \mathbf{O}_R with distance r . To simplify the following deductions, we define the origin of the rectified world coordinates to be the projection center \mathbf{O}_C of a camera with zero tilt angle. The Z-axis of the world coordinates is along the optical axis of the camera, while the X- and Y-axis of the world coordinates are parallel to the x- and y-axis of the projected image plane, respectively. When the camera tilts, the projection center moves to \mathbf{O}_C' and the projected image plane is changed to a new 2-D plane. In this case, the y-axis of the image plane is no longer parallel to the Y-axis of the world coordinates, while the x-axis is still parallel to the X-axis.

Assume $P=[X, Y, Z, 1]^T$ denotes the homogeneous coordinates of a 3-D point \mathbf{P} in the world coordinates. For the case of a camera with zero tilt angle, we denote the perspective projection of \mathbf{P} as $p=[x, y, 1]^T$. Under perspective projection, the

relationship between P and p can be expressed as Equation (2.5), $p = \frac{1}{z} K [R \ t] P$.

With respect to the rectified world coordinate system, the extrinsic term $[R \ t]$ becomes $[I \ 0]$. To further simplify the mathematical deduction, we ignore the skew angle and assume the image coordinates have been translated by a translation vector $(-u_0, -v_0)$. Hence, (2.5) can be simplified as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \beta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}, \quad (3.1)$$

or in a reverse way as

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = Z \begin{bmatrix} \frac{1}{\alpha} & 0 & 0 \\ 0 & \frac{1}{\beta} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{xZ}{\alpha} \\ \frac{yZ}{\beta} \\ Z \end{bmatrix}. \quad (3.2)$$

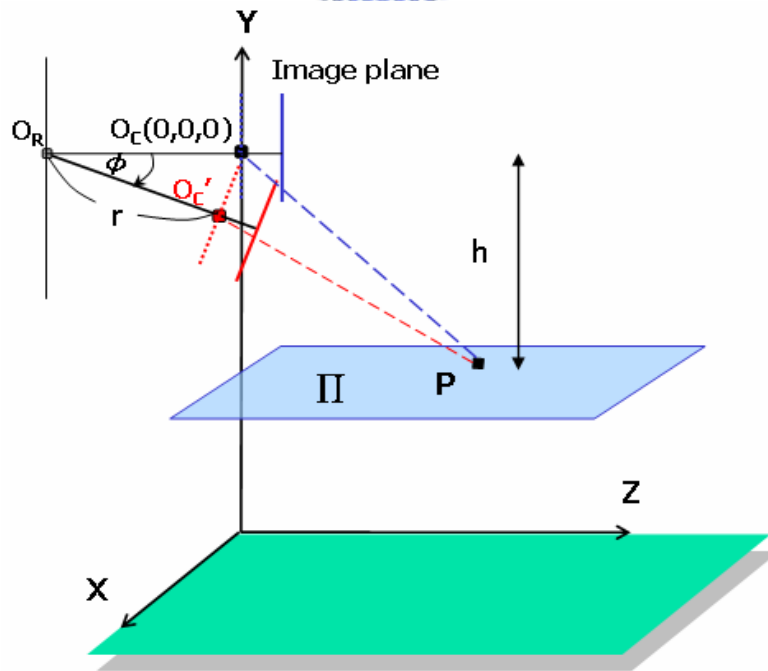


Fig. 3.2 Model of camera setup.

3.2 Pose Estimation of a Single Camera

In this section, we first deduce the projection equation to relate the world coordinates of a 3-D point \mathbf{p} to its image coordinates on a tilted camera. Then, under the constraint that all observed points are located on a horizontal plane, the mapping between the 3-D space and the 2-D image plane is further developed. Finally, we deduce the formulae for the pose estimation of a camera.

3.2.1 Coordinate Mapping on a Tilted Camera

When the PTZ camera tilts with an angle ϕ , the projection center \mathbf{O}_C translates to a new place $\mathbf{O}_{C'}$ with $O_{C'} = [0 \ -r\sin\phi \ -(r-r\cos\phi)]^T$. Assume we define a tilted world coordinate system (X', Y', Z') with respect to the tilted camera, with the origin being the new project center $\mathbf{O}_{C'}$, the Z' -axis being the optical axis of the tilted camera, and the X' - and Y' -axis being parallel to the x' - and y' -axis of the new projected image plane, respectively. Then, it can be easily deduced that in the tilted world coordinate system the coordinates of the 3-D point \mathbf{P} become

$$\begin{aligned} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} X \\ Y + r\sin\phi \\ Z + r(1 - \cos\phi) \end{bmatrix} \\ &= \begin{bmatrix} X \\ Y \cos\phi + Z \sin\phi + r\sin\phi \\ -Y \sin\phi + Z \cos\phi + r(\cos\phi - 1) \end{bmatrix}. \end{aligned} \quad (3.3)$$

After applying the perspective projection formula, we know that the homogeneous coordinates of the projected image point now move to

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \frac{X'}{Z'} \\ \beta \frac{Y'}{Z'} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \frac{X}{-Y \sin \phi + Z \cos \phi + r(\cos \phi - 1)} \\ \beta \frac{Y \cos \phi + Z \sin \phi + r \sin \phi}{-Y \sin \phi + Z \cos \phi + r(\cos \phi - 1)} \\ 1 \end{bmatrix}. \quad (3.4)$$

3.2.2 Constrained Coordinate Mapping

In the rectified world coordinates, all points on a horizontal plane have the same Y coordinate. That is, $Y = -h$ for a constant h . The homogeneous form of this plane Π can be defined as $\pi = [0 \ 1 \ 0 \ h]^T$. Assume the camera is tilted with an angle ϕ . Then, in the tilted world coordinate system, the homogeneous form of this plane Π becomes $\pi' = [0 \ \cos \phi \ -\sin \phi \ (h - r \sin \phi)]^T$, as shown in Fig. 3.3.

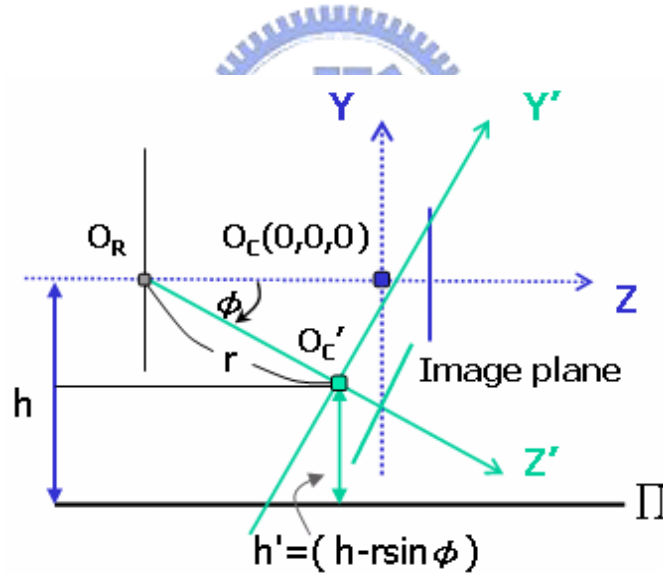


Fig. 3.3 Geometry of a horizontal plane Π with respect to a tilted camera.

Assume a 3-D point p is located on the horizontal plane Π . Then, in the rectified world coordinate system, we have $\pi \cdot P = 0$, where $P = [X, Y, Z, 1]^T$. Similarly, in the tilted world coordinate system, we have $\pi' \cdot P' = 0$, where $P' = [X', Y', Z', 1]^T$. With (3.2), Z' can be found to be

$$Z' = \frac{\beta(r \sin \phi - h)}{y' \cos \phi - \beta \sin \phi}. \quad (3.5)$$

Moreover, the tilted world coordinates of p become

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} \frac{x' \beta(r \sin \phi - h)}{\alpha(y' \cos \phi - \beta \sin \phi)} \\ \frac{y'(r \sin \phi - h)}{y' \cos \phi - \beta \sin \phi} \\ \frac{\beta(r \sin \phi - h)}{y' \cos \phi - \beta \sin \phi} \end{bmatrix}. \quad (3.6)$$

With (3.3) and (3.6), we may transfer $[X', Y', Z']^T$ back to $[X, Y, Z]^T$ to obtain

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{x' \beta(r \sin \phi - h)}{\alpha(y' \cos \phi - \beta \sin \phi)} \\ -h \\ \frac{(y' \sin \phi + \beta \cos \phi)(r \sin \phi - h)}{y' \cos \phi - \beta \sin \phi} - r + r \cos \phi \end{bmatrix}. \quad (3.7)$$

If the principal point (u_0, v_0) is taken into account, then (3.7) can be reformulated as

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{(x' - u_0) \beta(r \sin \phi - h)}{\alpha[(v_0 - y') \cos \phi - \beta \sin \phi]} \\ -h \\ \frac{[(v_0 - y') \sin \phi + \beta \cos \phi](r \sin \phi - h)}{(v_0 - y') \cos \phi - \beta \sin \phi} - r + r \cos \phi \end{bmatrix}. \quad (3.8)$$

This formula indicates the back projection formula from the image coordinates of a tilted camera to the rectified world coordinates, under the constraint that all the observed points are lying on a horizontal plane with $Y = -h$.

3.2.3 Pose Estimation Based on the Back-Projections

As aforementioned, in real life, based on the image contents of a captured image, people can usually have a rough estimate about the relative position of the camera with respect to the captured objects. In this section, we demonstrate that, with a few corners or a few line segments lying on a horizontal plane, we can easily estimate the tilt angle of the camera based on the back projection of the captured image.

3.2.3.1 Back-projected Angle w.r.t. Gussed Tilt Angle

Suppose we use a tilted camera to capture the image of a corner, which is located on a horizontal plane. Based on the captured image and a guessed tilt angle, we may use (3.8) to back-project the captured image onto a horizontal plane on $Y = -h$. Assume three 3-D points, P_A , P_B , and P_C , on a horizontal plane form a rectangular corner at P_A . The original image is captured by a camera with $\phi = 16$ degrees, as shown in Fig. 3.4(a). In Fig. 3.4(b), we plot the back-projected images for various choices of tilt angles. The guessed tilt angles range from 0 to 30 degrees, with a 2-degree step. The back-projection for the choice of 16° is plotted in red, specifically. It can be seen that the back-projected corner becomes a rectangular corner only if the guessed tilt angle is correct. Besides, it is worth mentioning that a different choice of h only causes a scaling effect of the back-projected shape.

To formulate this example, we express the angle ψ at P_A as

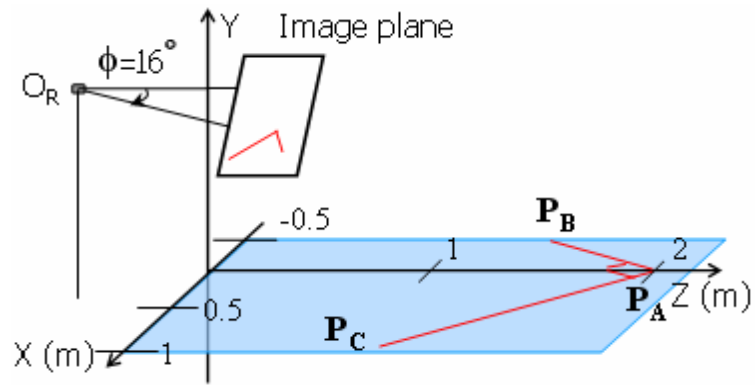
$$\cos \psi = \frac{\langle \overline{P_A P_B}, \overline{P_A P_C} \rangle}{\| \overline{P_A P_B} \| \times \| \overline{P_A P_C} \|}. \quad (3.9)$$

After capturing the image of these three points, we can use (3.8) to build the relation between the back-projected angle and the guessed tilt angle.

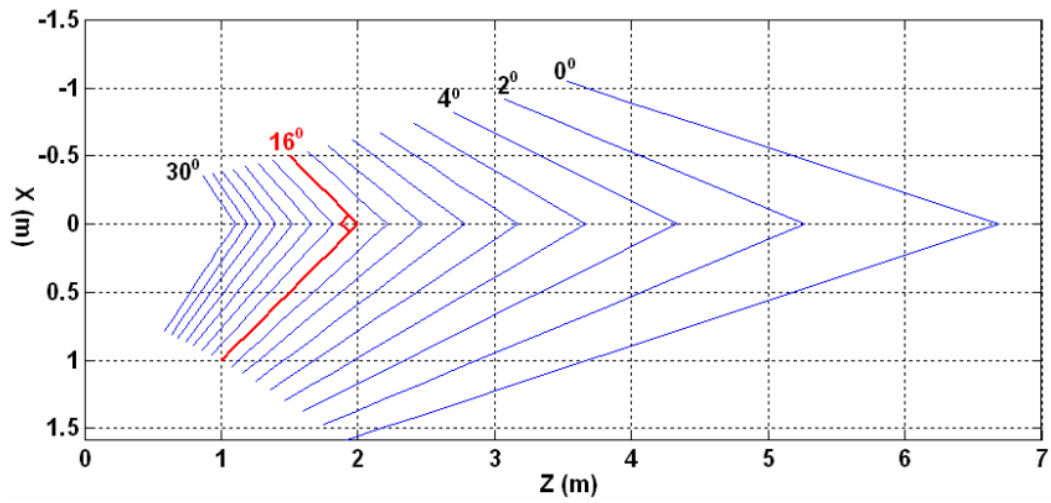
$$\begin{aligned}
\psi = \cos^{-1} \{ & \left(\frac{x'_B \beta}{\alpha(y'_B \cos \phi - \beta \sin \phi)} - \frac{x'_A \beta}{\alpha(y'_A \cos \phi - \beta \sin \phi)} \right) \\
& \times \left(\frac{x'_C \beta}{\alpha(y'_C \cos \phi - \beta \sin \phi)} - \frac{x'_A \beta}{\alpha(y'_A \cos \phi - \beta \sin \phi)} \right) \\
& + \left(\frac{y'_B \sin \phi + \beta \cos \phi}{y'_B \cos \phi - \beta \sin \phi} - \frac{y'_A \sin \phi + \beta \cos \phi}{y'_A \cos \phi - \beta \sin \phi} \right) \\
& \times \left(\frac{y'_C \sin \phi + \beta \cos \phi}{y'_C \cos \phi - \beta \sin \phi} - \frac{y'_A \sin \phi + \beta \cos \phi}{y'_A \cos \phi - \beta \sin \phi} \right) \} \\
& \times \left\{ \left(\frac{x'_B \beta}{\alpha(y'_B \cos \phi - \beta \sin \phi)} - \frac{x'_A \beta}{\alpha(y'_A \cos \phi - \beta \sin \phi)} \right)^2 \right. \\
& \left. + \left(\frac{y'_B \sin \phi + \beta \cos \phi}{y'_B \cos \phi - \beta \sin \phi} - \frac{y'_A \sin \phi + \beta \cos \phi}{y'_A \cos \phi - \beta \sin \phi} \right)^2 \right\}^{-1/2} \\
& \times \left\{ \left(\frac{x'_C \beta}{\alpha(y'_C \cos \phi - \beta \sin \phi)} - \frac{x'_A \beta}{\alpha(y'_A \cos \phi - \beta \sin \phi)} \right)^2 \right. \\
& \left. + \left(\frac{y'_C \sin \phi + \beta \cos \phi}{y'_C \cos \phi - \beta \sin \phi} - \frac{y'_A \sin \phi + \beta \cos \phi}{y'_A \cos \phi - \beta \sin \phi} \right)^2 \right\}^{-1/2} \}
\end{aligned} \tag{3.10}$$

Note that in (3.10) we have ignored the offset terms, u_0 and v_0 , to reduce the complexity of the formulation.

In Fig. 3.5, we show the back-projected angle ψ with respect to the guessed tilt angle, assuming α and β are known in advance. In this simulation, the red and blue curves are generated by placing the rectangular corner on two different places of the horizontal plane. Again, the back-projected angle is equal to 90 degrees only if we choose the tilt angle to be 16 degrees. This simulation demonstrates that if we know in advance the angle of the captured corner, we can easily deduce camera's tilt angle. Moreover, the red curve and the blue curve intersect at $(\phi, \psi) = (16, 90)$. This means that if we don't know in advance the actual angle of the corner, we can simply place that corner on more than two different places of the horizontal plane. Then, based on the intersection of the deduced ψ -v.s.- ϕ curves, we may not only estimate the tilt angle of the camera but also the actual angle of the corner.



(a)



(b)

Fig. 3.4 (a) Rectangular corner captured by a tilted camera (b) Illustration of back-projection onto a horizontal plane on for different choices of tilt angles.

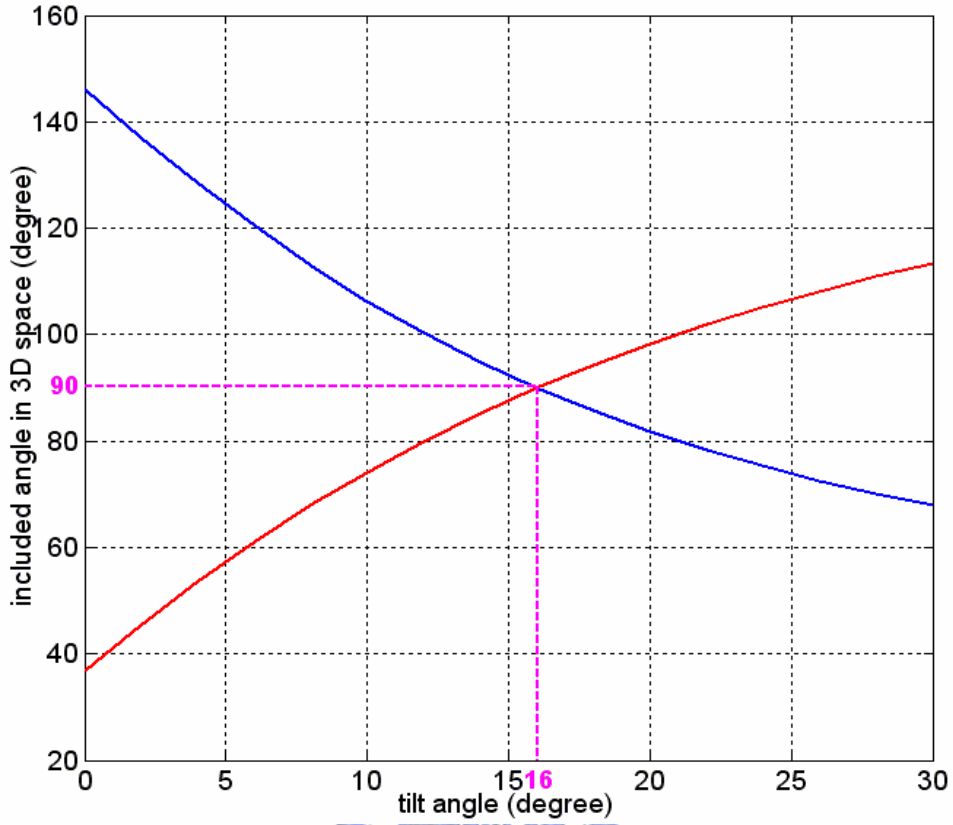


Fig. 3.5 Back-projected angle with respect to guessed tilt angles.

3.2.3.2 Back-projected Length w.r.t. Guessed Tilt Angle

Assume two 3-D points, P_A and P_B , on a horizontal plane form a line segment with length L . Similarly, we can build a similar relationship between the back-projected length and the guessed tilt angle by setting the constraint: $\|\overline{P_A P_B}\| = L$. Based on this

constraint and (3.8), we can deduce that

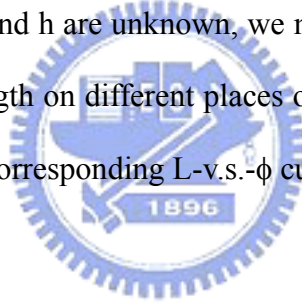
$$L = \ell(\phi) = \left\{ \left(\frac{(x'_B - u_0)\beta(r \sin \phi - h)}{\alpha[(v_0 - y'_B) \cos \phi - \beta \sin \phi]} - \frac{(x'_A - u_0)\beta(r \sin \phi - h)}{\alpha[(v_0 - y'_A) \cos \phi - \beta \sin \phi]} \right)^2 + \left(\frac{[(v_0 - y'_B) \sin \phi + \beta \cos \phi](r \sin \phi - h)}{(v_0 - y'_B) \cos \phi - \beta \sin \phi} - \frac{[(v_0 - y'_A) \sin \phi + \beta \cos \phi](r \sin \phi - h)}{(v_0 - y'_A) \cos \phi - \beta \sin \phi} \right)^2 \right\}^{\frac{1}{2}}. \quad (3.11)$$

Similarly, if α , β , r , and h are known in advance, we can deduce the tilt angle directly based on the projected value of L .

Note that in (3.11), the right-side terms contain a common factor $(r\sin\phi-h)^2$. This means the values of r and h only affect the scaling of L . Hence, we can rewrite the formula of the L -v.s.- ϕ curve as

$$\begin{aligned}
 L' &\equiv \frac{L}{r\sin\phi-h} \\
 &= \left\{ \frac{(x'_B - u_0)\beta}{\alpha[(v_0 - y'_B)\cos\phi - \beta\sin\phi]} - \frac{(x'_A - u_0)\beta}{\alpha[(v_0 - y'_A)\cos\phi - \beta\sin\phi]} \right\}^2 \\
 &\quad + \left\{ \frac{[(v_0 - y'_B)\sin\phi + \beta\cos\phi]}{(v_0 - y'_B)\cos\phi - \beta\sin\phi} - \frac{[(v_0 - y'_A)\sin\phi + \beta\cos\phi]}{(v_0 - y'_A)\cos\phi - \beta\sin\phi} \right\}^2 \frac{1}{2}.
 \end{aligned} \tag{3.12}$$

Then, even if the values of r and h are unknown, we may simply place more than two line segments of the same length on different places of a horizontal plane and seek to find the intersection of these corresponding L -v.s.- ϕ curves, as shown in Fig. 3.6.



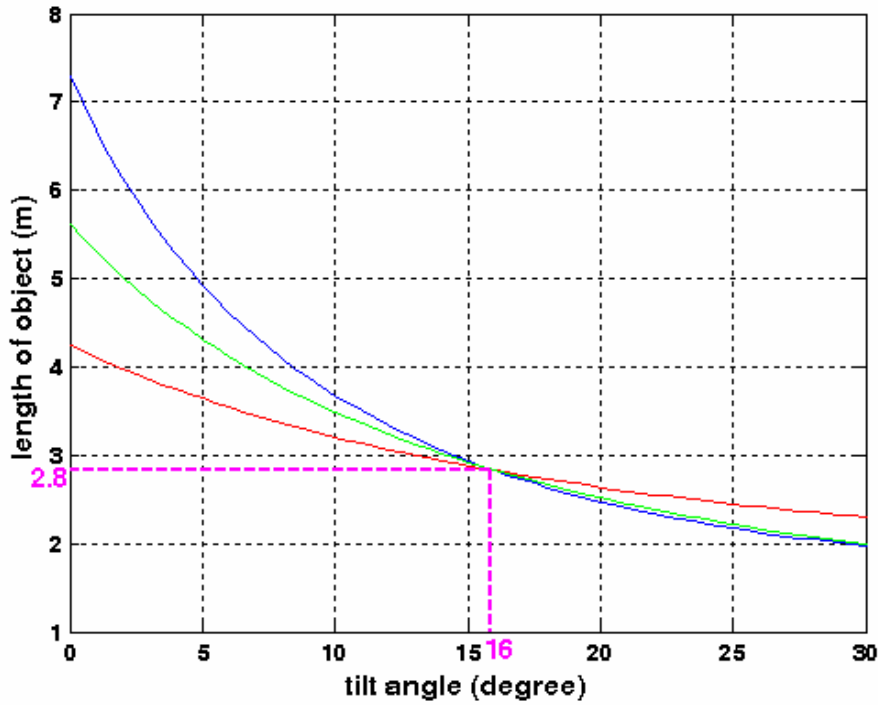


Fig. 3.6 Back-projected length with respect to guessed tilt angle. Each curve is generated by placing a line segment on some place of a horizontal plane.

As mentioned above, the tilt angle can be easily estimated from the ψ -v.s.- ϕ curves or L-v.s.- ϕ curves. However, in practice, due to errors in the estimation of camera parameters and errors in the measurement of (x', y') coordinates, the deduced ψ -v.s.- ϕ curves or L-v.s.- ϕ curves do not intersect at a single point. Hence, we may also seek to perform parameter estimation based on an optimization process. Here, we take (3.11) as an example. We assume several line segments with known lengths (not necessary of the same length) are placed on different positions of a horizontal plane and we use a tilted camera to capture the image. Assume the length of the i th segment is L_i , then we aim to find a set of parameters $\{\alpha, \beta, u_0, v_0, \phi, r, h\}$ that minimize

$$\begin{aligned}
 & F(x'_1, y'_1, x'_2, y'_2, \dots, x'_m, y'_m, \alpha, \beta, u_0, v_0, \phi, r, h) \\
 & = \sum_{i=1}^m \|\ell_i(x'_i, y'_i, \alpha, \beta, u_0, v_0, \phi, r, h) - L_i\|^2.
 \end{aligned} \tag{3.13}$$

In this way, camera parameters can also be easily estimated. In the optimization process, we adopt the Levenberg-Marquardt algorithm. Under our scene model, the tilt angle ϕ and altitude h can be roughly estimated simply based on visual observations. In our experiments, the error range for the guessed tilt angle ϕ is within ± 20 degrees and the error range for the guessed altitude h is within ± 1.5 meters. With these initial guesses, the optimization process is very stable and the estimation results are satisfactorily accurate.

3.3 Calibration of Multiple Static Cameras

3.3.1 Static Calibration Method of Multiple Cameras

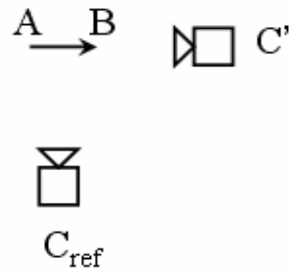
In our camera model, each camera has its own world coordinate system. If a vector in the 3-D space, like a line segment on a tabletop, is observed by several cameras at the same time, we can achieve the calibration of these cameras by mapping the individual back-projected world coordinates of this vector to a common reference world coordinates. In Fig. 3.7, we take two calibrated cameras as an example. Fig. 3.7(a) shows the scene model of these two cameras. Fig. 3.7(b) shows the vector locations in the world coordinates of these two cameras, respectively. Based on the estimated ϕ and h , and the image projections of the vector points, we can get the world coordinates of points \mathbf{A}_{ref} , \mathbf{B}_{ref} , and \mathbf{A}' , \mathbf{B}' from (3.8). The difference of the rotation angle ω between the two world coordinate systems can then be easily computed by

$$\cos \omega = \frac{\langle \overline{A'B'}, \overline{A_{\text{ref}}B_{\text{ref}}} \rangle}{\| \overline{A'B'} \| \times \| \overline{A_{\text{ref}}B_{\text{ref}}} \|}. \quad (3.14)$$

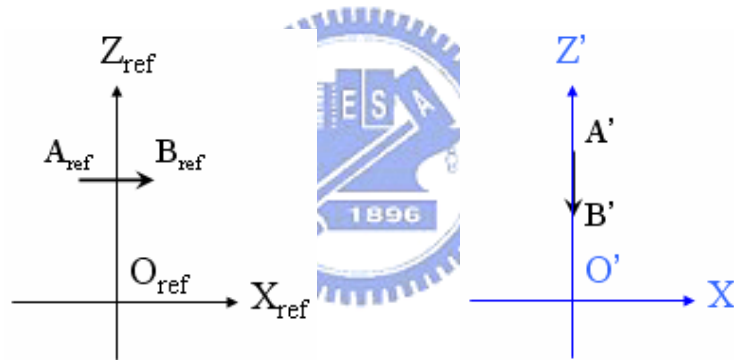
After applying the rotation to point \mathbf{A}' , the position translation t between these two cameras can be expressed as

$$\mathbf{t} = \mathbf{A}_{\text{ref}} - \begin{bmatrix} \cos \omega & 0 & -\sin \omega \\ 0 & 1 & 0 \\ \sin \omega & 0 & \cos \omega \end{bmatrix} \cdot \mathbf{A}' \quad (3.15)$$

Hence, the 3-D relationship between these two cameras can be easily deduced.



(a)

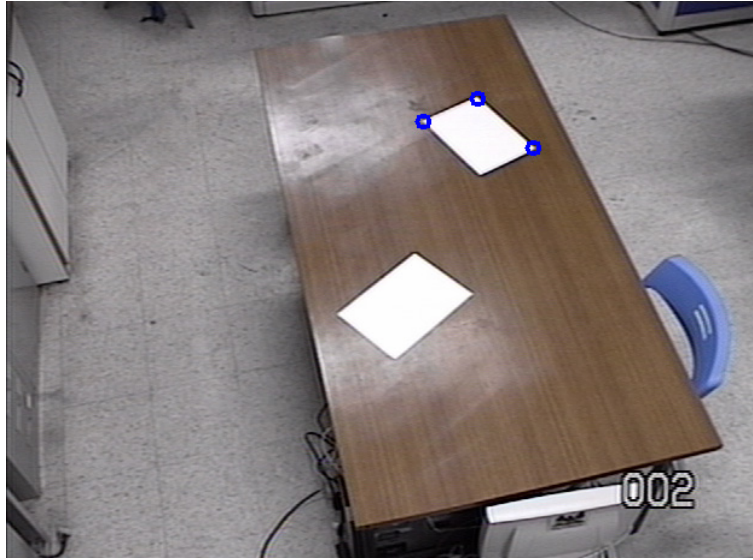


(b)

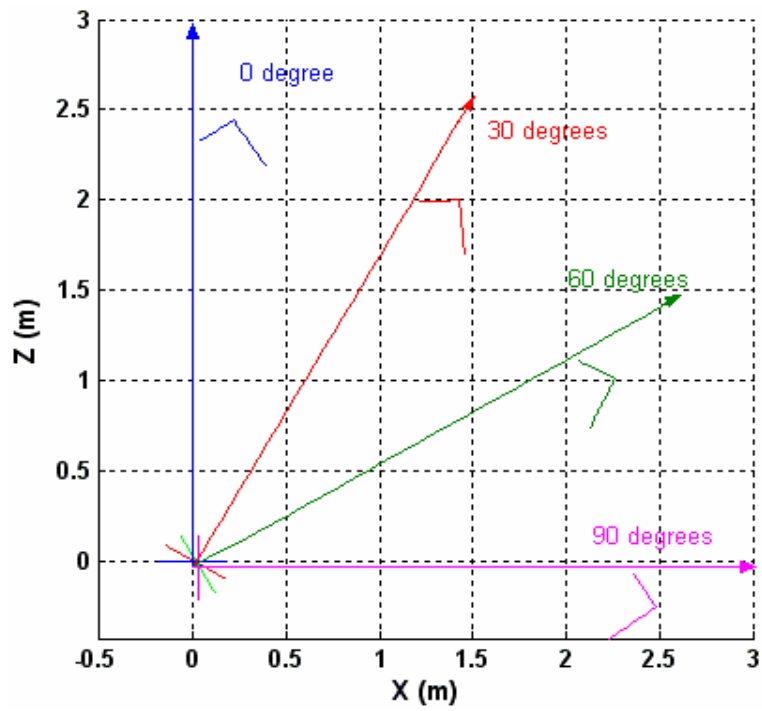
Fig. 3.7 (a) Top view of two cameras and a vector in the space (b) The world coordinates of the vector with respect to these two cameras.

3.3.2 Discussion of Pan Angle

Notice that, in the above deductions, we don't care about the pan angles of cameras. This is because, in the back-projection process, to guess a pan angle only implies to rotate the X and Z coordinates in our camera model. It does not change the space relationship between the camera and the back-projected objects. In Fig. 3.8 we show such an example. In Fig. 3.8(a), we show the image captured by a camera with three corner points being marked in blue. Fig. 3.8(b) shows the top view (i.e. X-Z plane) of the back-projected corner points with respect to four guessed pan angles, 0° , 30° , 60° , and 90° . The arrows indicate the optical axes of the camera with respect to these four pan angles. It can be seen that the space relationship between the optical axis and the back-projected corner points is almost the same when the camera pans. The little variation comes from the fact that the panning center is not the same as the projection center. However, since the rotation radius r is so small if compared with the distance between the camera and the object, this small variation can actually be ignored.



(a)



(b)

Fig. 3.8 (a) Three points marked in the image captured by a PTZ camera (b) Top view of the back-projected corners and the optical axes with respect to different guessed pan angles.

3.4 Sensitivity Analysis

The tilt angle of a camera is the key factor in our multi-camera calibration method. It affects the back-projections of image coordinates, the information utilized to calibrate cameras. In this section, we try to analyze how sensitive the estimation of tilt angle is with respect to parameter fluctuations and measurement errors. Among these parameters, the distance r between the camera center and the rotation center has no impact over (3.10) and (3.12). Even in (3.11), r tends to have negligible impact since the term $r\sin\phi$ is usually much smaller than h . Hence, the parameter r can be ignored or be estimated via direct measurement. This means (3.13) can be reformulated as

$$\begin{aligned} & F(x'_1, y'_1, x'_2, y'_2, \dots, x'_m, y'_m, \alpha, \beta, u_0, v_0, \phi, h) \\ &= \sum_{i=1}^m \|\ell_i(x'_i, y'_i, \alpha, \beta, u_0, v_0, \phi, h) - L_i\|^2. \end{aligned} \quad (3.16)$$

Besides, several parameters tangle together in a fairly complicated way in the formulae relating the back-projected angle and length with respect to the guessed tilt angle. Hence, we figure out the variations of the tilt angle caused by several parameters via computer simulations in addition. In this section, the values of $\{u_0, v_0, \alpha, \beta\}$ are estimated to be $\{348, 257, 770, 750\}$ based on Zhang's calibration method [4].

3.4.1 Mathematical Analysis of Sensitivity

We assume the total variation of ϕ or h is the summation of the individual variation with respect to different parameter fluctuation. It leads to equation (3.17) as follows,

$$\begin{cases}
\Delta\phi = \left(\frac{\partial\phi}{\partial x'_i} \Delta x'_i + \frac{1}{2} \frac{\partial^2\phi}{\partial^2 x'_i} (\Delta x'_i)^2 + R(\Delta x'_i)_n^\phi \right) + \left(\frac{\partial\phi}{\partial y'_i} \Delta y'_i + \frac{1}{2} \frac{\partial^2\phi}{\partial^2 y'_i} (\Delta y'_i)^2 + R(\Delta y'_i)_n^\phi \right) \\
+ \left(\frac{\partial\phi}{\partial\alpha} \Delta\alpha + \frac{1}{2} \frac{\partial^2\phi}{\partial^2\alpha} (\Delta\alpha)^2 + R(\Delta\alpha)_n^\phi \right) + \left(\frac{\partial\phi}{\partial\beta} \Delta\beta + \frac{1}{2} \frac{\partial^2\phi}{\partial^2\beta} (\Delta\beta)^2 + R(\Delta\beta)_n^\phi \right) \\
+ \left(\frac{\partial\phi}{\partial u_0} \Delta u_0 + \frac{1}{2} \frac{\partial^2\phi}{\partial^2 u_0} (\Delta u_0)^2 + R(\Delta u_0)_n^\phi \right) + \left(\frac{\partial\phi}{\partial v_0} \Delta v_0 + \frac{1}{2} \frac{\partial^2\phi}{\partial^2 v_0} (\Delta v_0)^2 + R(\Delta v_0)_n^\phi \right) \\
\Delta h = \left(\frac{\partial h}{\partial x'_i} \Delta x'_i + \frac{1}{2} \frac{\partial^2 h}{\partial^2 x'_i} (\Delta x'_i)^2 + R(\Delta x'_i)_n^h \right) + \left(\frac{\partial h}{\partial y'_i} \Delta y'_i + \frac{1}{2} \frac{\partial^2 h}{\partial^2 y'_i} (\Delta y'_i)^2 + R(\Delta y'_i)_n^h \right) \\
+ \left(\frac{\partial h}{\partial\alpha} \Delta\alpha + \frac{1}{2} \frac{\partial^2 h}{\partial^2\alpha} (\Delta\alpha)^2 + R(\Delta\alpha)_n^h \right) + \left(\frac{\partial h}{\partial\beta} \Delta\beta + \frac{1}{2} \frac{\partial^2 h}{\partial^2\beta} (\Delta\beta)^2 + R(\Delta\beta)_n^h \right) \\
+ \left(\frac{\partial h}{\partial u_0} \Delta u_0 + \frac{1}{2} \frac{\partial^2 h}{\partial^2 u_0} (\Delta u_0)^2 + R(\Delta u_0)_n^h \right) + \left(\frac{\partial h}{\partial v_0} \Delta v_0 + \frac{1}{2} \frac{\partial^2 h}{\partial^2 v_0} (\Delta v_0)^2 + R(\Delta v_0)_n^h \right)
\end{cases} \quad (3.17)$$

where we use the second order of Taylor series to approximate the variations. Next,

the terms of right hand side, such as $\frac{\partial\phi}{\partial x'_i}, \frac{\partial^2\phi}{\partial^2 x'_i}$, etc., are deduced.

For estimations of ϕ and h , the optimization of (3.16) conforms to the following equations:

$$\begin{cases}
\frac{\partial F}{\partial\phi} = 2 \sum_{i=1}^m [\ell_i(x'_i, y'_i, \alpha, \beta, u_0, v_0, \phi, h) - L_i] \frac{\partial\ell_i}{\partial\phi} = 0, \\
\frac{\partial F}{\partial h} = 2 \sum_{i=1}^m [\ell_i(x'_i, y'_i, \alpha, \beta, u_0, v_0, \phi, h) - L_i] \frac{\partial\ell_i}{\partial h} = 0,
\end{cases} \quad (3.18)$$

By eliminating the multiple of 2 in (3.18), we define f_1 and f_2 as

$$\begin{cases}
f_1 = \sum_{i=1}^m \ell_i \frac{\partial\ell_i}{\partial\phi} - \sum_{i=1}^m L_i \frac{\partial\ell_i}{\partial\phi} = 0 \\
f_2 = \sum_{i=1}^m \ell_i \frac{\partial\ell_i}{\partial h} - \sum_{i=1}^m L_i \frac{\partial\ell_i}{\partial h} = 0
\end{cases} \quad (3.19)$$

Hence, the estimated ϕ and h satisfy $f_1(\phi, h) = 0$ and $f_2(\phi, h) = 0$. Now, we apply the implicit function theorem to (3.19) to find how ϕ and h deviate with respect the

measurement error in x_i' . Equation (3.20) is the differential of (3.19) with respect to x_i' by using the chain rule.

$$\begin{cases} \frac{\partial}{\partial x_i'} f_1(x_i', \phi(x_i'), h(x_i')) = \frac{\partial f_1}{\partial x_i'} + \frac{\partial f_1}{\partial \phi} \frac{\partial \phi}{\partial x_i'} + \frac{\partial f_1}{\partial h} \frac{\partial h}{\partial x_i'} = 0 \\ \frac{\partial}{\partial x_i'} f_2(x_i', \phi(x_i'), h(x_i')) = \frac{\partial f_2}{\partial x_i'} + \frac{\partial f_2}{\partial \phi} \frac{\partial \phi}{\partial x_i'} + \frac{\partial f_2}{\partial h} \frac{\partial h}{\partial x_i'} = 0 \end{cases} \quad (3.20)$$

The matrix form of (3.20) is

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_i'} \\ \frac{\partial f_2}{\partial x_i'} \end{bmatrix} + \begin{bmatrix} \frac{\partial f_1}{\partial \phi} & \frac{\partial f_1}{\partial h} \\ \frac{\partial f_2}{\partial \phi} & \frac{\partial f_2}{\partial h} \end{bmatrix} \begin{bmatrix} \frac{\partial \phi}{\partial x_i'} \\ \frac{\partial h}{\partial x_i'} \end{bmatrix} = 0. \quad (3.21)$$

Then we can obtain

$$\begin{bmatrix} \frac{\partial \phi}{\partial x_i'} \\ \frac{\partial h}{\partial x_i'} \end{bmatrix} = - \begin{bmatrix} \frac{\partial f_1}{\partial \phi} & \frac{\partial f_1}{\partial h} \\ \frac{\partial f_2}{\partial \phi} & \frac{\partial f_2}{\partial h} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial f_1}{\partial x_i'} \\ \frac{\partial f_2}{\partial x_i'} \end{bmatrix}. \quad (3.22)$$

If we assume the total variations of ϕ and h are caused by individual variations with respect to parameter fluctuations in $\{\alpha, \beta, u_0, v_0\}$ and measurement errors in $\{x_i', y_i'\}$,

then we have

$$\begin{aligned} \Delta \phi \approx & \sum_{i=1}^m \left(\frac{\partial \phi}{\partial x_i'} \Delta x_i' \right) + \sum_{i=1}^m \left(\frac{\partial \phi}{\partial y_i'} \Delta y_i' \right) + \frac{\partial \phi}{\partial \alpha} \Delta \alpha + \frac{\partial \phi}{\partial \beta} \Delta \beta \\ & + \frac{\partial \phi}{\partial u_0} \Delta u_0 + \frac{\partial \phi}{\partial v_0} \Delta v_0 \end{aligned} \quad (3.23)$$

and

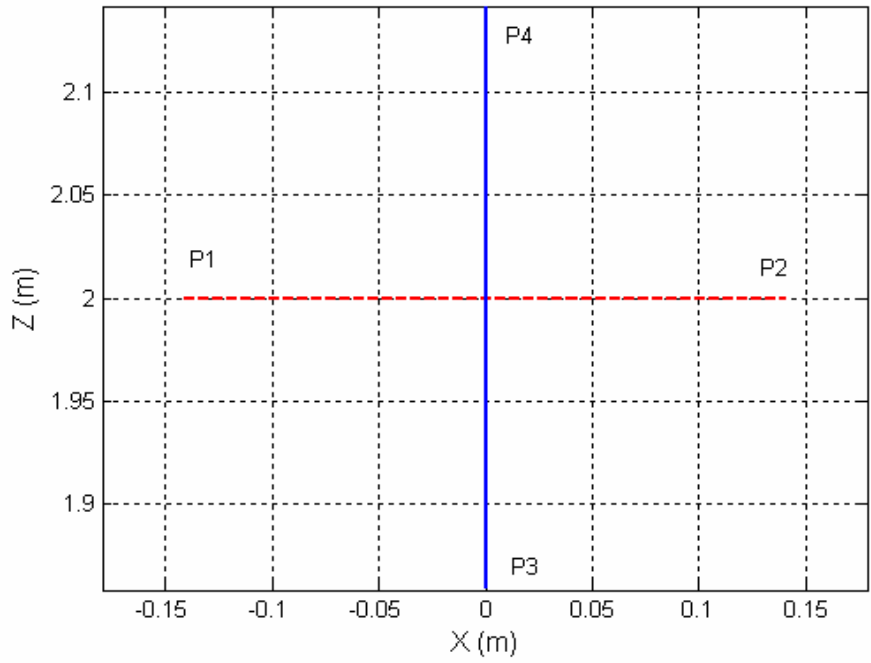
$$\begin{aligned} \Delta h \approx & \sum_{i=1}^m \left(\frac{\partial h}{\partial x_i'} \Delta x_i' \right) + \sum_{i=1}^m \left(\frac{\partial h}{\partial y_i'} \Delta y_i' \right) + \frac{\partial h}{\partial \alpha} \Delta \alpha + \frac{\partial h}{\partial \beta} \Delta \beta \\ & + \frac{\partial h}{\partial u_0} \Delta u_0 + \frac{\partial h}{\partial v_0} \Delta v_0. \end{aligned} \quad (3.24)$$

Note that we ignore the second differential terms because the variations can be approximated by only the first differential terms appropriately.

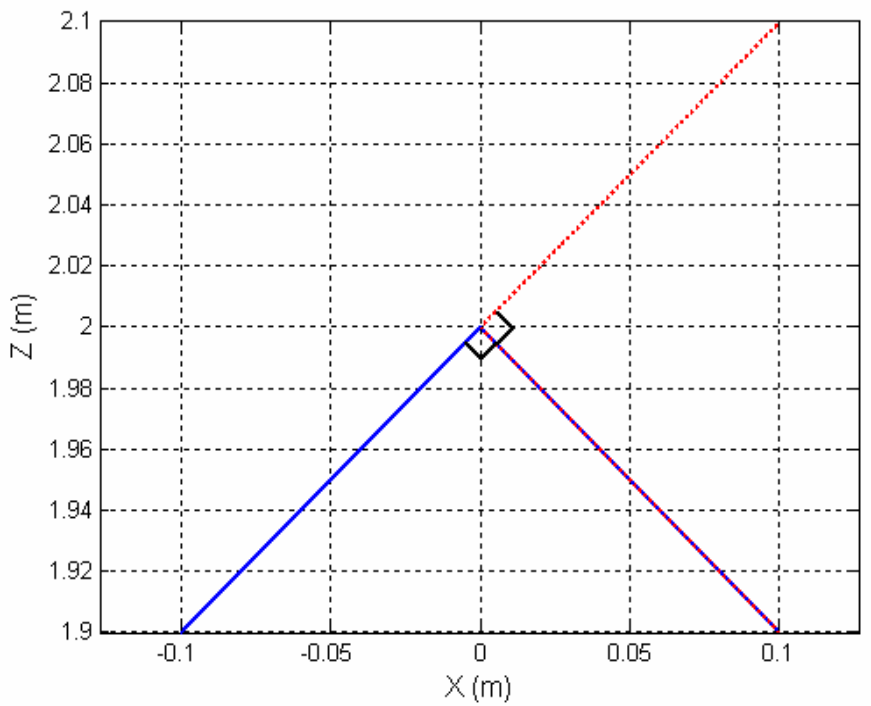
3.4.2 Sensitivity Analysis via Computer Simulations

We verify the prediction of the variation deduced in the previous sub-section via computer simulations. Ideally, when we place several corners or line segments on a horizontal plane, the deduced ψ -v.s.- ϕ curves or L-v.s.- ϕ curves should intersect at a single point. However, due to errors in the estimation of camera parameters and errors in the measurement of (x_i', y_i') , these curves usually do not intersect at a single point. Hence, in practice, we estimate ϕ and h based on (3.16) by using the optimization of Levenberg-Marquardt algorithm.

In the following simulations, two line segments are placed as Fig. 9(a). The lengths are both equal to 0.283 meters. Moreover, the images are assumed to be captured by a camera with tilt angle $\phi = 60$ degrees. We change the values of camera parameters and the measurement of (x_i', y_i') individually. Again, ϕ and h are estimated via LM optimization. The variations of these estimation results, together with the variations deduced by (3.23) and (3.24) are listed in Table 3.1. It can be seen that the deduced variations based on (3.23) and (3.24) well approximate the simulation results. Besides, we figure out L-v.s.- ϕ curves and apply the simulation results to ψ -v.s.- ϕ curves. For ψ -v.s.- ϕ curves, the corners are placed as Fig. 9(b). We can find that the variations of tilt angle in the ψ -v.s.- ϕ figures match those in the L-v.s.- ϕ figures.



(a)



(b)

Fig. 3.9 (a) Top view of line segments placed on a horizontal plane (b) Top view of corners placed on a horizontal plane.

3.4.2.1 Sensitivity w.r.t. u_0 and v_0

As indicated in (3.12), the parameter u_0 only affects the numerator part of the back-projected X coordinate. Since the calculations of ψ and L depend on the distance between back-projected points, but not the absolute positions, this parameter has little impact over the deduced ψ -v.s.- ϕ curves and L -v.s.- ϕ curves. Even if the value of u_0 is changed by an amount of 100, the deduced tilt angle only changes about 0.1 degrees. On the other hand, the parameter v_0 has a larger, but still acceptable impact over the estimation of tilt angle. In Fig. 3.10, we plot the deduced ψ -v.s.- ϕ curves and L -v.s.- ϕ curves for the example in Fig. 3.9, with respect to different choices of v_0 . It can be seen in Fig. 3.10 that a change of ± 20 pixels in v_0 may cause only a 1-degree deviation in the estimated tilt angle.

3.4.2.2 Sensitivity w.r.t. α and β

Similarly, as the value α is changed by the amount of ± 20 , the estimated tilt angle is found to have a ± 1.5 -degree fluctuation. On the other hand, as the value β is changed by the amount of ± 20 , the estimated tilt angle is found to have a ± 2 -degree fluctuation, as shown in Fig. 3.11.

3.4.2.3 Sensitivity w.r.t. $\Delta x_i'$ and $\Delta y_i'$

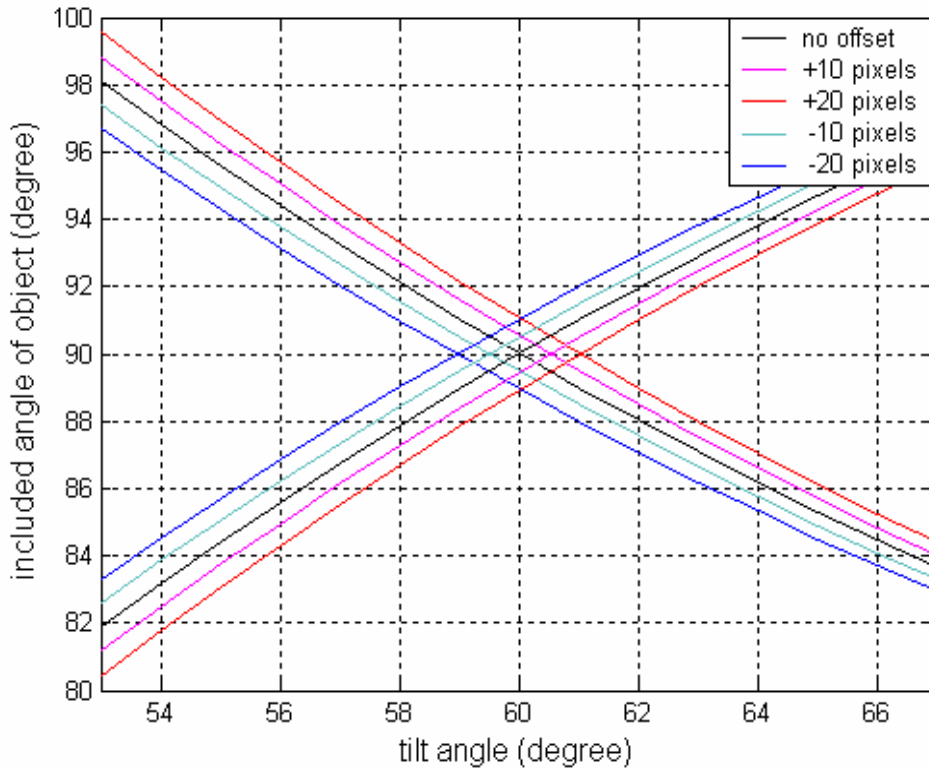
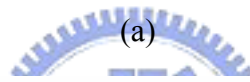
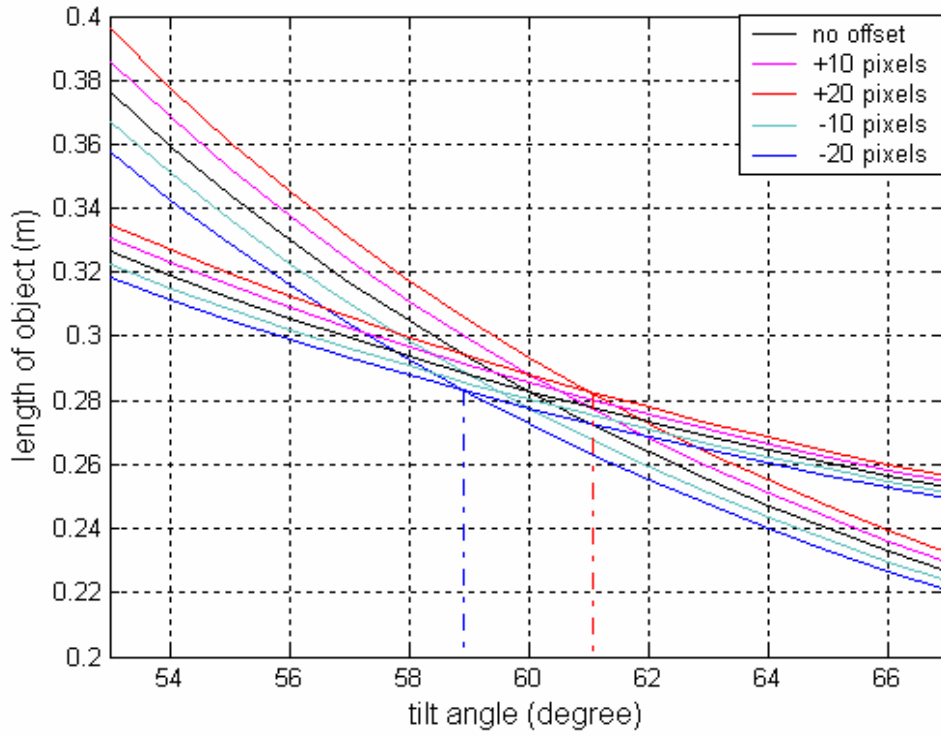
The values of x_i' and y_i' may also affect the construction of ψ -v.s.- ϕ curves and L -v.s.- ϕ curves. As shown in Fig. 3.12, a fluctuation of ± 4 pixels in x_1' or y_3' (x' coordinate of P_1 or y' coordinate of P_3 shown in Fig. 3.9(a)) causes the estimated tilt angle to change by the amount of ± 3 - or ± 4 -degree, respectively. In practice, the fluctuations of x_i' and y_i' are likely to be much less than 4 pixels. Hence, the estimation error caused by the measurement error of x_i' and y_i' is expected to be smaller than 3 or 4 degrees.

3.4.2.4 Sensitivity w.r.t. different choices of tilt angle

In the second simulation, we change the tilt angle ϕ from 15 to 75 degrees, with a 15-degree step. Table 3.2 demonstrates that (3.23) and (3.24) conform to the variations of the simulation results for a wide range of tilt angle ϕ . In practice, the fluctuations of camera parameters are likely to be less than 20 and the measurement errors of (x_i', y_i') are likely to be less than 4 pixels. Hence, the estimation errors of ϕ and h are expected to be acceptable in real cases.

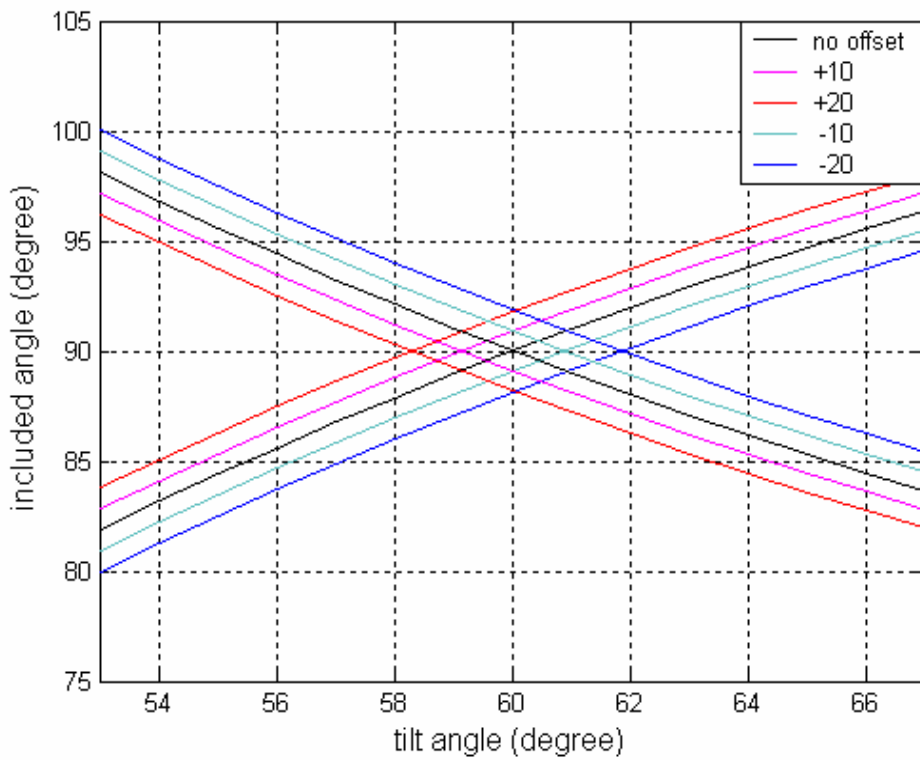
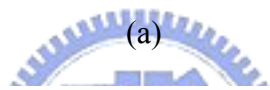
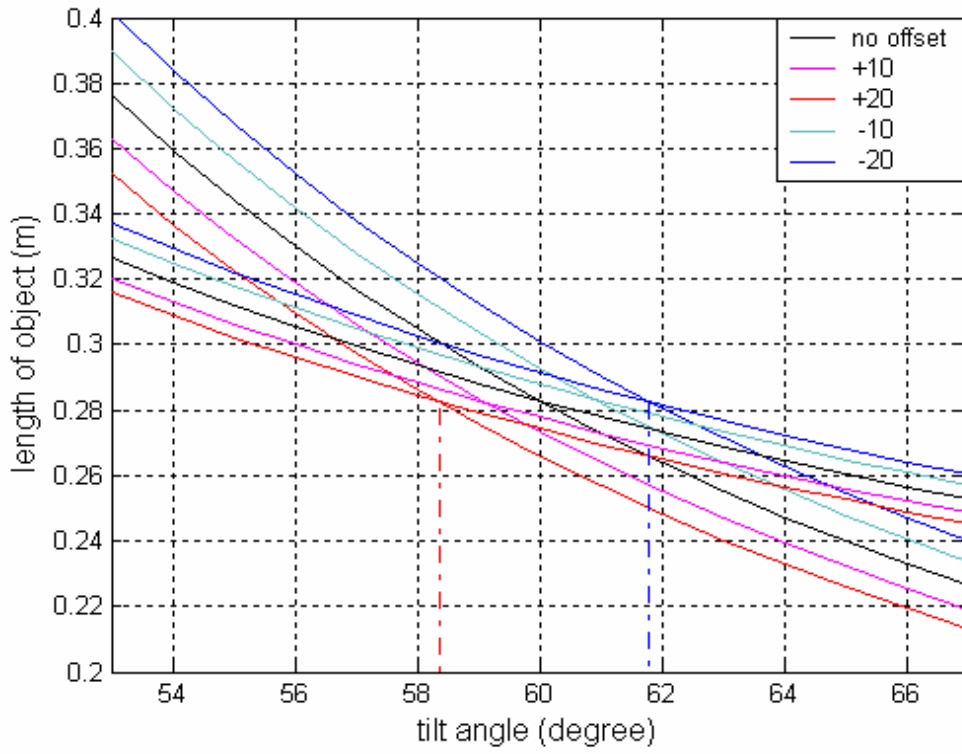
Table 3.1 Variations of Tilt Angle and Altitude with respect to Different Parameter Fluctuations and Measurement Errors.

Δu_0	-20	-10	+10	+20
simulated $(\Delta \phi, \Delta h)$	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)
deduced $(\Delta \phi, \Delta h)$	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)
Δv_0	-20	-10	+10	+20
simulated $(\Delta \phi, \Delta h)$	(-1.05, 0.00)	(-0.52, 0.00)	(0.51, 0.00)	(1.00, 0.00)
deduced $(\Delta \phi, \Delta h)$	(-1.02, 0.00)	(-0.51, 0.00)	(0.51, 0.00)	(1.02, 0.00)
$\Delta \alpha$	-20	-10	+10	+20
simulated $(\Delta \phi, \Delta h)$	(-1.44, -0.10)	(-0.72, -0.05)	(0.73, 0.05)	(1.47, 0.10)
deduced $(\Delta \phi, \Delta h)$	(-1.48, -0.11)	(-0.74, -0.05)	(0.72, 0.05)	(1.41, 0.10)
$\Delta \beta$	-20	-10	+10	+20
simulated $(\Delta \phi, \Delta h)$	(1.85, 0.05)	(0.91, 0.03)	(-0.88, -0.03)	(-1.72, -0.05)
deduced $(\Delta \phi, \Delta h)$	(1.72, 0.05)	(0.89, 0.03)	(-0.92, -0.03)	(-1.89, -0.06)
$\Delta x_1'$	4	-2	+2	+4
simulated $(\Delta \phi, \Delta h)$	(-2.60, -0.18)	(-1.34, -0.09)	(1.45, 0.1)	(3.01, 0.21)
deduced $(\Delta \phi, \Delta h)$	(-2.84, -0.20)	(-1.41, -0.10)	(1.39, 0.1)	(2.73, 0.19)
$\Delta y_3'$	4	-2	+2	+4
simulated $(\Delta \phi, \Delta h)$	(-3.62, -0.13)	(-1.94, -0.07)	(1.91, 0.07)	(3.90, 0.13)
deduced $(\Delta \phi, \Delta h)$	(-4.45, -0.17)	(-1.97, -0.07)	(1.78, 0.06)	(3.25, 0.11)



(b)

Fig. 3.10 Variations of the (a) L-v.s.- ϕ curves and (b) ψ -v.s.- ϕ curves with respect to the variation of v_0 .



(b)

Fig. 3.11 Variations of the (a) L -v.s.- ϕ curves and (b) ψ -v.s.- ϕ curves with respect to the variation of β .

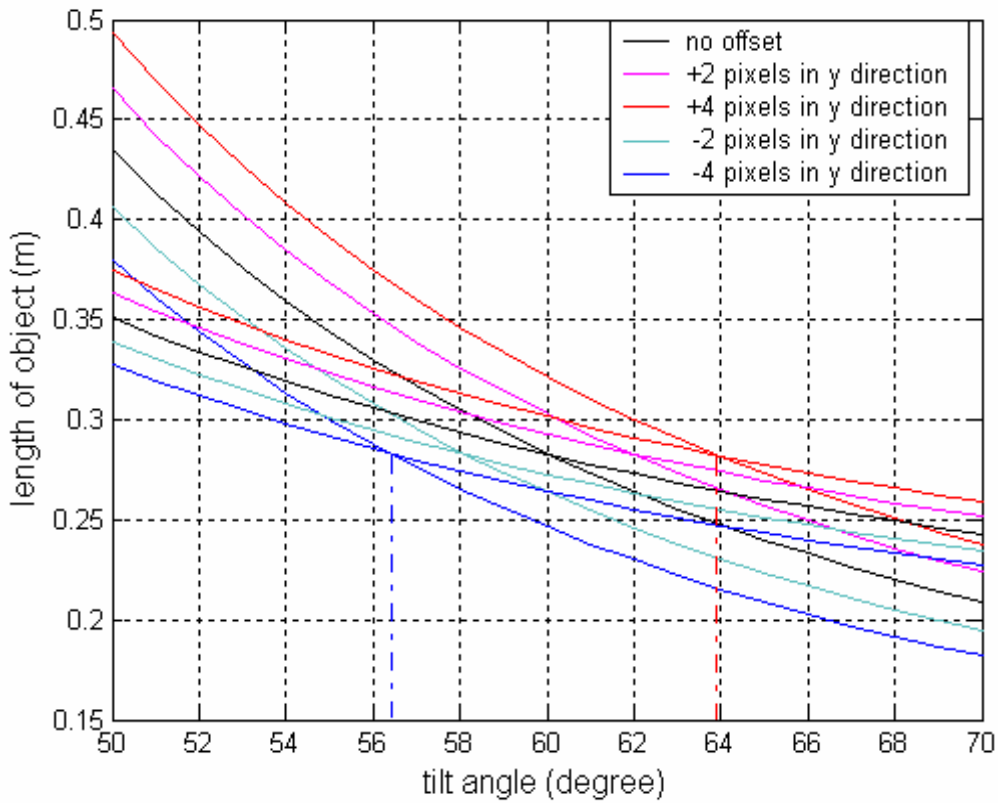
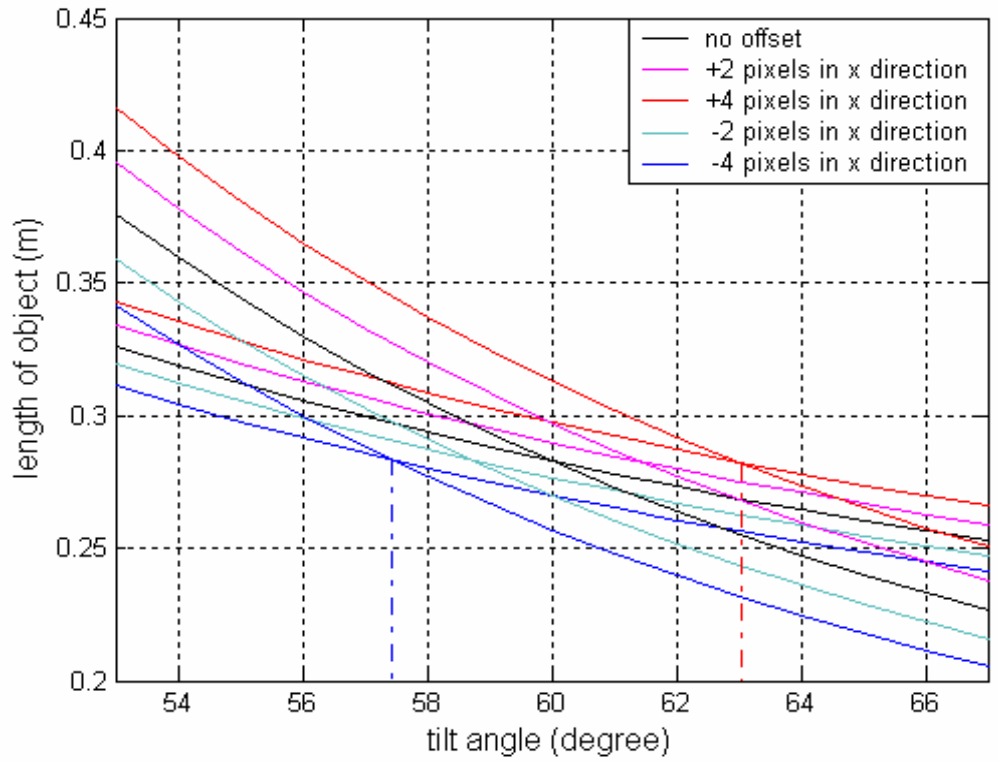


Fig. 3.12. Sensitivity w.r.t. fluctuations of x_i' and y_i' .

Table 3.2 Variations of Tilt Angle and Altitude with respect to Different Choices of Tilt Angle.

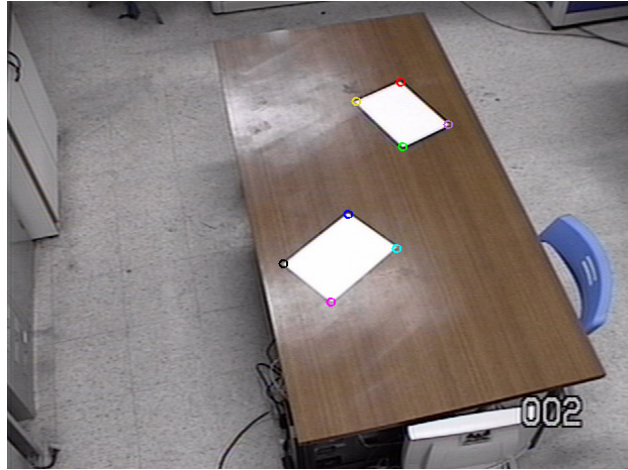
	$(\Delta\phi, \Delta h)$	15°	30°	Tilt Angle		
				45°	60°	75°
$\Delta u_0 = 20$	simulated	(0.02, 0.00)	(0.01, 0.00)	(0.01, 0.00)	(0.00, 0.00)	(0.00, 0.00)
	deduced	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)	(0.00, 0.00)
$\Delta v_0 = 20$	simulated	(1.82, 0.00)	(1.79, 0.00)	(1.50, 0.00)	(1.00, 0.00)	(0.46, 0.00)
	deduced	(1.81, 0.00)	(1.80, 0.00)	(1.51, 0.00)	(1.02, 0.00)	(0.47, 0.00)
$\Delta\alpha = 20$	simulated	(1.50, 0.11)	(1.49, 0.11)	(1.48, 0.10)	(1.47, 0.10)	(1.46, 0.10)
	deduced	(1.44, 0.10)	(1.43, 0.10)	(1.42, 0.10)	(1.41, 0.10)	(1.40, 0.10)
$\Delta\beta = 20$	simulated	(-0.47, -0.05)	(-1.00, -0.05)	(-1.46, -0.05)	(-1.72, -0.05)	(-1.73, -0.05)
	deduced	(-0.46, -0.05)	(-1.05, -0.05)	(-1.58, -0.06)	(-1.89, -0.06)	(-1.83, -0.06)
$\Delta x_1' = 4$	simulated	(2.75, 0.19)	(3.07, 0.22)	(3.16, 0.22)	(3.01, 0.21)	(2.65, 0.19)
	deduced	(2.52, 0.18)	(2.78, 0.20)	(2.85, 0.20)	(2.73, 0.19)	(2.43, 0.17)
$\Delta y_3' = 4$	simulated	(2.97, 0.10)	(3.80, 0.13)	(4.15, 0.14)	(3.90, 0.13)	(3.14, 0.11)
	deduced	(2.58, 0.09)	(3.18, 0.11)	(3.42, 0.11)	(3.25, 0.11)	(2.71, 0.09)

3.5 Experiments over Real Images

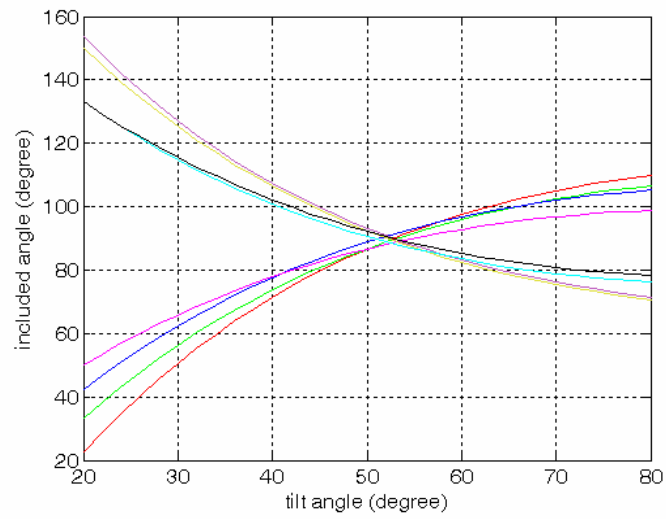
In this section, some experimental results over real data are demonstrated. We show the pose estimation of a PTZ camera first. Then, the calibration of multiple PTZ cameras will be performed. Finally, we give some discussions and a comparison with the conventional calibration method based on homography.

3.5.1 Calibration Results

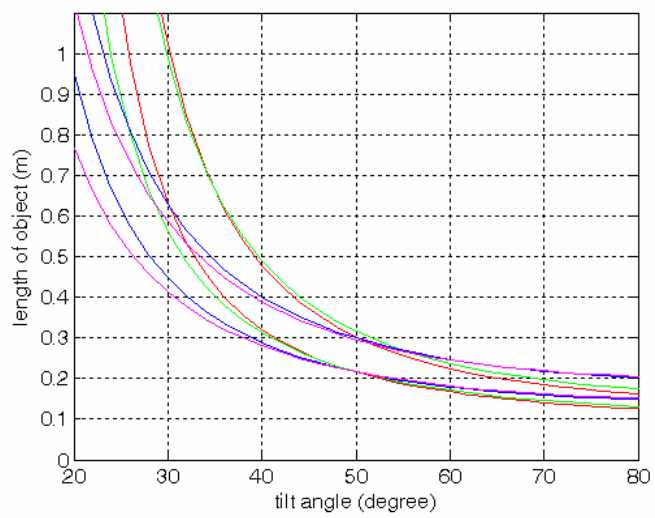
In this section, some static calibration results of multiple cameras on real data are demonstrated. In this simulation, the test images are captured by a camera mounted on the ceiling with an unknown tilt angle. The image resolution is 320 by 240 pixels. A few A4 papers are randomly placed on a horizontal table, as shown in Fig. 3.13. The corners of these A4 paper sheets can be easily identified either by hand or by a corner detection algorithm. In our experiment, we identify these corners manually and we have developed a software package to facilitate the identification of corners and line segments in images. In the 3-D space, all the corners are 90 degrees, while the length and width of an A4 paper are 297 mm and 210 mm, respectively. In Fig. 3.13(b) and (c), we show the deduced ψ -v.s.- ϕ curves and L-v.s.- ϕ curves based on the corners and the boundaries of these two A4 papers. The tilt angle is then estimated to be around 52 degrees. Note that there are two intersection points in Fig. 13(c) whose vertical coordinate correspond to the length and width of an A4 paper, respectively.



(a)



(b)



(c)

Fig. 3.13 (a) Test image (b) Deduced ψ -v.s.- ϕ curves (c) Deduced L-v.s.- ϕ curves.

Table 3.3 Upper Table: Estimation of Tilt Angle and Altitude.

Lower Table: Spatial Relationship among Cameras.

	Mean		Deviation	
	ϕ (degree)	h (m)	ϕ (degree)	h (m)
Camera 1	20.4	1.96	1.5	0.12
Camera 2	51.1	2.32	0.6	0.03
Camera 3	24.8	1.95	1.2	0.17
Camera 4	44.2	2.01	0.3	0.14

Relative position & orientation				
	X(m)	Y(m)	Z(m)	φ (degree)
Camera 1	2.7	-0.36	5.70	144.1
Camera 2	0	0	0	0
Camera 3	4.27	-0.37	1.93	87.1
Camera 4	-1.64	-0.31	3.2	-135.4

The upper part of Table 3.3 lists some experimental results for the calibration of multiple cameras. Here, the intrinsic parameters $\{\alpha, \beta, u_0, v_0\}$ of each camera are estimated in advance, based on Zhang's calibration method [4]. The parameter r can be estimated via direct measurement. Hence, equation (3.13) includes only 2 unknown variables: ϕ and h . Each row of Table 3.3 lists the mean and standard deviation of the estimated parameters for a single camera. To calculate the mean and standard deviation, five observations are made with each observation including 8 selected line segments on the boundary of these A4 papers, as shown in Fig. 3.14(a). It can be seen that all the estimated parameters have an acceptably small standard deviation.

In the lower part of Table 3.3, each row corresponds to the estimations of the position and orientation of each camera with respect to Camera 2. The relative position and orientation are computed based on the mean value of ϕ and h and one common vector in Fig. 3.14(a). The top view of the relative positions in the 3D space

is illustrated in Fig. 3.14(b). The eight chosen corners of the A4 papers in Fig. 3.14(a) are also plotted in Fig. 3.14(b) to offer a clearer geometric sense.

To evaluate the calibration results, we randomly pick up a few test points in the image captured by Camera 2 and use the calibration result to find the corresponding points on the other three images. The results are shown in Fig. 3.15, with all corresponding points being represented in the same color. It can be seen that all the corresponding relationships are quite reasonable. This verifies the credibility of the proposed calibration method.



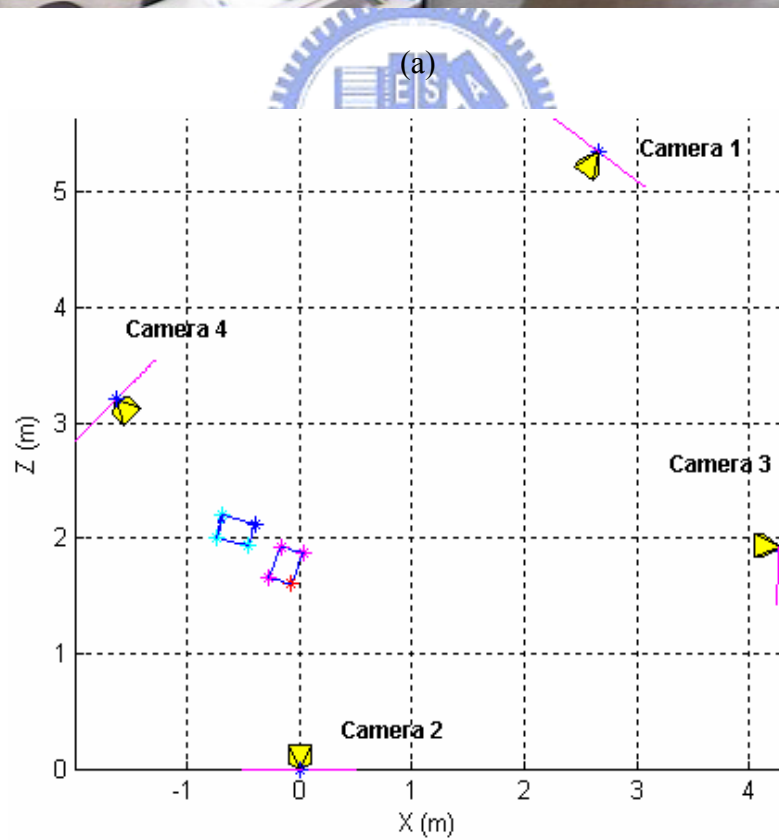


Fig. 3.14 (a) Test image captured by four cameras. (b) Top view of the relative positions between four cameras.



Fig. 3.15 Evaluation of calibration results.

3.5.2 Discussion and Comparison with the Homography Technique

In practice, a commonly used technique for camera pose estimation is to find the homography matrix between a reference plane in the 3-D space and the camera's image plane. The rotation and translation matrices can then be extracted by applying the SVD method over the homography matrix. In the homography approach, we need to define a reference world coordinate system and need to pick up a few spatial points with known reference world coordinates in advance. In other words, not only the distances but also the relative spatial information among the calibration points needs to be known. In comparison, our approach does not need to know the world coordinates of these calibration objects. We only need to measure the lengths or

angles of the calibration objects. Hence, the preparation of calibration objects becomes much easier in our approach.

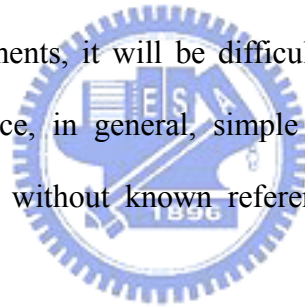
Besides, we may use fewer spatial points for the calibration of camera pose. This is because there is an implicit constraint in our approach. In a typical setup of PTZ cameras, the horizontal axis of the camera's image coordinate system is usually parallel to the ground plane. This parallelism is kept all the time even though the camera is under the panning, tilting, and zooming operations from time to time. Moreover, in our approach, we do not actually care about the exact pan angle of the camera. These two constraints correspond to the constraints over the rotation about the Z axis and the rotation about the Y axis in our rectified world coordinate system. Hence, our method may lead to stable pose estimations even when we only use a few calibration points.

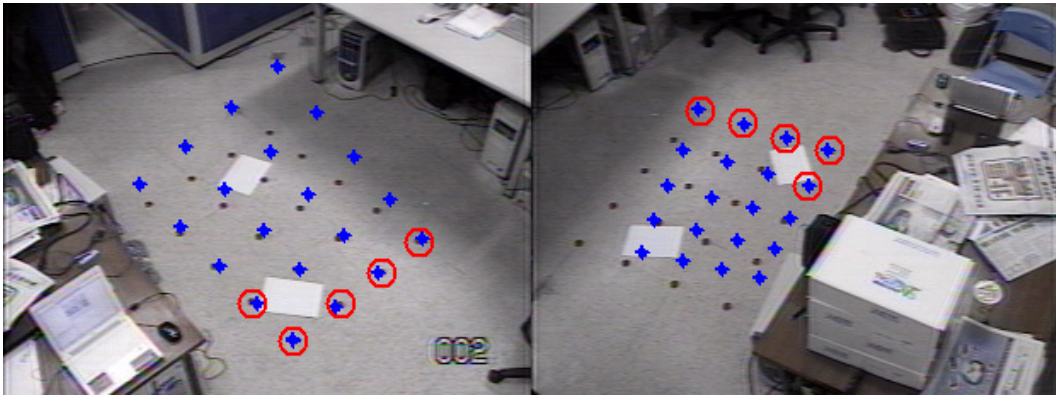


Fig. 3.16 Test images with a rectangular calibration pattern.

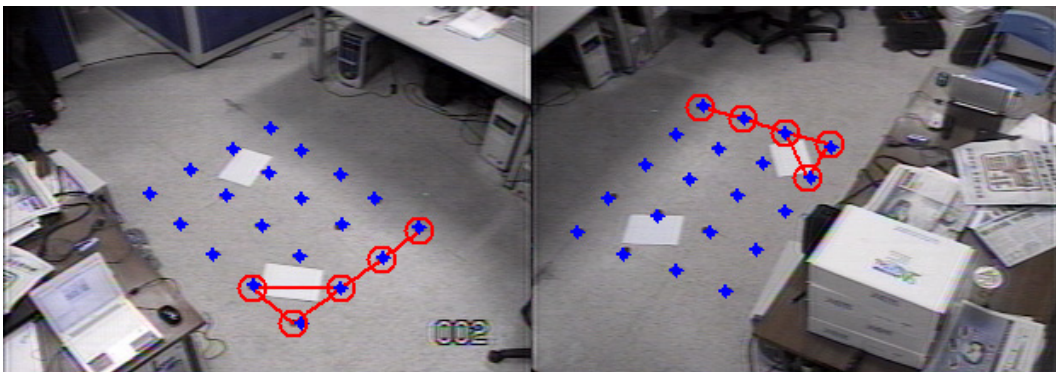
To compare with the homography technique, we marked 20 points on the ground floor to form a rectangular pattern, as shown in Fig. 3.16. Five of these 20 points are chosen to be the calibration points, as marked by the circles in Fig. 3.17. The asterisk markers in Fig. 3.17(a) show the point correspondence based on the calibration result

of the homography technique using the OpenCV library. On the other hand, the asterisk markers in Fig. 3.17(b) show the point correspondence based on our approach using five segments with known lengths. Besides, we calibrated these two cameras twenty times by randomly choosing five of these 20 points as the calibration points. We then checked the point correspondence in the image captured by Camera 4 based on these 20 image points of Camera 2 and the calibration results. Table 3.4 shows the mean absolute distance and standard deviation of the point-wise correspondence. It can be easily seen that our approach offers a more reliable and stable calibration result even when we only use a few calibration points. When all twenty points are used, on the other hand, there would be no obvious difference between the performance of the homography technique and the performance of our approach. Nevertheless, for general surveillance environments, it will be difficult to place this kind of specific patterns for calibration. Hence, in general, simple calibration objects or a small amount of calibration points without known reference spatial coordinates will be preferred.





(a)



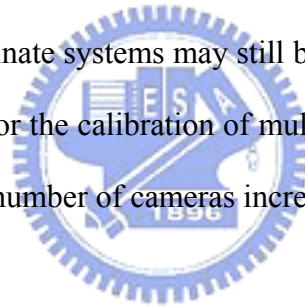
(b)

Fig. 3.17 Evaluation of calibration results by using five points. (a) Point correspondence based on the homography technique. (b) Point correspondence based on the proposed method.

Table 3.4 Mean Absolute Distance and Standard Deviation of the Point-wise Correspondence.

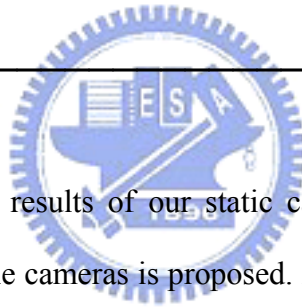
	Mean Absolute Distance (pixels)	Deviation (pixels)
Homography technique	6.0	5.6
Proposed method	3.4	1.4

Another advantage of our method is the comprehensible sense of camera pose. The tilt angle, altitude, and orientation of the camera offer more direct physical sense about the camera pose in the 3D space, especially when the PTZ cameras are under panning or tilting operations from time to time. In our approach, we derive some explicit formula to describe how the tilt angle and altitude of a PTZ camera affects the 3D-to-2D projection. This makes the calculation of 2D-to-3D back-projection much easier without the need of indirect depth computation. Besides, based on the comprehensible space sense, the relationships among multiple cameras can be easily obtained without complicated computations. In comparison, if using the conventional homography technique, the relative position and orientation between each pair of cameras offer less comprehensible sense about the setting of multiple cameras. Although these relative coordinate systems may still be transformed into an integrated coordinate system, the work for the calibration of multiple cameras will become more and more elaborate when the number of cameras increases.



CHAPTER 4

Dynamic Calibration of Multiple Cameras



In this chapter, based on the results of our static calibration, a new algorithm for dynamic calibration of multiple cameras is proposed. After the setup of PTZ cameras, we perform static camera calibration first based on the calibration method proposed in Chapter 3. As cameras begin to pan or tilt, we keep extracting and tracking feature points based on the Kanade-Lucas-Tomasi (KLT) algorithm [45]. In Section 4.1, 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. 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. In Section 4.2, we describe how to filter out undesired feature points when moving objects are present. In Fig. 4.1, we show an overall picture of the proposed dynamic calibration algorithm. Besides, the sensitivity analysis with respect to measurement errors and the fluctuations of previous estimations will be addressed in

Section 4.3. Finally, in Section 4.4, the efficiency and feasibility of this approach has been demonstrated in some experiments over real scenery.

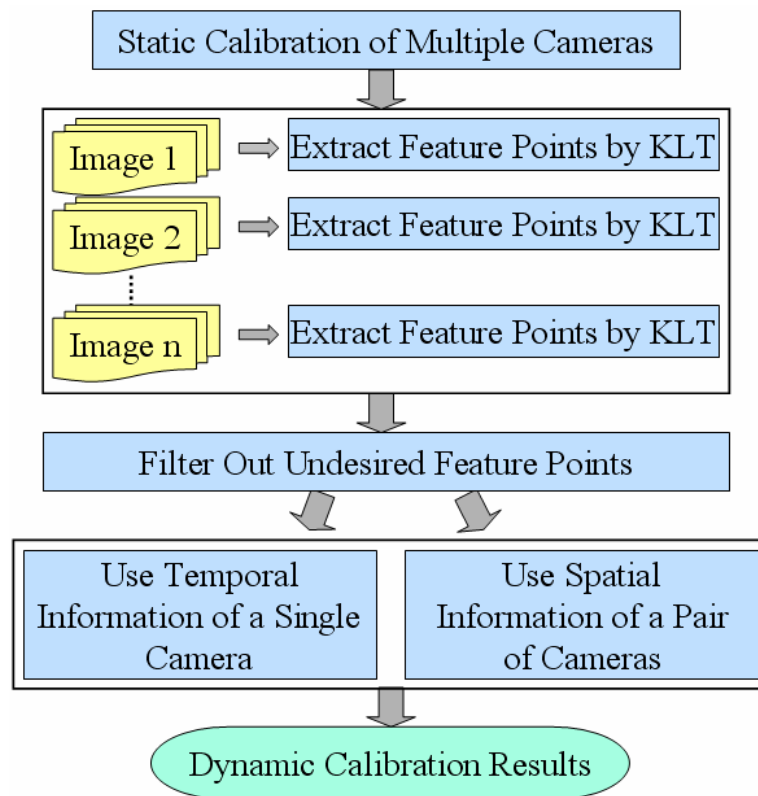


Fig. 4.1 Flowchart of the proposed dynamic calibration algorithm.

4.1 Dynamic Calibration of Multiple Cameras

In this section, we explain how we perform dynamic calibration process based on temporal and 3-D spatial information. In Section 4.1.1, we first deduce the formulae related to the mapping between the 3-D space and the image plane on a tilted and panned camera. Next, we will introduce how to calibrate a dynamic camera based on the displacement of feature points in the temporal domain in Section 4.1.2. After that, in Section 4.1.3, we will apply the epipolar-plane constraint over each pair of cameras to obtain more robust calibration.

4.1.1 Coordinate Mapping on a Tilted and Panned Camera

In Section 3.3.2, we indicated that the pan angles of cameras do not need to be considered when performing our static calibration. In this section, however, we further deal with the problem of dynamic calibration. In Section 3.2, we have deduced the back projection formula from the image coordinates of a tilted camera to the rectified world coordinates, under the constraint that all of the observed points are lying on a horizontal plane with $Y = -h$. Here, we slightly modify the equations in Section 3.2 to take into account both tilt angle and pan angle.

When a camera has a tilt angle ϕ and a pan angle θ with respect to its rectified pose, the projection center moves to a new position O_C' , with $O_C' = [r\cos\phi\sin\theta \ -r\sin\phi \ r(\cos\phi\cos\theta-1)]^T$. With respect to the tilted and panned camera, we define a new world coordinate system (X', Y', Z') , with the origin being the new project center O_C' , the Z' -axis being along the optical axis of the camera, and the X' - and Y' -axis being parallel to the x- and y-axis of the new projected image plane, respectively. Via straightforward calculations, it can be deduced that in the new world coordinate system the coordinates of the 3-D point p become

$$\begin{aligned} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \begin{bmatrix} X - r\cos\phi\sin\theta \\ Y + r\sin\phi \\ Z + r(1 - \cos\phi\cos\theta) \end{bmatrix} \\ &= \begin{bmatrix} X\cos\theta - Z\sin\theta - r\sin\theta \\ X\sin\phi\sin\theta + Y\cos\phi + Z\sin\phi\cos\theta + r\sin\phi\cos\theta \\ X\cos\phi\sin\theta - Y\sin\phi + Z\cos\phi\cos\theta + r(\cos\phi\cos\theta - 1) \end{bmatrix}. \end{aligned} \quad (4.1)$$

After perspective projection, the homogeneous coordinates of the projected image point become

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \frac{X'}{Z'} \\ \beta \frac{Y'}{Z'} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \frac{X \cos \theta - Z \sin \theta - r \sin \theta}{X \cos \phi \sin \theta - Y \sin \phi + Z \cos \phi \cos \theta + r(\cos \phi \cos \theta - 1)} \\ \beta \frac{X \sin \phi \sin \theta + Y \cos \phi + Z \sin \phi \cos \theta + r \sin \phi \cos \theta}{X \cos \phi \sin \theta - Y \sin \phi + Z \cos \phi \cos \theta + r(\cos \phi \cos \theta - 1)} \\ 1 \end{bmatrix}. \quad (4.2)$$

Furthermore, via a straightforward deduction similar to that in Section 3.2.2, we may deduce (4.3) to express the back projection function $B(p, \theta, \phi, h, \mathbf{\Omega})$ from the image coordinates $p = (x, y)$ on a tilted and panned camera to the rectified world coordinate system, under the constraint that the observed 3-D point is lying on a horizontal plane with $Y = -h$.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{\beta x \cos \theta (r \sin \phi - h) + \alpha y \sin \theta (r - h \sin \phi) - \alpha \beta h \cos \phi \sin \theta}{\alpha (y \cos \phi - \beta \sin \phi)} \\ -h \\ \frac{\beta x \sin \theta (h - r \sin \phi) + \alpha y \cos \theta (r - h \sin \phi) - \alpha \beta h \cos \phi \cos \theta}{\alpha (y \cos \phi - \beta \sin \phi)} - r \end{bmatrix} \quad (4.3)$$

$\equiv B(p, \theta, \phi, h, \mathbf{\Omega})$

Here, $\mathbf{\Omega}$ represents the set of intrinsic parameters of the camera. Note that in (4.3) we have ignored the offset terms, u_0 and v_0 , to simplify the formulation.

4.1.2 Dynamic Calibration of a Single Camera Based on Temporal Information

Assume we have a set of PTZ cameras. At the beginning, we calibrate the 3-D pose of each camera via the static calibration method introduced in Chapter 3. After that, we allow each PTZ camera to pan and tilt freely.

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 content. Here, we use the KLT method [45] 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. 4.2, if three 3-D points, P_A , P_B and P_C , are replaced by another three points, \hat{P}_A , \hat{P}_B and \hat{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 image onto a 3-D pseudo plane with a constant Z coordinate, as shown in Fig. 4.2.

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 θ_{t-1} and the tilt angle ϕ_{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 capturing I_t , we try to find the optimal $\Delta\theta_t$ and $\Delta\phi_t$ that minimize the following formula:

$$D = \sum_{k=1}^K \left\| \hat{B}(\hat{p}_k, \theta_{t-1} + \Delta\theta_t, \phi_{t-1} + \Delta\phi_t) - \hat{B}(p_k, \theta_{t-1}, \phi_{t-1}) \right\|^2. \quad (4.4)$$

In (4.4), \hat{B} represents the back projection function of an image feature point onto a pseudo 3-D plane Π' . Here, we especially use “hat” to denote that the back-projection is restricted to a vertical pseudo plane Π' . Besides, \mathbf{p}_k denotes a feature point in I_{t-1} and $\hat{\mathbf{p}}_k$ denotes the same feature point in I_t . K is the total number of image feature points for calibration. Note that in (4.4), we ignore the altitude parameter h of these back-projected points. This is because the altitude h can be obtained from (4.3) once if the Z coordinate is fixed. We also ignore the intrinsic parameters $\mathbf{\Omega}$ since they are not changed when the camera pans and tilts.

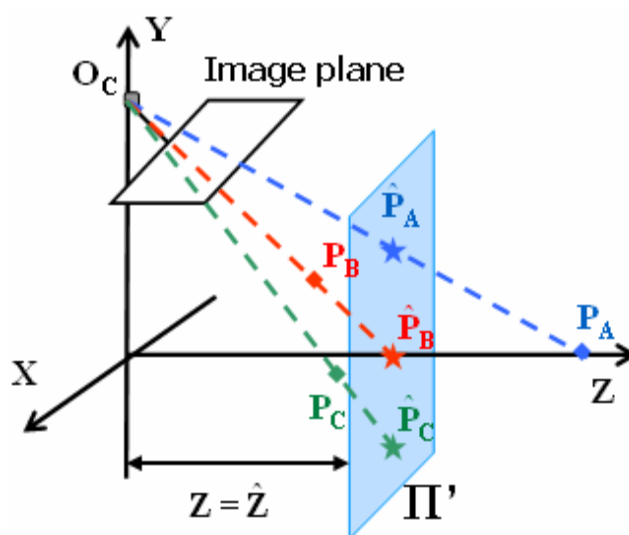


Fig. 4.2. Illustration of a pseudo plane Π' .

4.1.3 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. 4.3, we show the epipolar geometry for a pair of cameras [3, pp. 216-219]. For these two cameras, their projection centers, O_{C1} and O_{C2} , together with a 3-D point P_A , determine an epipolar plane Π . This epipolar plane Π intersects the image planes of the cameras to form two epipolar lines l_1 and l_2 . If \mathbf{p}_A^1 and \mathbf{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} , \mathbf{p}_A^1 , and \mathbf{p}_A^2 are coplanar and the epipolar plane Π can be expressed as

$$\begin{aligned} \pi(O_{C1}, O_{C2}, p_A^1, \theta^1, \phi^1) &\equiv \overline{O_{C2}O_{C1}} \times \overline{O_{C1}B(p_A^1, \theta^1, \phi^1)} \\ \text{or } \pi(O_{C1}, O_{C2}, p_A^2, \theta^2, \phi^2) &\equiv \overline{O_{C1}O_{C2}} \times \overline{O_{C2}B(p_A^2, \theta^2, \phi^2)}. \end{aligned} \quad (4.5)$$

In (4.5), we use the $B(\cdot)$ function defined in (4.4). 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.

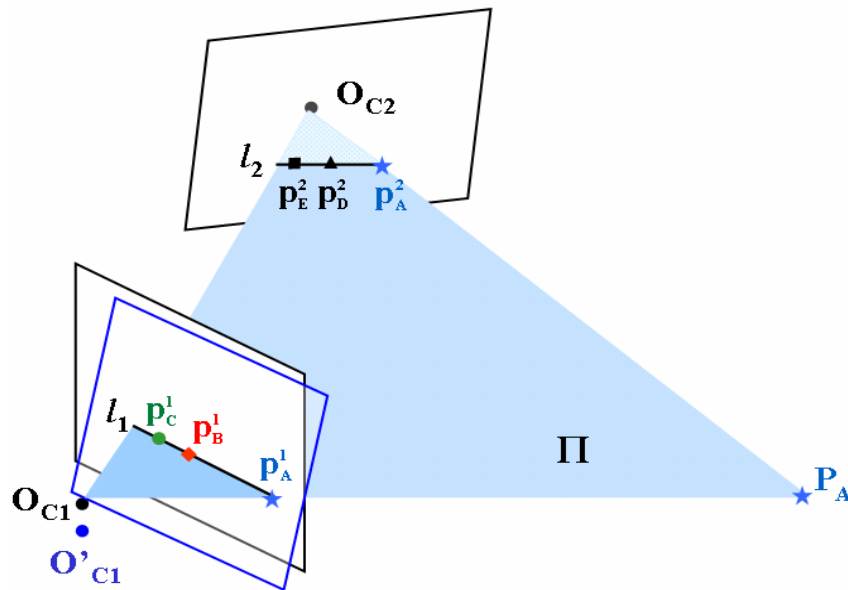


Fig. 4.3 Illustration of epipolar-plane constraint.

On the other hand, if some other points lie on the same pair of epipolar lines, like \mathbf{p}_B^1 and \mathbf{p}_C^1 on l_1 and \mathbf{p}_D^2 and \mathbf{p}_E^2 on l_2 , the back-projected points of these points also have to lie on the same epipolar plane Π . Traditionally, when we deal with the calibration of this camera pair, we try to figure out the pair-wise correspondence between $\{\mathbf{p}_A^1, \mathbf{p}_B^1, \mathbf{p}_C^1\}$ and $\{\mathbf{p}_A^2, \mathbf{p}_D^2, \mathbf{p}_E^2\}$. If we may place some pre-defined calibration patterns or landmarks in the 3-D scene, the correspondence of feature points can be easily achieved. However, in real cases, especially when cameras are panning and tilting all the time, calibration patterns or landmarks may get occluded or move out of image scopes.

If we do not have calibration patterns or landmarks with us, one possible way to achieve dynamic calibration is to automatically extract new feature points from the image contents and use them as pseudo landmarks. However, this kind of approach requires point-wise correspondence between each image pair and this point-wise correspondence has long been recognized as a cumbersome problem in computer vision, especially when a lot of feature points are involved. Moreover, for a wide-range video surveillance system, the image contents of different cameras may be very different. In this case, the correspondence of image feature points among different cameras is even more difficult.

In this section, we adopt a different approach to avoid the troublesome point-wise correspondence. As illustrated in Fig. 4.3, we assume a pair of camera has initially been calibrated via some kind of calibration algorithm. We assume a few features, like $\mathbf{p}_A^1, \mathbf{p}_B^1, \mathbf{p}_C^1, \mathbf{p}_A^2, \mathbf{p}_D^2, \mathbf{p}_E^2$, are located on a pair of corresponding epipolar lines. Without performing point-wise 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. 4.3, we assume a pair of cameras has been calibrated at the time instant $t-1$ and an epipolar plane Π has been identified. Assume at that time instant $t-1$, the pan and tilt angles of Camera-1 are θ^1_{t-1} and ϕ^1_{t-1} , while the pan and tilt angles of Camera-2 are θ^2_{t-1} and ϕ^2_{t-1} . Camera-1 captures the image I^1_{t-1} , while Camera-2 captures I^2_{t-1} . On the other hand, at the time instant t , Camera-1 rotates to a new pan angle $(\theta^1_{t-1} + \Delta\theta^1_t)$ and a new tilt angle $(\phi^1_{t-1} + \Delta\phi^1_t)$, while Camera-2 rotates to $(\theta^2_{t-1} + \Delta\theta^2_t)$ and $(\phi^2_{t-1} + \Delta\phi^2_t)$. 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 \mathbf{p}_A^1 has been extracted from I^1_{t-1} . This feature moves to $\hat{\mathbf{p}}_A^1$ in I^1_t . Based on \mathbf{p}_A^1 , θ^1_{t-1} , and ϕ^1_{t-1} , we may form an epipolar plane Π . At the time instant t , we then seek to find the angles $(\theta^1_{t-1} + \Delta\theta^1_t)$ and $(\phi^1_{t-1} + \Delta\phi^1_t)$ such that $\hat{\mathbf{p}}_A^1$ still locates on the same epipolar plane. That is, we seek to find $\Delta\theta^1_t$ and $\Delta\phi^1_t$ such that

$$B(\hat{\mathbf{p}}_A^1, \theta^1_{t-1} + \Delta\theta^1_t, \phi^1_{t-1} + \Delta\phi^1_t) \cdot \pi(O_{C1}, O_{C2}, \mathbf{p}_A^1, \theta^1_{t-1}, \phi^1_{t-1}) = 0. \quad (4.6)$$

Similarly, for \mathbf{p}_B^1 and \mathbf{p}_C^1 that share the same epipolar line with \mathbf{p}_A^1 , we have

$$\begin{aligned} & B(\hat{\mathbf{p}}_B^1, \theta^1_{t-1} + \Delta\theta^1_t, \phi^1_{t-1} + \Delta\phi^1_t) \cdot \pi(O_{C1}, O_{C2}, \mathbf{p}_A^1, \theta^1_{t-1}, \phi^1_{t-1}) \\ & \text{and } B(\hat{\mathbf{p}}_C^1, \theta^1_{t-1} + \Delta\theta^1_t, \phi^1_{t-1} + \Delta\phi^1_t) \cdot \pi(O_{C1}, O_{C2}, \mathbf{p}_A^1, \theta^1_{t-1}, \phi^1_{t-1}). \end{aligned} \quad (4.7)$$

Note that in (4.6) and (4.7), 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

$\{\mathbf{p}_{j,1}^1, \mathbf{p}_{j,2}^1, \dots, \mathbf{p}_{j,n_j}^1\}$ on I_{t-1}^1 . These n_j feature points move to $\{\hat{\mathbf{p}}_{j,1}^1, \hat{\mathbf{p}}_{j,2}^1, \dots, \hat{\mathbf{p}}_{j,n_j}^1\}$ on I_t^1 . Besides, we assume \mathbf{p}_j^1 denotes one of the feature points in $\{\mathbf{p}_{j,1}^1, \mathbf{p}_{j,2}^1, \dots, \mathbf{p}_{j,n_j}^1\}$.

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

$$G_t^1 = \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| B(\hat{p}_{j,i}^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_j^1, \theta_{t-1}^1, \phi_{t-1}^1) \right\|^2. \quad (4.8)$$

Furthermore, by integrating (4.4) and (4.8), the changes of pan angle and tilt angle of Camera-1 can be estimated by minimizing the following formula:

$$F_t^1 = \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| \hat{B}(\hat{p}_{j,i}^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) - \hat{B}(p_{j,i}^1, \theta_{t-1}^1, \phi_{t-1}^1) \right\|^2 + \lambda \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| B(\hat{p}_{j,i}^1, \theta_{t-1}^1 + \Delta\theta_t^1, \phi_{t-1}^1 + \Delta\phi_t^1) \cdot \pi(O_{C1}, O_{C2}, p_j^1, \theta_{t-1}^1, \phi_{t-1}^1) \right\|^2. \quad (4.9)$$

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

$$F_t^2 = \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| \hat{B}(\hat{p}_{j,i}^2, \theta_{t-1}^2 + \Delta\theta_t^2, \phi_{t-1}^2 + \Delta\phi_t^2) - \hat{B}(p_{j,i}^2, \theta_{t-1}^2, \phi_{t-1}^2) \right\|^2 + \lambda \sum_{j=1}^m \sum_{i=1}^{n_j} \left\| B(\hat{p}_{j,i}^2, \theta_{t-1}^2 + \Delta\theta_t^2, \phi_{t-1}^2 + \Delta\phi_t^2) \cdot \pi(O_{C1}, O_{C2}, p_j^2, \theta_{t-1}^2, \phi_{t-1}^2) \right\|^2. \quad (4.10)$$

Here, λ 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 (4.9) or (4.10) to solve $\Delta\theta_t$ and $\Delta\phi_t$. That term assumes the $[X, Y, Z]$ coordinates of a back-projected point is fixed when a camera is under panning or tilting. Since each 3-D point is with a fixed Z coordinate, a feature point may offer two constraints over the X and Y coordinates and these two constraints can be used to solve $\Delta\theta_t$ and $\Delta\phi_t$. On the other hand, 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 (4.9) or (4.10) to make the estimation more robust.

To deduce $\Delta\theta^1_t$, $\Delta\phi^1_t$, $\Delta\theta^2_t$, and $\Delta\phi^2_t$, we adopt the Levenberg-Marquardt (LM) algorithm. In our experiments, the initial guesses of pan/tilt angle changes are set to be zero degrees. 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 (4.9) may be different from the n_j in (4.10). 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 (4.9) and (4.10) can also be merged together into a single formula during the optimization process.

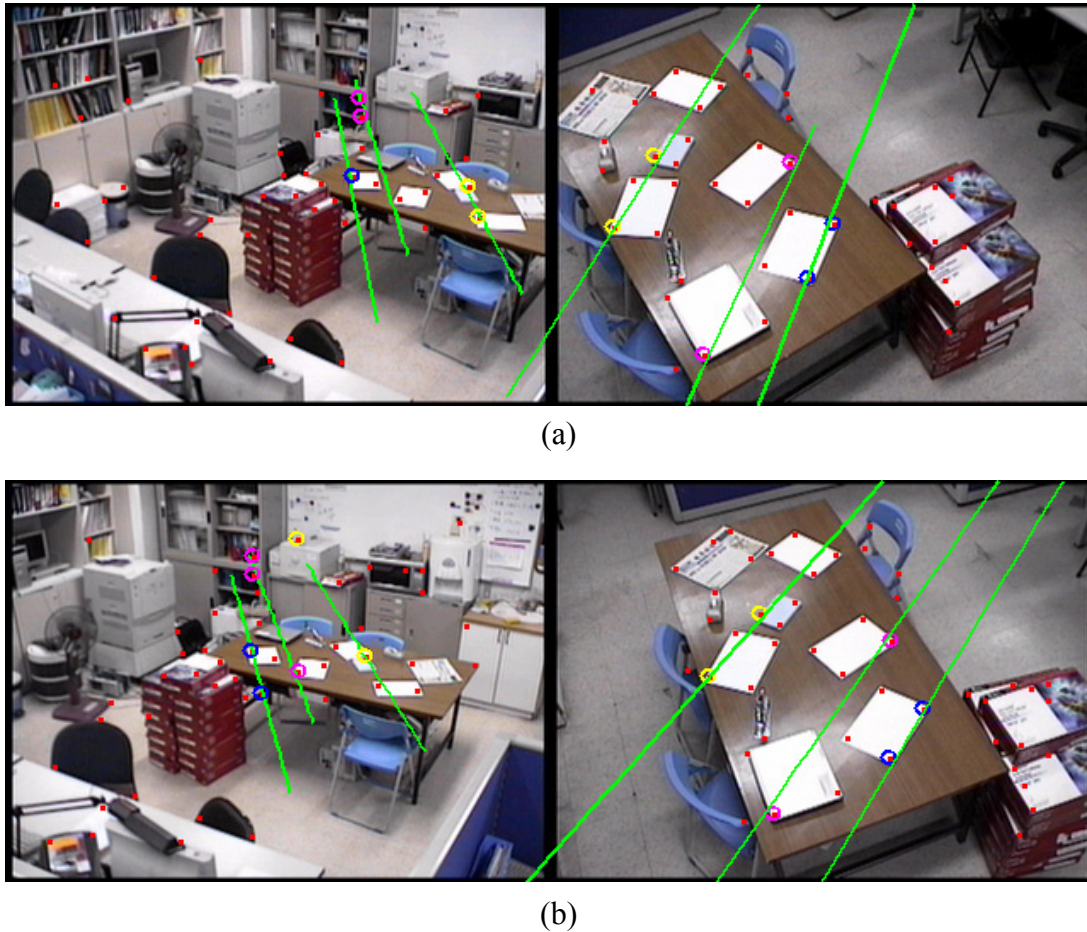


Fig. 4.4 Image pairs captured at two different time instants. Green lines indicate three pairs of corresponding epipolar lines.

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

Step 1. After the setup of multiple cameras, we perform static camera calibration based on the method introduced in Chapter 3. 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 [45]. 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 feature point is passed through by the epipolar line.

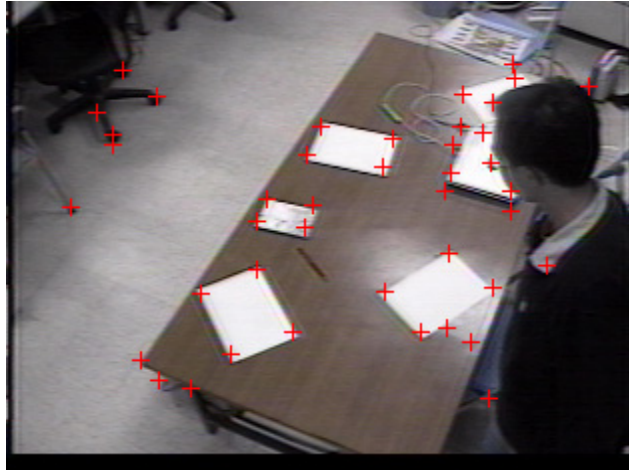
Step 4. Based on the extracted feature points and the information of epipolar lines, we calibrate the new pan angle and tilt angle for each pair of cameras by optimizing (4.9) and (4.10). After that, go back to Step 2.

The above procedure is repeated to acquire the new poses of all cameras. In Fig. 4.4(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.

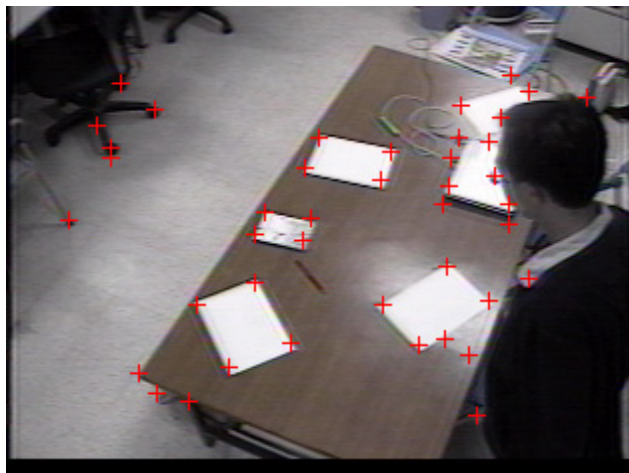
4.2 Dynamic Calibration with Presence of Moving Objects

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, people or moving objects may enter or leave the scene while cameras are capturing images. To guarantee accurate calibration, we need to get rid of these feature points related to moving objects.

In Fig. 4.5, we show two successive image frames where the camera tilts up by 0.5-degrees. For each feature point, we calculate its spatial displacement (dx , dy). The distribution of (dx , dy) is plotted in Fig. 4.6, 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.



(a)



(b)

Fig. 4.5 (a) Image captured by a camera with 55.1° tilt angle. (b) Image captured by a camera with 54.6° tilt angle. Red crosses represent feature points extracted by the KLT algorithm.

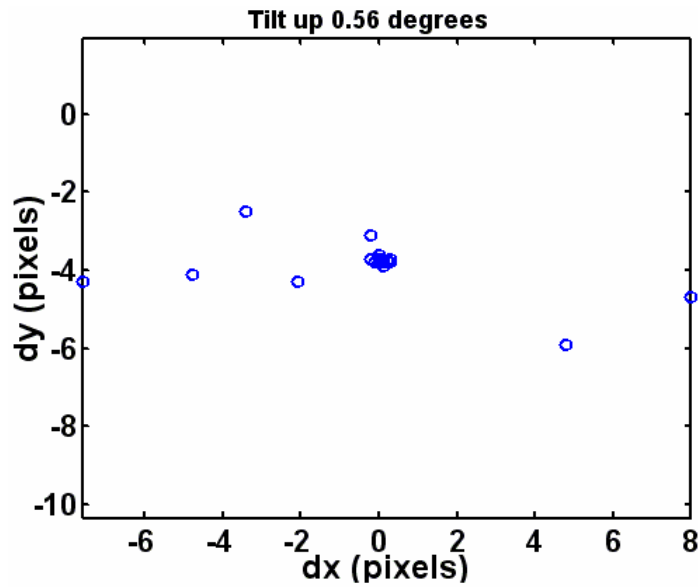
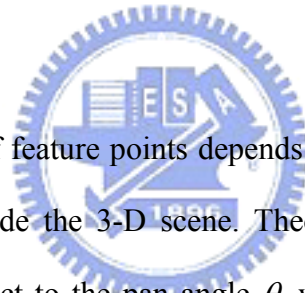


Fig. 4.6 The distribution of spatial displacement for the extracted feature points in Fig. 4.5.



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, by taking the partial derivative of (4.2) with respect to the pan angle θ , we have (4.11), 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 (4.12). Similarly, by ignoring the effect of pan angle, (4.13) indicates how the location of a feature point varies with respect to the change of tilt angle. Both (4.12) and (4.13) 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.

$$\left\{ \begin{array}{l} \frac{\partial x}{\partial \theta} = -\frac{\alpha \cos \phi (X \cos \theta - Z \sin \theta - r \sin \theta)^2}{(X \cos \phi \sin \theta - Y \sin \phi + Z \cos \phi \cos \theta + r(\cos \phi \cos \theta - 1))^2} \\ \quad - \frac{\alpha (X \sin \theta + Z \cos \theta + r \cos \theta)}{X \cos \phi \sin \theta - Y \sin \phi + Z \cos \phi \cos \theta + r(\cos \phi \cos \theta - 1)} \\ \frac{\partial y}{\partial \theta} = -\beta (X \cos \theta \cos \phi - Z \cos \phi \sin \theta - r \cos \phi \sin \theta) \\ \quad + \frac{(X \sin \phi \sin \theta + Y \cos \phi + Z \sin \phi \cos \theta + r \sin \phi \cos \theta)}{(X \cos \phi \sin \theta - Y \sin \phi + Z \cos \phi \cos \theta + r(\cos \phi \cos \theta - 1))^2} \\ \quad + \frac{\beta (X \sin \phi \cos \theta - Z \sin \phi \sin \theta - r \sin \phi \sin \theta)}{X \cos \phi \sin \theta - Y \sin \phi + Z \cos \phi \cos \theta + r(\cos \phi \cos \theta - 1)} \end{array} \right. \quad (4.11)$$

$$\left[\begin{array}{l} \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial \theta} \end{array} \right] = \left[\begin{array}{l} -\alpha \frac{(X \cos \theta - Z \sin \theta - r \sin \theta)^2}{(X \sin \theta + Z \cos \theta + r(\cos \theta - 1))^2} - \alpha \left(1 + \frac{r}{X \sin \theta + Z \cos \theta + r(\cos \theta - 1)} \right) \\ \beta Y \frac{(X \cos \theta - Z \sin \theta - r \sin \theta)}{(X \sin \theta + Z \cos \theta + r(\cos \theta - 1))^2} \end{array} \right] \quad (4.12)$$

$$\left[\begin{array}{l} \frac{\partial x}{\partial \phi} \\ \frac{\partial y}{\partial \phi} \end{array} \right] = \left[\begin{array}{l} \alpha X \frac{(Y \cos \phi + Z \sin \phi + r \sin \phi)}{(-Y \sin \phi + Z \cos \phi + r(\cos \phi - 1))^2} \\ \beta \frac{(Y \cos \phi + Z \sin \phi + r \sin \phi)^2}{(-Y \sin \phi + Z \cos \phi + r(\cos \phi - 1))^2} + \beta \left(1 + \frac{r}{-Y \sin \phi + Z \cos \phi + r(\cos \phi - 1)} \right) \end{array} \right] \quad (4.13)$$

Furthermore, we illustrate the term $X \cos \theta - Z \sin \theta - r \sin \theta$ and $X \sin \theta + Z \cos \theta + r(\cos \theta - 1)$ of (4.12) in Fig 4.7. Assume there is a 3-D point \mathbf{P} with the world coordinates $[X, Y, Z]^T$. In Fig 4.7, when the camera rotates with a pan angle θ , its projection center O_c moves to O_c' and the world coordinates (X, Y, Z) changes to (X', Y', Z') . The term $X \cos \theta - Z \sin \theta - r \sin \theta$ represents the distance between \mathbf{P} and Z' axis, while the term $X \sin \theta + Z \cos \theta + r(\cos \theta - 1)$ represents the distance between \mathbf{P} and X' axis. In other words, from the view of a camera, (4.12) depends on the relative positions between the observed objects and the projection center. Formula (4.12) can also be

expressed as

$$\begin{bmatrix} \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial \theta} \end{bmatrix} = \begin{bmatrix} -\alpha \frac{X'^2}{Z'^2} - \alpha \left(1 + \frac{r}{Z'}\right) \\ \beta \frac{X'Y'}{Z'^2} \end{bmatrix}. \quad (4.14)$$

If r is far smaller than Z' , r/Z' in (4.14) can be ignored. For our cameras, r is with a centimeter level (about 3.5 centimeters), while most of the observed scenes are away from the cameras with a meter level. The situation about tilt angle is similar to that about pan angle. Hence, we may simply dismiss r here.

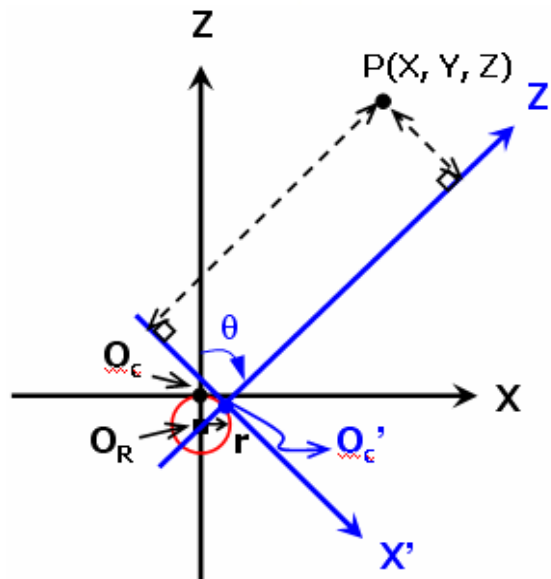
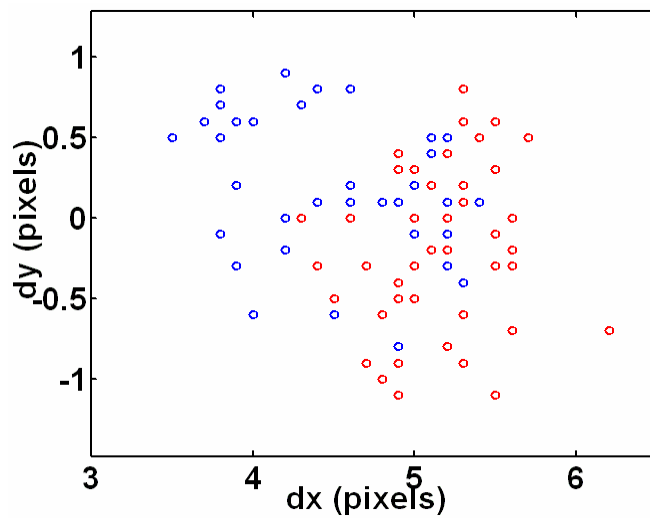
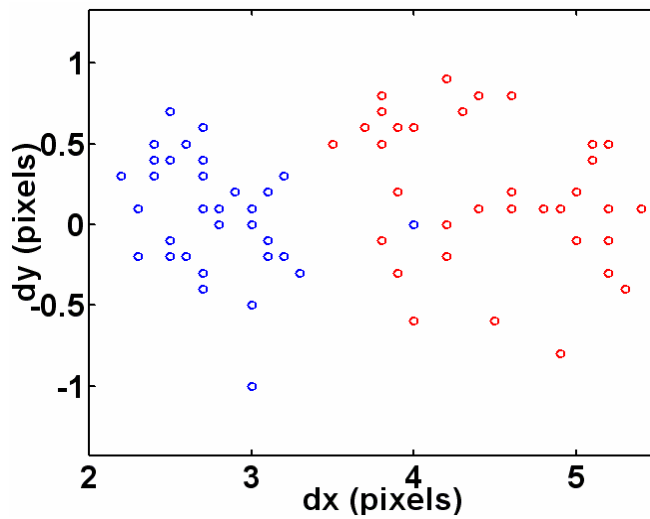


Fig. 4.7 Illustration of the coordinate system when camera is panning. If r is far smaller than Z' , we may simply dismiss r .



(a)



(b)

Fig. 4.8 (a) The displacements of feature points observed by two different cameras. Both cameras are under a 1-degree pan-angle change, while their tilt angles are fixed at 34.8° . (b) The displacements of feature points observed by the same camera but with different pan-angle changes. (Blue: 0.6-degree pan-angle change. Red: 1-degree pan-angle change.)

Figure 4.8 shows two simulation results that demonstrate the effects of 3-D scene and camera pose over the value of displacement. In Figure 4.8(a), we plot the

displacement of feature points observed by cameras at two different locations. Both cameras are under a 1-degree change of pan angle, while their tilt angles are fixed at 34.8° . Due to the different observed scenes, the displacements of feature points are different. On the other hand, Figure 4.8(b) shows the displacement observed by the same camera but with two different pan-angle changes. It can be observed that not only the displacement magnitudes are different; the distributions of displacement are also 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 training 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. 4.11(a) shows an example of images captured by these four cameras. In Fig. 4.9, 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. 4.10(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, Figure 4.10(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 is treated as an undesired feature point and is discarded in the dynamic calibration process.

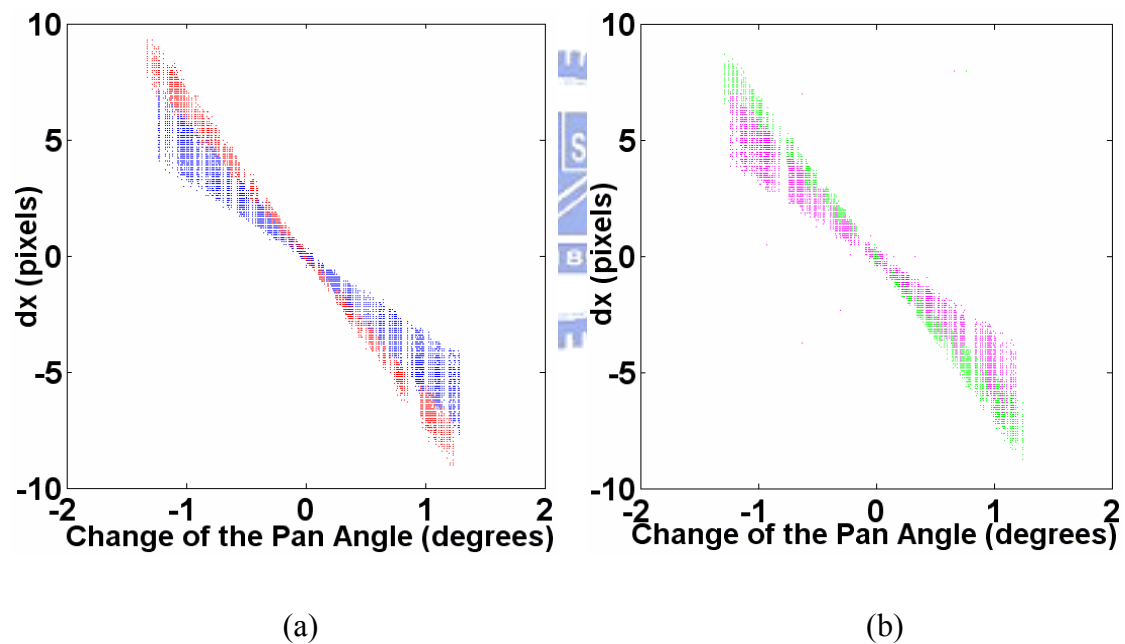
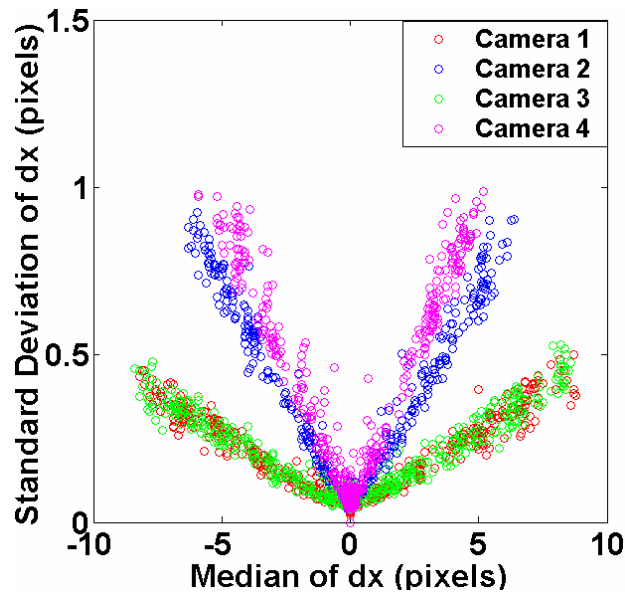
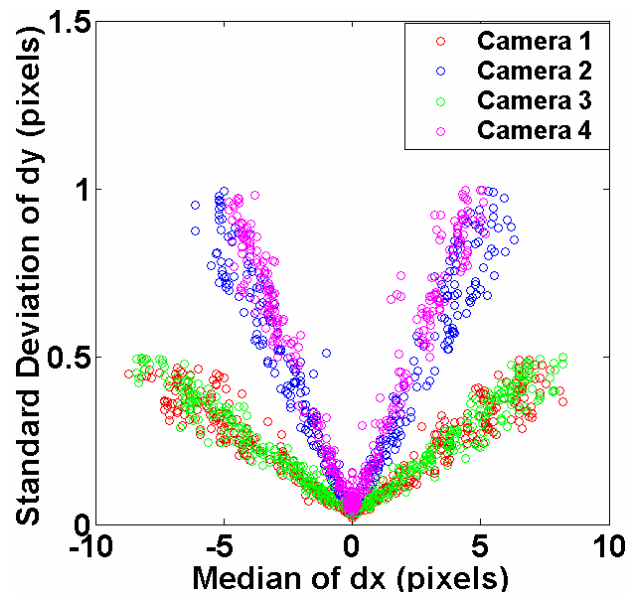


Fig. 4.9 The x-component displacement of feature points with respect to the changes of pan angle for four different cameras, without the presence of moving objects. The statistical relationships for Camera-1, Camera-2, Camera-3, and Camera-4 are plotted in red, blue, green, and magenta, respectively.



(a)



(b)

Fig. 4.10 (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.

4.3 Sensitivity Analysis

Based on (4.9) and (4.10), 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 (4.9) conforms to $f_1 \equiv \frac{\partial F_t^1}{\partial(\Delta\theta_t^1)} = 0$ and $f_2 \equiv \frac{\partial F_t^1}{\partial(\Delta\phi_t^1)} = 0$. Note that

in (4.9), 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 (4.9), we intentionally ignored that part in Section 4.1. 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 θ_{t-1}^1 and ϕ_{t-1}^1 , but also on θ_{t-1}^2 and ϕ_{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 θ_{t-1}^k , where $k = 1$ or 2 , we may apply the implicit function theorem over f_1 and f_2 to get

$$\begin{bmatrix} \frac{\partial(\Delta\theta_t^1)}{\partial(\theta_{t-1}^k)} \\ \frac{\partial(\Delta\phi_t^1)}{\partial(\theta_{t-1}^k)} \end{bmatrix} = - \begin{bmatrix} \frac{\partial f_1}{\partial(\Delta\theta_t^1)} & \frac{\partial f_1}{\partial(\Delta\phi_t^1)} \\ \frac{\partial f_2}{\partial(\Delta\theta_t^1)} & \frac{\partial f_2}{\partial(\Delta\phi_t^1)} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial f_1}{\partial(\theta_{t-1}^k)} \\ \frac{\partial f_2}{\partial(\theta_{t-1}^k)} \end{bmatrix}. \quad (4.15)$$

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

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 θ_{t-1}^k and ϕ_{t-1}^k and the measurement errors in $\{\hat{p}_{j,1}^1, \hat{p}_{j,2}^1, \dots, \hat{p}_{j,n_j}^1\}$, we have

$$\delta(\Delta\theta_t^1) \approx \sum_{k=1}^2 \left\{ \frac{\partial(\Delta\theta_t^1)}{\partial(\theta_{t-1}^k)} \delta(\theta_{t-1}^k) + \frac{\partial(\Delta\theta_t^1)}{\partial(\phi_{t-1}^k)} \delta(\phi_{t-1}^k) + \sum_{j=1}^m \sum_{i=1}^{n_j} \frac{\partial(\Delta\theta_t^1)}{\partial(\hat{p}_{j,i}^1)} \delta(\hat{p}_{j,i}^1) \right\} \quad (4.16)$$

and

$$\delta(\Delta\phi_t^1) \approx \sum_{k=1}^2 \left\{ \frac{\partial(\Delta\phi_t^1)}{\partial(\theta_{t-1}^k)} \delta(\theta_{t-1}^k) + \frac{\partial(\Delta\phi_t^1)}{\partial(\phi_{t-1}^k)} \delta(\phi_{t-1}^k) + \sum_{j=1}^m \sum_{i=1}^{n_j} \frac{\partial(\Delta\phi_t^1)}{\partial(\hat{p}_{j,i}^1)} \delta(\hat{p}_{j,i}^1) \right\}. \quad (4.17)$$

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 meters. If Camera-1 is translated by -0.69, -0.13 and 6.25 meters along X -, Y - and Z -axis, respectively, and then rotated by -143.64 degrees 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 Π with the homogeneous coordinates $\pi = [0.63, 0.77, 0.09, 0.01]$. Based on this plane Π , we deduce the corresponding epipolar lines on the image planes of 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 degrees 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 $\Delta\theta_1^1$ and $\Delta\phi_1^1$ 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 $\Delta\theta_1^1$ and $\Delta\phi_1^1$ vary. Here, the LM algorithm is applied to (4.9) for the estimation of $\Delta\theta_1^1$ and $\Delta\phi_1^1$. The variations of these estimation results, together with the variations deduced by (4.16) and (4.17) are listed in Table 4.1. Besides, we also show in Table 4.2 how $\Delta\theta_1^1$ and $\Delta\phi_1^1$ 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

$$\delta(\Delta\theta_1^1) \approx \frac{\partial(\Delta\theta_1^1)}{\partial(\hat{x}_{1,1}^1)} \frac{\partial(\hat{x}_{1,1}^1)}{\partial(d)} \delta(d) + \frac{\partial(\Delta\theta_1^1)}{\partial(\hat{y}_{1,1}^1)} \frac{\partial(\hat{y}_{1,1}^1)}{\partial(d)} \delta(d) \quad (4.18)$$

and

$$\delta(\Delta\phi_1^1) \approx \frac{\partial(\Delta\phi_1^1)}{\partial(\hat{x}_{1,1}^1)} \frac{\partial(\hat{x}_{1,1}^1)}{\partial(d)} \delta(d) + \frac{\partial(\Delta\phi_1^1)}{\partial(\hat{y}_{1,1}^1)} \frac{\partial(\hat{y}_{1,1}^1)}{\partial(d)} \delta(d). \quad (4.19)$$

It can be seen that the all deduced variations in Table 4.1 and 4.2 well approximate the simulation results.

Table 4.1 Variations of Estimation Results with respect to Previous Estimation Errors and Measurement Errors.

$(\theta_0^1, \phi_0^1) = (0^\circ, 20^\circ), (\Delta\theta_1^1, \Delta\phi_1^1) = (0^\circ, 0.5^\circ)$				
	$\phi_0^1 - 2^\circ$	$\phi_0^1 - 1^\circ$	$\phi_0^1 + 1^\circ$	$\phi_0^1 + 2^\circ$
simulated $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	-0.52, 0.60	-0.26, 0.30	0.27, -0.30	0.53, -0.60
deduced $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	-0.49, 0.60	-0.25, 0.30	0.25, -0.30	0.49, -0.59
	$\theta_n^1 - 2^\circ$	$\theta_n^1 - 1^\circ$	$\theta_n^1 + 1^\circ$	$\theta_n^1 + 2^\circ$
simulated $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.39, -0.44	0.19, -0.22	-0.19, 0.22	-0.38, 0.44
deduced $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.39, -0.47	0.20, -0.23	-0.20, 0.23	-0.39, 0.47
	$\hat{x}_{1i}^1 - 2$ pixels	$\hat{x}_{1i}^1 - 1$ pixels	$\hat{x}_{1i}^1 + 1$ pixels	$\hat{x}_{1i}^1 + 2$ pixels
simulated $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.08, -0.01	0.04, -0.01	-0.04, 0.00	-0.08, 0.01
deduced $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.08, -0.02	0.04, -0.01	-0.04, 0.01	-0.08, 0.02
	$\hat{y}_{1i}^1 - 2$ pixels	$\hat{y}_{1i}^1 - 1$ pixels	$\hat{y}_{1i}^1 + 1$ pixels	$\hat{y}_{1i}^1 + 2$ pixels
simulated $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.01, -0.09	0.01, -0.04	-0.01, 0.04	-0.01, 0.09
deduced $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.01, -0.09	0.00, -0.04	-0.00, 0.04	-0.01, 0.09
	$\phi_0^2 - 2^\circ$	$\phi_0^2 - 1^\circ$	$\phi_0^2 + 1^\circ$	$\phi_0^2 + 2^\circ$
simulated $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.49, -0.59	0.25, -0.29	-0.25, 0.28	-0.51, 0.54
deduced $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	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 $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.27, -0.32	0.14, -0.16	-0.15, 0.16	-0.30, 0.33
deduced $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	0.26, -0.32	0.13, 0.16	-0.13, 0.16	-0.27, 0.32

Table 4.2 Variations of Estimation Results with respect to Distance Fluctuations in Epipolar Lines.

$(\theta_0^1, \phi_0^1) = (0^\circ, 20^\circ), (\Delta\theta_1^1, \Delta\phi_1^1) = (0^\circ, 0.5^\circ)$						
d (pixels)	-3	-2	-1	+1	+2	+3
simulated $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	-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 $\Delta(\Delta\theta_1^1), \Delta(\Delta\phi_1^1)$	-0.04, 0.05	-0.03, 0.04	-0.02, 0.02	0.02, -0.02	0.04, -0.04	0.05, -0.05

Additionally, when the number of epipolar line pair doubles, the errors of estimated $\Delta\theta_1^1$ and $\Delta\phi_1^1$ caused by the fluctuations of the feature points are roughly halved. On the other hand, the errors of estimated $\Delta\theta_1^1$ and $\Delta\phi_1^1$ caused by the fluctuations of $\{\theta_0^1, \phi_0^1, \theta_0^2, \phi_0^2\}$ have no apparent changes. Besides, if we change the value of $\Delta\phi_1^1$ up to 5 degrees, 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 Table 4.1 and 4.2. Finally, we also change the tilt angle ϕ_0^1 from 20 to 80 degrees with a 20-degree step, and repeat the simulation. The variations of the simulation results also confirm to that in Table 4.1 and 4.2. 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 degrees. 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.

4.4 Experiments over Real Scenes

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 by 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. Figure 4.11(a) shows an example of captured images by these four cameras. In comparison, Fig. 4.11(b) shows the same images but with the presence of landmarks.

At the beginning of the experiment, the static calibration introduced in Chapter 3 was applied to calibrate the initial setup of these 4 cameras. The static calibration results are listed in Table 4.3. The left part of Table 4.3 lists for each camera the estimated tilt angle and its altitude above the brown table in the scene. The right part of Table 4.3 lists for each camera the estimated position and orientation with respect to Camera-2. To evaluate the static calibration result, we use the landmarks in the image captured by Camera-2 to infer the corresponding points on the other three images. The results are shown in Fig. 4.12. It can be seen that the correspondences are reasonably accurate. In addition, we also calculated the 3-D coordinates of these landmarks and used them as a ground truth for the evaluation of our dynamic calibration algorithm.



(a)



(b)

Fig. 4.11 (a) Test images captured by four cameras. (b) 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.

Table 4.3 Results of the Static Calibration.

	Tilt Angle & Altitude		Relative Position & Orientation			
	ϕ (degrees)	h (m)	X (m)	Y (m)	Z (m)	φ (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

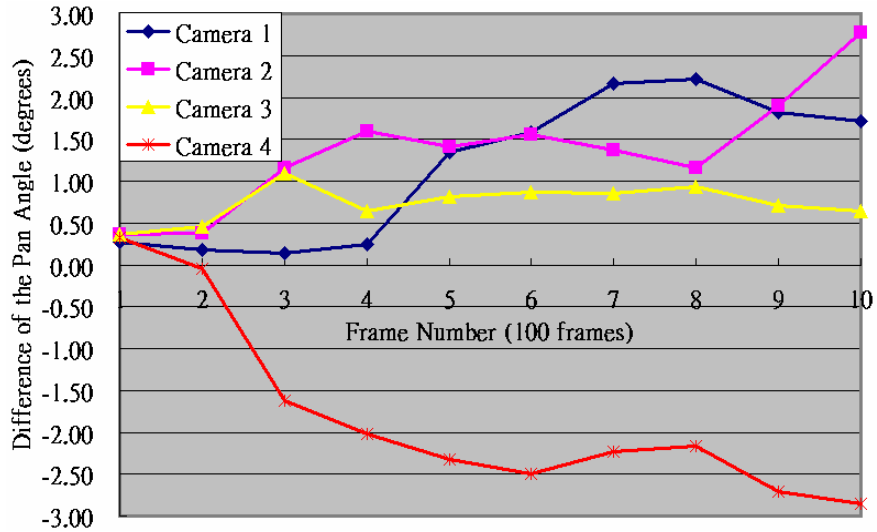


Fig. 4.12 Evaluation of initial calibration.

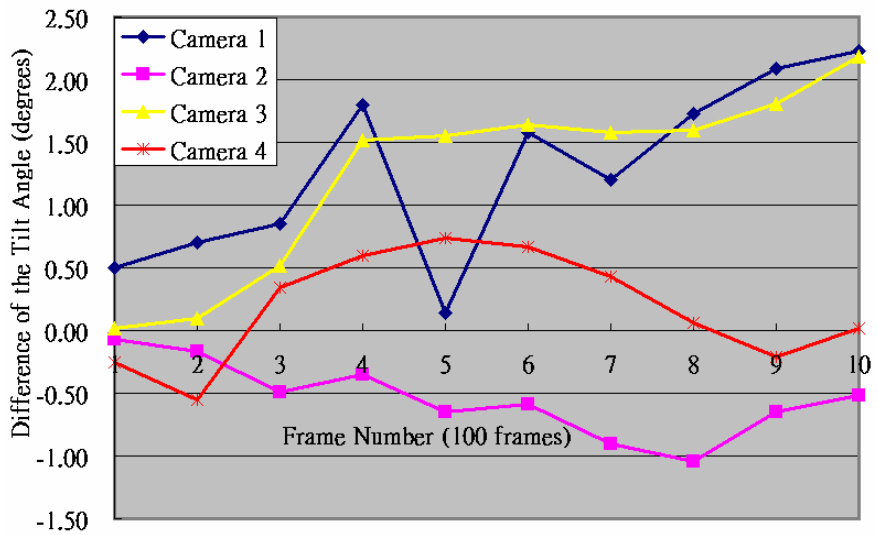
As cameras began to pan and tilt, we extracted 50 prominent feature points from each of these four initial images and tracked these feature points by the KLT method. Based on (4.9) and (4.10), we performed dynamic calibration for every image pair. In our experiment, we calibrated six camera pairs {Camera-k, Camera-k'}, with $\{k, k'\} \in \{\{2, 1\}; \{2, 3\}; \{2, 4\}; \{4, 1\}; \{4, 3\}; \{1, 3\}\}$ and averaged the calibration results for each camera.

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. Figure 4.13 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 beginning of the experiment. In Fig. 4.13, it shows that the differences gradually increase when the frame number increases. However, the deviation at the 1000th frame is still acceptable and is within the range of ± 3 degrees. 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. 4.14.

Furthermore, if we fixed one of the four cameras while let the other three cameras pan and tilt freely, it turns out the results of dynamic calibration become even more reliable, as shown in Fig. 4.15. Now the differences of pan angles and tilt angles are within the range of ± 1.5 degrees. Besides, the differences do not gradually increase this time.



(a)

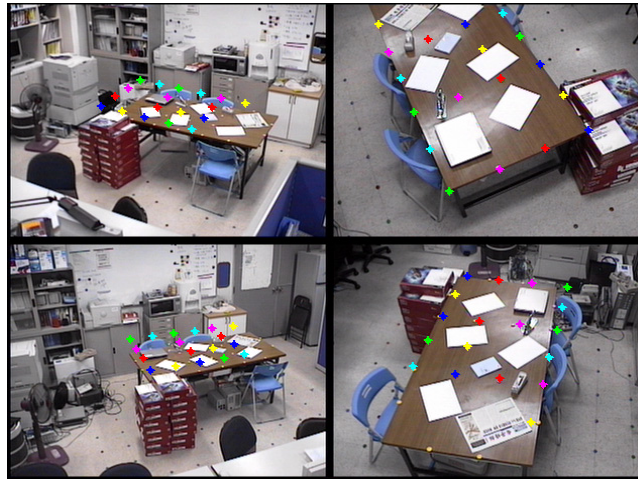


(b)

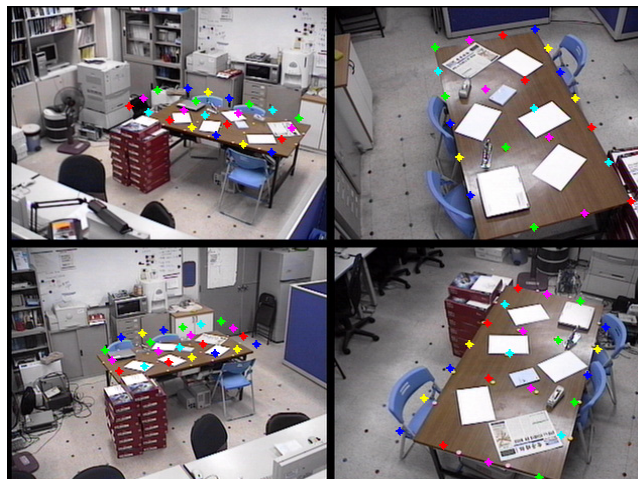
Fig. 4.13 (a) Differences of the pan angles between the dynamic calibration results and the static calibration results. (b) Differences of the tilt angles between the dynamic calibration results and the static calibration results.



(a)

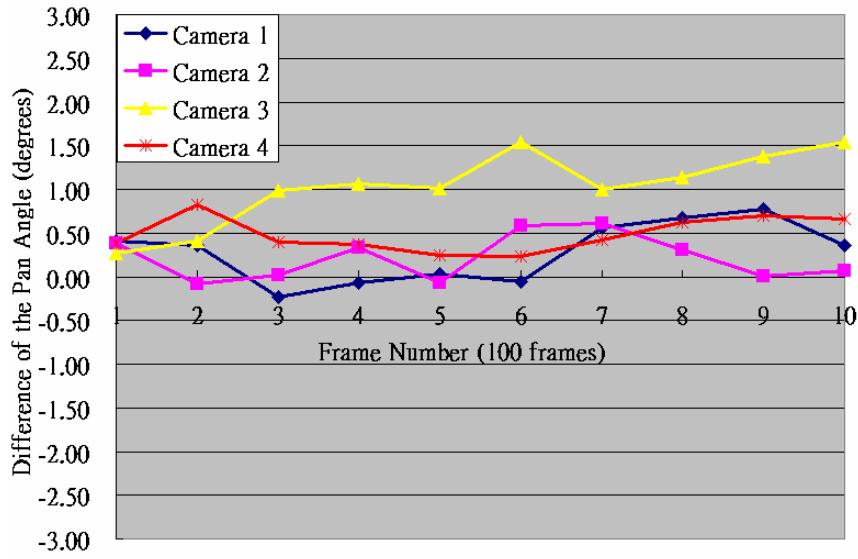


(b)

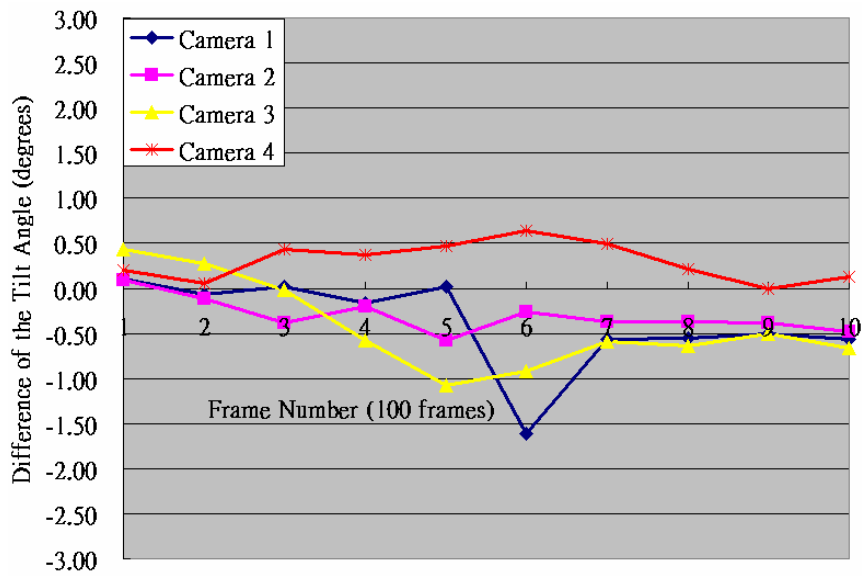


(c)

Fig. 4.14 Evaluations of dynamical calibration at (a) the 300th frame, (b) the 600th frame, and (c) the 1000th frame.



(a)



(b)

Fig. 4.15 (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.

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. 4.16 shows a sample of the captured sequence. In Fig. 4.17, 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



Fig. 4.16 One sample of the test sequence with the presence of a moving person.



Fig. 4.17 Evaluated corresponding relationship of the 1000th frame in the test sequence with a moving person.

CHAPTER 5

Conclusions

In this dissertation, we present two new and efficient pose calibration techniques: 1) the static calibration for multiple cameras based on the back-projections of simple objects lying on the same plane, and 2) the dynamic calibration for multiple cameras with no complicated point correspondence technique.

In the problem of static calibration for multiple cameras, we infer the relative positioning and orientation among multiple cameras. We first deduced the 3D-to-2D coordinate transformation in terms of the tilt angle of a camera. After having established the 3D-to-2D transformation, the tilt angle and altitude of each camera are estimated based on the observation of some simple objects lying on a horizontal plane. With the estimated tilt angles and altitudes, the relative orientations among multiple cameras can be easily obtained by comparing the back-projected world coordinates of some common vectors in the 3-D space. If compared with these conventional calibration approaches which extract the homography matrix and the rotation matrix, our approach offers apprehensible geometric sense and can simplify the calibration

process. No coordinated calibration pattern is needed and the computational load is light. In this dissertation, the sensitivity analysis with respect to parameter fluctuations and measurement errors is also discussed. Both mathematical analysis and computer simulation results are shown to verify our analysis. Experiment results over real images have demonstrated the efficiency and feasibility of this approach.

In the problem of dynamic calibration for multiple cameras, we added the pan angle factor on the mapping between a horizontal plane in the 3-D space and the 2-D image plane on a panned and tilted camera. Based on the mapping, 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 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. The sensitivity analysis of our dynamic calibration algorithm with respect to measurement errors and fluctuations in previous estimations is also discussed mathematically. From the simulation results, the estimation errors of pan and tilt angle changes are proved to be acceptable in real cases. The efficiency and feasibility of this approach has been demonstrated in some experiments over real scenery.

In this dissertation, we adopt a system model that is general enough to fit for a large class of surveillance systems with multiple cameras. Both our static and dynamic calibration methods do not require particular system setup or specific calibration patterns. In some sense, our static calibration can be thought to have decomposed the computation of homography matrix into two simple calibration processes so that the computational load becomes lighter for the calibration of multiple cameras. In addition, the major advantage of our dynamic calibration is that

no complicated correspondence of feature points is needed. Hence, our calibration methods can be well applied to a wide-range surveillance system with multiple cameras. However, in this thesis, we do not combine our calibration algorithm into the related applications of surveillance systems with multiple cameras. The calibration results would offer useful three-dimensional information for surveillance applications, such as object tracking or 3-D positioning. Besides, the zooming effect is not discussed in this dissertation. It should be worthwhile to study these two topics in the future.





Bibliography

- [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 Journal of Robotics and Automation*, vol. 3, issue 4, pp. 323–344, Aug. 1987.
- [2] O. Faugeras, *Three-Dimensional Computer Vision*, The MIT Press Cambridge, Massachusetts London, England, 1993, pp.230–240.
- [3] David A. Forsyth and Jean Ponce, *Computer Vision: A Modern Approach*, Prentice Hall, 2003.
- [4] Zhengyou Zhang, "Flexible Camera Calibration by Viewing a Plane from Unknown Orientations," *The Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 1, pp. 666–673, Sept. 1999.
- [5] Guo-Qing Wei and Song De Ma, "A Complete Two-Plane Camera Calibration Method and Experimental Comparisons," *In Proc. Fourth International Conference on Computer Vision*, pp. 439–446, May 1993.
- [6] D. Oberkampf, D. F. DeMenthon and L. S. Davis, "Iterative Pose Estimation Using Coplanar Points," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 626-627, June 1993.
- [7] Peter F. Sturm and Stephen J. Maybank, "On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 432-437, June 1999.
- [8] T. Kanade, P.J. Narayanan, and P.W Rander, "Virtualized Reality: Constructing Virtual Worlds from real scenes," *IEEE Multimedia*, vol. 4, issue 1, pp. 34-47, May, 1997.
- [9] Charles Wiles and Allan Davison, "Calibrating a Multi-Camera System for 3D Modelling," *In Proceedings. IEEE Workshop on Multi-View Modeling and Analysis of Visual Scenes*, pp. 29–36, June 1999.
- [10] Peter Sturm, "Algorithms for Plane-Based Pose Estimation," *In Proceedings. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, pp.

- 706-711, June 2000.
- [11] T. Ueshiba and F. Tomita, "Plane-Based Calibration Algorithm for Multi-Camera Systems via Factorization of Homography Matrices," *Proceedings of Ninth IEEE International Conference on Computer Vision*, vol. 2, pp. 966–973, Oct. 2003.
- [12] Zhengyou Zhang, "Camera Calibration with One-Dimensional Objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, issue 7, pp. 892–899, July 2004.
- [13] P. Hammarstedt, P. Sturm and A. Heyden, "Degenerate cases and closed-form solutions for camera calibration with one-dimensional objects," *Tenth IEEE International Conference on Computer Vision*, vol. 1, pp. 317-324, Oct. 2005.
- [14] S. J. Maybank and O. D. Faugeras, "A Theory of a Moving Camera," *Int. Journal of Computer Vision*, 8(2), pp. 123-151, Aug. 1992.
- [15] R. Hartley, "Euclidean Reconstruction from Uncalibrated Views," *Int. Journal of Applications of Invariance in Computer Vision*, Springer-Verlag, pp. 237-256, 1994.
- [16] A. Heyden and K. Astrom, "Euclidean Reconstruction from Constant Intrinsic Parameters," *Proc. of the 13th International Conference on Pattern Recognition*, pp. 339-343, 1996.
- [17] M. Pollefeys, L. V. Gool and A. Oosterlinck, "The Modulus Constraint: A new Constraint for Self-Calibration," *Proc. of the 13th International Conference on Pattern Recognition*, pp. 349-353, 1996.
- [18] Q. T. Luong and O. D. Faugeras, "Self-calibration of a moving camera from point correspondences and fundamental matrices," *Int. Journal of Computer Vision*, 22(3), pp. 261-289, 1997.
- [19] M. Pollefeys and L. V. Gool, "Self-Calibration from the Absolute Conic on the Plane at Infinity," *Proc. Computer Analysis of Image and Patterns*, vol. 1396, Springer-Verlag, pp. 175-182, 1997.
- [20] B. Triggs, "Autocalibration and the Absolute Quadric," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 609-614, 1997.
- [21] M. A. Lourakis and R. Deriche, "Camera Self-calibration Using the Singular Value Decomposition of the Fundamental Matrix," *In the 4th Asian Conference on Computer Vision*, vol. I, pp. 403-408, Jan. 2000.
- [22] R. Hartley, "Self-calibration from Multiple Views with a Rotating Camera," *Lecture Notes in Computer Science*, vol. 800-801, Springer-Verlag, pp. 471-478, 1994.
- [23] T. Moons, L. V. Gool, M. Proesmans and E. Pauwels, "Affine Reconstruction from Perspective Image Pairs with a Relative Object-camera Translation in

- between,” *IEEE Trans. PAMI*, vol. 18, no. 1, pp. 77-83, Jan. 1996.
- [24] M. Armstrong, A. Zisserman and R. Hartley, “Euclidean Reconstruction from Image Triplets,” *ECCV*, vol. 1064, Springer-Verlag, pp. 3-16, 1996.
- [25] O. Faugeras, L. Quan and P. Sturm, “Self-calibration of a 1-D Projective Camera and Its Application to the Self-calibration of a 2-D Projective Camera,” *IEEE Trans. PAMI*, 22(10), pp. 1179-1185, Oct. 2001.
- [26] L. Wang, S.B. Kang, H. Y. Shum, and G. Xu, “Error Analysis of Pure Rotation-based Self-calibration,” *Proc. 8th Int. conf. on Computer Vision*, pp. I: 464-471, July 2001.
- [27] M. Pollefeys, L. V. Gool and M. Proesmans, “Euclidean 3D Reconstruction from Image Sequences with Variable Focal Lengths,” *ECCV*, vol. 1064, pp. 31-42, 1996.
- [28] B. Rousso and E. Shilat, “Varying Focal Length Self-calibration and Pose Estimation from Two Images,” *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 469-475, June 1998.
- [29] M. Pollefeys, R. Koch, and L. V. Gool, “Self-calibration and Metric Reconstruction in spite of Varying and Unknown Intrinsic Camera Parameters,” *Int. Journal of Computer Vision*, 32(1), pp. 7-25, Aug. 1999.
- [30] F. Kahl, B. Triggs and K. Astrom, “Critical Motions for Auto-calibration When Some Intrinsic Parameters can Vary,” *Journal of Mathematical Image and Vision*, 13(2), pp. 131-146, Oct. 2000.
- [31] L. De Agapito, R. Hartley and E. Hayman, “Linear Calibration of a Rotating and Zooming Camera,” *Proc. IEEE CVPR*, pp. 15-21, 1999.
- [32] Y. Seo and K. Hong, “About Self-calibration of a Rotating and Zooming Camera: Theory and Practice,” *In Proc. 7th Int. Conf. on Computer Vision*, vol. 1, pp. 183-189, Sept. 1999.
- [33] B. Tordoff and D. W. Murray, “Violating Rotating Camera Geometry: the Effect of Radial Distortion on Self-calibration,” *Proc. of Int. Conf. on Pattern Recognition*, pp. 1423-1427, 2000.
- [34] L. De Agapito, E. Hayman and I. Reid, “Self-calibration of a Rotating and Zooming Camera,” *Int. Journal of Computer Vision*, 45(2), pp. 107-127, 2001.
- [35] H. Kim and K. S. Hong, “A Practical Self-calibration Method of Rotating and Zooming Cameras,” *IEEE Proc. on Vision, Image and Signal*, vol. 148(5), pp. 349-355, Oct. 2001.
- [36] Elsayed E. Hemayed, “A Survey of Camera Self-Calibration”, *In Proc. IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 351–357, July 2003.
- [37] E. Kruppa, “Zur Ermittlung eines Objektes aus zwei Perspektiven mit innerer

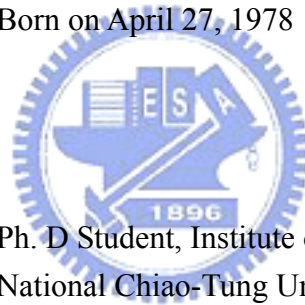
- Orientierung,” Sitz.-Ber. Akad. Wiss., Wien, math. Naturw. KL., Abt. IIa., 122: 1939-1948, 1913.
- [38] A. Jain, D. Kopell, K. Kakligian, and Y-F Wang, “Using Stationary-Dynamic Camera Assemblies for Wide-Area Video Surveillance and Selective Attention,” *IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 537–544, June 2006.
- [39] G. Schweighofer and A. Pinz, “Robust Pose Estimation from a Planar Target,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2024-2030, December 2006.
- [40] K.-T. Song and J.-C. Tai, “Dynamic Calibration of Pan–Tilt–Zoom Cameras for Traffic Monitoring,” *IEEE Transactions on Systems, Man and Cybernetics, Part B*, vol. 36, issue 5, pp. 1091–1103, Oct. 2006.
- [41] Y. Li, F. Zhu, Y. Ai and F.-Y. Wang, “On Automatic and Dynamic Camera Calibration Based on Traffic Visual Surveillance,” *IEEE Symposium on Intelligent Vehicles*, pp. 358-363, June 2007.
- [42] C.T. Huang and O.R. Mitchell, “Dynamic Camera Calibration,” in *Proc. Int. Symposium on Computer Vision*, pp. 169–174, Nov. 1995.
- [43] B. Zhang, Y. F. Li and Y. H. Wu, “Self-recalibration of a Structured Light System via Plane-Based Homography,” *Pattern Recognition*, vol. 40, Issue 4, pp. 1368-1377, April 2007.
- [44] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge, 2nd edition, 2003.
- [45] J. Shi and C. Tomasi, “Good Features to Track,” In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 593 – 600, June 1994.
- [46] Ser-Nam Lim, L.S. Davis, and Ahmed Elgammal, “A Scalable Image-Based Multi-Camera Visual Surveillance System”, In *Proc. IEEE Conference on Advanced Video and Signal Based Surveillance*, pp. 205–212, July 2003.

Curriculum Vitae

Name:

I-Hsien Chen (陳宜賢)

Born on April 27, 1978

**Education:**

Sep. 2001 ~ Jan. 2008

Ph. D Student, Institute of Electronics,
National Chiao-Tung University, Hsin-chu, Taiwan

Sep. 2000 ~ June 2001

M.S. Student, Institute of Electronics,
National Chiao-Tung University, Hsin-chu, Taiwan

Sep. 1996 ~ June 2000

B.S. Student, Institute of Electronics,
National Chiao-Tung University, Hsin-chu, Taiwan

Work Experience:

Aug. 2004 ~ Jan. 2005

Lecturer in Yuanpei University (元培科學技術大學)

Publications:**Dissertation:**

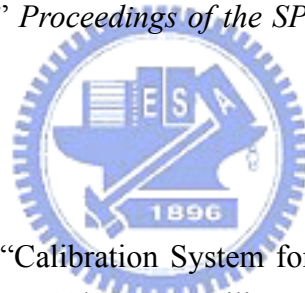
Ph. D Dissertation: Static and Dynamic Calibration of Multiple Cameras (多台攝

Journal Paper:

- [1] I-H Chen and S-J Wang, “An efficient approach for the calibration of multiple PTZ cameras,” *IEEE Transactions on Automation Science and Engineering*, vol. 4, no. 2, pp. 286 – 293, April 2007.
- [2] I-H Chen and S-J Wang, “An efficient approach for dynamic calibration of multiple cameras,” *IEEE Transactions on Automation Science and Engineering*, to be published.

Conference Papers:

- [1] I-H Chen and S-J Wang, “Efficient vision-based calibration for visual surveillance systems with multiple PTZ cameras,” *IEEE Conference on Computer Vision Systems*, pp.24-24. Jan. 2006
- [2] I-H Chen and S-J Wang, “A vision-based approach to extracting the tilt angle and altitude of a PTZ camera,” *Proceedings of the SPIE*, vol. 6066, pp.606609-1 – 9, Jan. 2006.



Filed U.S. Patents:

- [1] I-H Chen and S-J Wang, “Calibration System for Image Capture Apparatus and Method Thereof”, Filing No.: 11/483,542, Filing Date: July 11, 2006.

Filed Taiwan Patents:

- [1] 陳宜賢、王聖智， “影像擷取裝置之校正系統及其方法” ， 申請案號：095105830， 申請日：95.02.21.