Keeping a Pan-Tilt-Zoom Camera Calibrated

Ziyan Wu, *Student Member, IEEE,* and Richard J. Radke, *Senior Member, IEEE*

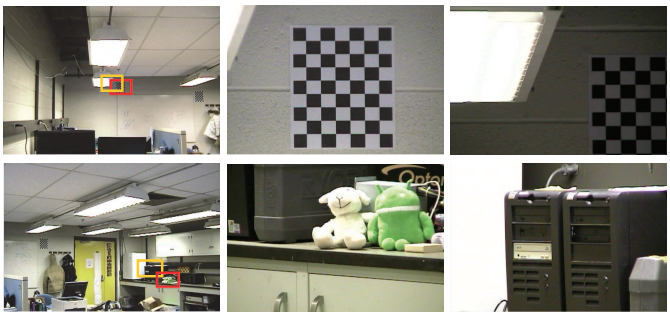
Abstract—Pan-tilt-zoom (PTZ) cameras are pervasive in modern surveillance systems. However, we demonstrate that the (pan, tilt) coordinates reported by PTZ cameras become inaccurate after many hours of operation, endangering tracking and 3D localization algorithms that rely on the accuracy of such values. To solve this problem, we propose a complete model for a pan-tilt-zoom camera that explicitly reflects how focal length and lens distortion vary as a function of zoom scale. We show how the parameters of this model can be quickly and accurately estimated using a series of simple initialization steps followed by a nonlinear optimization. Our method requires only ten images to achieve accurate calibration results. Next, we show how the calibration parameters can be maintained using a one-shot dynamic correction process; this ensures that the camera returns the same field of view every time the user requests a given (pan, tilt, zoom), even after hundreds of hours of operation. The dynamic calibration algorithm is based on matching the current image against a stored feature library created at the time the PTZ camera is mounted. We evaluate the calibration and dynamic correction algorithms on both experimental and real-world datasets, demonstrating the effectiveness of the techniques.

Index Terms—Pan-Tilt-Zoom, Calibration, Dynamic Correction

1 Introduction

Most modern wide-area camera networks make extensive use of pan-tilt-zoom (PTZ) cameras. For example, a large airport typically contains hundreds, or even thousands, of cameras, many of which have PTZ capability. In practice, these cameras move along predetermined paths or are controlled by an operator using a graphical or joystick interface. However, since such cameras are in constant motion, accumulated errors from imprecise mechanisms, random noise, and power cycling render any calibration in absolute world coordinates useless after many hours of continuous operation. For example, Figure 1 illustrates an example in which a PTZ camera is directed to the same absolute (pan, tilt, zoom) coordinates both before and after 36 hours of continuous operation. We can see that the images are quite different, which means that these absolute coordinates are virtually meaningless in a real-world scenario. Consequently, in practice, operators direct PTZ cameras almost exclusively in a relative, human-in-the-loop mode, such as using on-screen arrow keys to manually track a target. While high-quality PTZ cameras do exist that could mitigate such issues, most of the PTZ cameras used in security and surveillance are relatively cheap and suffer from the mechanical problems described here.

This paper has several contributions. First, we characterize and measure the sources of error in real PTZ cameras to demonstrate that a non-negligible problem exists. Second, we propose a complete model for a PTZ camera in which all internal parameters are a function of the *zoom scale*, a number in arbitrary units that defines the field of view subtended by the camera. Third, we present a novel method to calibrate the proposed PTZ camera model. This method requires no information other than features from the scene, and initial estimates of all parameters of the model can be computed easily prior to a non-linear optimization. Finally, we show how the calibration of a PTZ camera can be automatically maintained after this initial calibration, so that when a user directs the camera to given (pan, tilt, zoom) coordinates, the same field of view is always attained. This online maintenance requires no special calibration object, and instead uses a library of natural features detected during the initial calibration. As a consequence of our proposed algorithms, the absolute PTZ coordinates for a given camera can be trusted to be accurate, leading to improved performance on important tasks like the 3D triangulation of a tracked target.



(a) (b) (c)

Fig. 1. A PTZ camera acquires two images at the same absolute (pan, tilt, zoom) coordinates both before and after 36 hours of continuous random operation. (a) The red rectangles indicate the initial images and the yellow rectangles indicate the final images. (b) Close-ups of the acquired initial images. (c) Close-ups of the acquired final images.

• *Z. Wu and R.J. Radke are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, 12180. E-mail: wuz5@rpi.edu, rjradke@ecse.rpi.edu*

An earlier version of part of the work in this paper appeared in [26]. This material is based upon work supported by the U.S. Department of Homeland Security under Award Number 2008-ST-061-ED0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied of the U.S. Department of Homeland Security.

2 Related Work

Since PTZ cameras are usually used for video surveillance, self-calibration technologies are often adapted, such as the methods using pure rotation proposed by de Agapito et al. [5], [6] and Hartley [9]. Davis et al. [4] proposed an improved model of camera motion in which pan and tilt are modeled as rotations around arbitrary axes in space. However, this method relies on a well-calibrated tracking system.

Sinha and Pollefeys [22] proposed a calibration method for PTZ cameras that we compare our algorithm against later in the paper. The camera is first calibrated at the lowest zoom level and then the intrinsic parameters are computed for an increasing zoom sequence. Since the zoom sequence calibration is discrete, piecewise linear interpolation is used to compute intrinsic parameters for an arbitrary zoom level. However, only the focal length is calibrated in this method, and many small steps may be required to mitigate noise, which makes the calibration time-consuming.

Sarkis et al. [20] introduced a technique for modeling intrinsic parameters as a function of lens settings based on moving least squares. However, this method is computationally demanding and may be susceptible to over-fitting. Ashraf and Foroosh [1] presented a self-calibration method for PTZ cameras with non-overlapping fields of view (FOVs). Some calibration methods for surveillance cameras are based on vanishing points obtained by extracting parallel lines in the scene [11], [18].

Lim et al. [15] introduced an image-based model for camera control by tracking an object with multiple cameras and relating the trajectories in the image plane to the rotations of the camera.

However, none of these approaches use a complete model of a PTZ camera. Some ignore lens distortion, while others assume that lens distortion is estimated at only one zoom scale and model its variation by a magnification factor [3]. The PTZ camera model proposed in this paper solves this problem by explicitly posing models for focal length and lens distortion as a function of zoom scale and efficiently estimating the parameters of these models.

Even if a PTZ camera is initially well-calibrated, frequent rotation and zooming over many hours of operation can make the calibration very inaccurate (Figure 1), which can induce serious error in tracking applications. This is a critical issue for PTZ-camera-based video surveillance systems. Song and Tai [23] proposed a dynamic calibration method for a PTZ camera by estimating a vanishing point from a set of parallel lanes of known width. Similarly, Schoepflin and Dailey [21] proposed a dynamic calibration method with a simplified camera model by extracting the vanishing point of a roadway. However, the applications of such methods are limited to environments featuring long straight lines that can be extracted with high precision. The dynamic correction method proposed in this paper has no assumptions about the imaged environment.

3 Sources of Error

In this section, we characterize and measure the sources of error in real PTZ cameras. These sources include mechanical offsets in the cameras' stepper motors, random errors in the reported (pan, tilt, zoom) coordinates, accumulated errors in these coordinates that increase with extended continuous operation, and unpredictable jumps in error that occur when the power to the camera is cycled. These types of error combine to make open-loop calibration of PTZ cameras inherently inaccurate, especially at high zoom levels.

3.1 Mechanical Error

PTZ camera controls are usually based on stepper motors, the accuracy of which range from 0.009 to 1 degrees. From our observations, the mechanical error depends on the camera's manufacturing quality and can be compensated for. In order to quantify the error, we conducted a simplified experiment. A PTZ camera is first calibrated with a checkerboard target [27] at a fixed (pan, tilt, zoom) setting. We let $\mathbf{K}$ be the internal parameter matrix of the PTZ camera, after accounting for lens distortion. The camera is then directed to purely pan (or tilt) with a constant step length of Δp (or Δt), each position corresponding to a rotation matrix $\mathbf{R}$. A pair of images is acquired before and after each rotation, which are related by a homography matrix that can be estimated based on matched features, denoted $\widehat{\mathbf{H}}$. On the other hand, the ideal homography matrix induced by the rotation can be computed as $\mathbf{H} = \mathbf{KRK^{-1}}$. Thus, we can represent the error between the actual and desired rotations by the rotation matrix

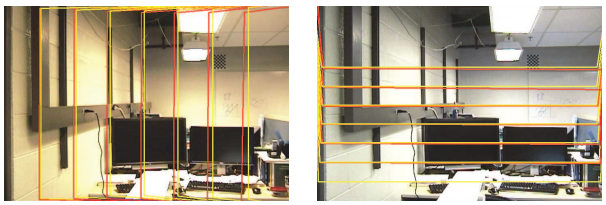
$$\mathbf{R_e} = \mathbf{K^{-1}H\widehat{H}^{-1}K} \tag{1}$$

The rotation errors in both the pan and tilt axes, e_p and e_t, are then extracted from this matrix. Figure 2 shows two examples illustrating the mechanical error in PTZ rotation. The error increases with rotation angle, which is quantified in Figure 3 for two PTZ camera models, the Axis AX213PTZ and Axis AX233D. The relationship between the rotation error and rotation angle is close to linear. Since the PTZ controls are based on stepper motors, no hysteretic phenomenon is observed.

3.2 Random Error

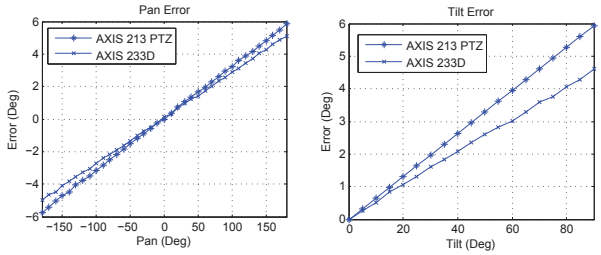
In addition to mechanical errors, we also observed non-negligible random errors, as illustrated in Figure 4. We compared two groups of images acquired at the same position before and after 30 minutes of random motion, and observed 5–7 pixels of error at the edges in the images.

We next conducted another experiment to observe the error over a longer time span of 200 hours, summarized



(a) (b)

Fig. 2. Sample images illustrating mechanical errors when panning and tilting the camera. Yellow trapezoids indicate the ideal field of view of the camera while red trapezoids indicate the actual field of view. (a) Step panning at 5 degrees per step. (b) Step tilting at 3 degrees per step.



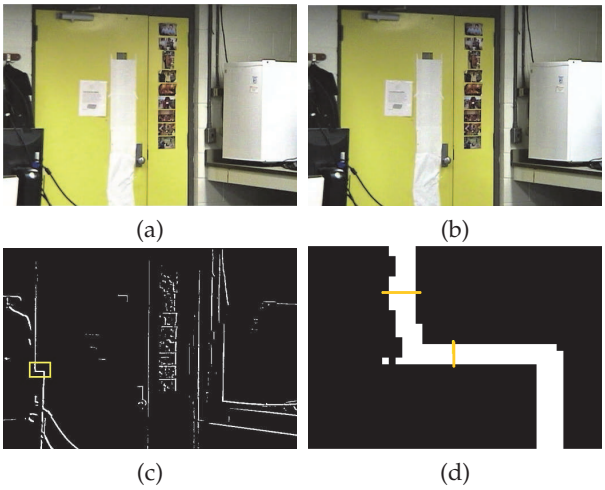
(a) (b)

Fig. 3. Errors as a function of (a) pan and (b) tilt angle for two models of PTZ cameras.

in Figure 5. We placed checkerboard targets at four monitored positions in the scene, corresponding to fixed (pan, tilt, zoom) coordinates. A PTZ camera was programmed to move to a random (pan, tilt, zoom) setting every 30 seconds. After every hour, the camera randomly chooses one of the four monitored positions and acquires an image, from which the corners of the target are automatically extracted and compared to the reference image. The error at time t is defined as $e(t) = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{x}_i^e(t) - \mathbf{x}_i^r\|$, in which N is the number of corners on the target, and $\mathbf{x}_i^e(t)$ and $\mathbf{x}_i^r$ are the image coordinates of the i^{th} corner in the image after random motion and in the reference image respectively. From the results, we can see that the error increases with zoom level, to a maximum of 38 pixels of error recorded at position A (the maximum zoom).

From Figure 5, we can see that besides the random error, there is a raising trend in the errors over time, i.e., an accumulation of error. Furthermore, we were surprised to find that serious error is introduced every time we restarted the PTZ camera — that is, the "home" position changes significantly between power cycles. The third row of Figure 5 shows the images acquired at the monitored positions after restarting the PTZ camera; in positions A and B, the targets have nearly disappeared from the field of view.

These factors — mechanical, random, accumulated, and power-cycling errors — all argue that a PTZ camera cannot simply be calibrated once and forgotten. Instead, the calibration should be constantly maintained over time if absolute (pan, tilt, zoom) coordinates are to have



(a) (b) (c) (d)

Fig. 4. Experiment illustrating random errors in PTZ cameras. (a) The original image. (b) The image after 30 minutes of random panning, tilting, and zooming. (c) The differences between the images in (a) and (b). (d) Zoomed-in details of vertical and horizontal edges in (c).

any meaning (e.g., used for active control of a camera based on a computer vision algorithm). The rest of this paper addresses a complete model for PTZ cameras, and methods for both initial calibration and calibration maintenance.

4 A Complete Model for a PTZ Camera

In this section, we propose a complete model for a PTZ camera. The common pinhole camera model with lens distortion is not sufficient to represent a PTZ camera, since we require a very important *zoom scale* parameter that greatly affects the other parameters. Our proposed model is simple, continuous as a function of zoom scale, and can be accurately calibrated with a small number of images.

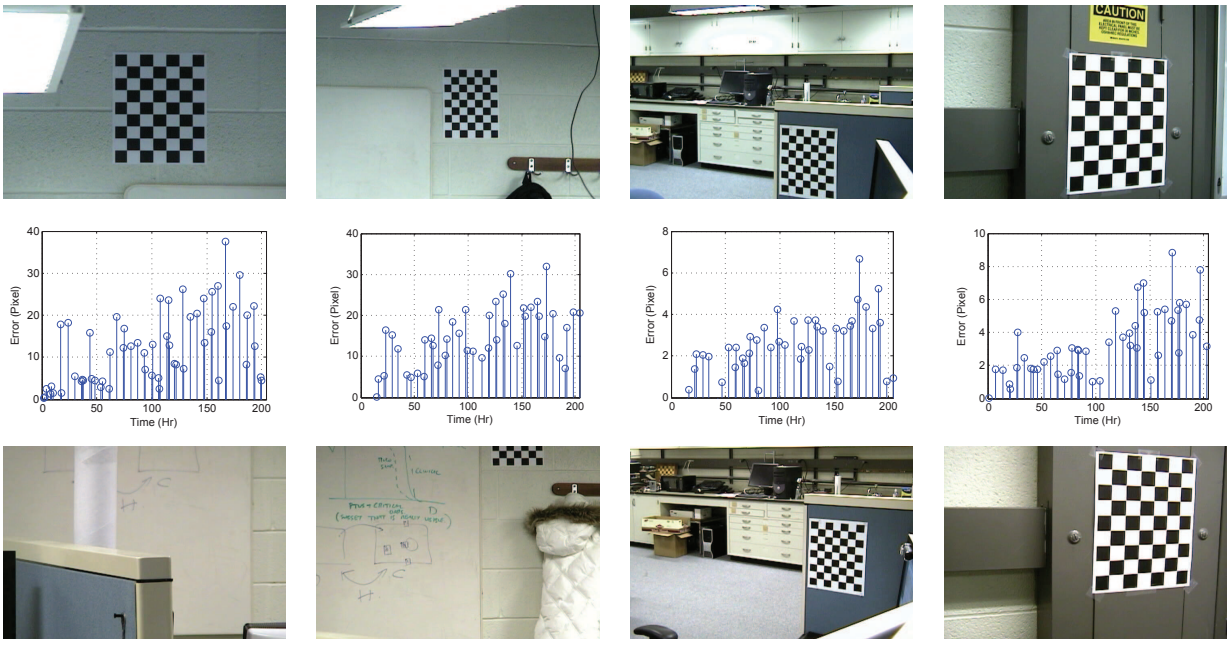
4.1 Camera Model

We assume the standard model for the internal parameters of a camera without lens distortion at a fixed zoom scale, namely a 3×3 calibration matrix $\mathbf{K}(z)$ given by

$$\mathbf{K}(z) = \begin{bmatrix} f_x(z) & 0 & c_x \\ 0 & f_y(z) & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

where $f_x(z), f_y(z)$ are the focal length in units of x and y pixel dimension and $\mathbf{c} = (c_x, c_y)$ is the principal point. We assume the principal point is fixed and the pixel skew at all zoom scales is 0 in this paper. The relationship between a 3D point $\mathbf{X}$ and its image projection $\mathbf{x}$ is given by $\mathbf{x} \sim \mathbf{K}(z)\ [\mathbf{R} \mid \mathbf{t}]\ \mathbf{X}$, in which $\mathbf{R}$ and $\mathbf{t}$ are the rotation matrix and translation vector that specify the external parameters of the camera, and $\mathbf{x}$ and $\mathbf{X}$ are the homogeneous coordinates of the image projection and 3D point, respectively.

Since wide-angle lenses are commonly used in PTZ cameras, we must also consider lens distortion. While



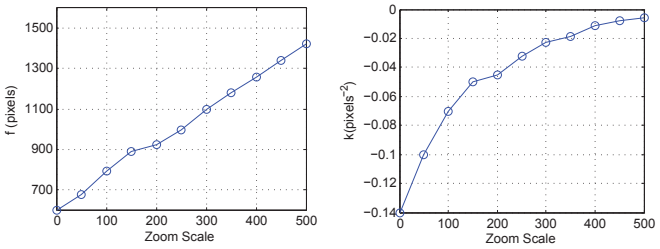
(a) Pos.A (1,0,8000) (b) Pos.B (16,-2,7000) (c) Pos.C (60,-10,1000) (d) Pos.D (-70,-7,0)

Fig. 5. Repeatability experiment using four different monitored (pan,tilt,zoom) coordinates over the course of 200 hours. The first row shows the original images. The second row shows the error in pixels as a function of time over the 200-hour experiment. The third row shows the monitored positions after power-cycling the camera, indicating serious errors.

polynomial models are often used to describe modest radial distortion, we use the division model proposed by Fitzgibbon [7], which is able to express high distortion at lower order. Hence, we model the radial distortion using a single-parameter division model, that is,

$$\widetilde{\mathbf{x}}_u = \frac{\widetilde{\mathbf{x}}_d}{1 + \kappa(z)\|\widetilde{\mathbf{x}}_d\|^2} \tag{3}$$

in which $\mathbf{x}_u$ is the undistorted image coordinate, $\mathbf{x}_d$ is the corresponding distorted image coordinate, $\widetilde{\mathbf{x}}_{\{u,d\}} = \mathbf{x}_{\{u,d\}} - \mathbf{c}$ are the normalized image coordinates, and $\kappa(z)$ is the distortion coefficient, which varies with zoom scale.



(a) (b)

Fig. 6. Intrinsic parameters as a function of zoom scale. (a) Focal length f_x vs. zoom scale. (b) Lens distortion coefficient κ as a function of zoom scale.

4.2 The Effects of Zooming

We investigated the change in intrinsic parameters with respect to different zoom scales, by calibrating a PTZ camera with resolution 704×480 at 10 zoom scales ranging from 0 to 500 using Zhang's calibration method [27]. We note that the zoom scale is measured in arbitrary units specified by the camera manufacturer. At each zoom scale, the camera was calibrated 20 times. The average relationships between the focal length f_x and lens distortion coefficient κ with respect to zoom scale are illustrated in Figure 6.

We observed that the principal point (c_x, c_y) is stable with respect to zoom scale and also consistent with the zooming center, so we drop its dependence on z in Equation (2). From Figure 6 we can see that it is reasonable to create continuous models for the intrinsic parameters of a PTZ camera as a function of zoom scale. Let $f_x(0)$, $f_y(0)$, $\kappa(0)$ and (c_x, c_y) be the calibrated intrinsic parameters at the initial zoom scale, which is $z = 0$. In practice, the relationship between the focal length and zoom scale is nearly linear. However, it is safer to consider the relationship to be up to second order, since the zooming control quality varies for different cameras. Thus, we propose the model for focal length as:

$$f_x(z) = f_x(0) + a_f z + b_f z^2 \tag{4}$$

We note that $f_y(z) = \alpha f_x(z)$, where α is the fixed pixel aspect ratio. We observed that the lens distortion coefficient κ can be modeled as a function of zoom scale z by

$$\kappa(z) = \kappa(0) + \frac{a_\kappa}{(f(z) + b_\kappa)^2} \tag{5}$$

In practice, the quadratic term in z in the denominator dominates the higher-order terms.

5 Fully Calibrating a PTZ Camera

We now present a novel method for the complete self-calibration of a PTZ camera using the model proposed in the previous section. The method proceeds in several steps, described in each subsection, using a minimal amount of information to initially estimate each parameter of the model. In particular, each step generally uses either:

- a "zoom set" $\mathcal{Z}$ of $M \geq 3$ images taken at the same pan-tilt position and several different zoom scales $\{z_1, \ldots, z_M\}$. One of these zoom scales should be the lowest (i.e., widest angle) zoom scale z_0. A pan-tilt position with a large number of features should be used.
- a "pan-tilt set" $\mathcal{PT}$ of $N \geq 3$ images taken at several different pan-tilt positions and the lowest (i.e., widest angle) zoom scale z_0. The images should be in general position; that is, they should avoid critical motions such as pure panning.

Each initial estimate is obtained using a simple linear-least-squares problem using feature point correspondences either between images in $\mathcal{Z}$ or images in $\mathcal{PT}$. In practice, we use the SURF feature and descriptor [2] to produce a large number of high-quality correspondences that form the basis for the estimation problems (see also Section 6.1).

These initial estimates are then jointly refined using nonlinear optimization (i.e., Levenberg-Marquardt) to produce the final calibration parameters, consisting of: the principal point, the parameters $\{a_f, b_f, f_x(0), f_y(0), a_\kappa, b_\kappa, \kappa(0)\}$ that specify the focal length and lens distortion as a function of zoom scale, and the linear coefficients $\{\beta_p, \beta_t\}$ that account for the mechanical error in pan and tilt measurements illustrated in Figure 3. The overall algorithm is illustrated in the top half of Figure 7. In Section 6, we will describe our approach to keeping a PTZ camera calibrated after many hours of operation, illustrated in the bottom half of Figure 7.

5.1 Principal Point

In this section, we use the zoom set $\mathcal{Z}$ to estimate the principal point $\mathbf{c} = (c_x, c_y)$. In our experiments, we observed that the center of zoom is well-modeled as the intersection of the optical axis with the image plane (i.e., the principal point) [13], [14], and that the principal point (c_x, c_y) is constant as a function of zoom scale.

We begin by considering two projections (x, y) and (x', y') of the same 3D scene point in different images of $\mathcal{Z}$, corresponding to zoom scales z and z' respectively. The following relationship holds, even in the presence of lens distortion:

$$\frac{x' - c_x}{x - c_x} = \frac{y' - c_y}{y - c_y} = \frac{z'}{z} \tag{6}$$

Hence

$$c_x(y - y') + c_y(x' - x) = x'y - xy' \tag{7}$$

Thus, every correspondence across zoom scales gives one equation in the two unknowns (c_x, c_y). A large number of cross-scale feature correspondences results in a linear least-squares problem of the form

$$\mathbf{A} \begin{bmatrix} c_x \\ c_y \end{bmatrix} = \mathbf{b} \tag{8}$$

We use the resulting estimate $(\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{b}$ as the principal point. From this point forward, all the coordinates are normalized with respect to the principal point, i.e., $\mathbf{x} \leftarrow \mathbf{x} - \mathbf{c}$.

5.2 Lens Distortion

Tordoff and Murray [25] showed that radial lens distortion has a great impact on the accuracy of camera self-calibration. Therefore, in this section, we use two images from the set $\mathcal{PT}$ that have the same tilt angle but different pan angles to estimate the important parameter $\kappa(0)$, the lens distortion coefficient at the lowest scale.

When long, straight lines are present in the scene, radial distortion can be estimated (e.g., [19]); however, we do not wish to impose any restrictions on scene content. Here, we propose a novel approach inspired by Fitzgibbon [7], who introduced a method for linearly estimating division-model lens distortion during the estimation of the fundamental matrix for a moving camera. This approach to simultaneously estimating lens distortion and the fundamental matrix proved to be flexible and reliable in practice [12], [24]. Here, we extend this idea, simultaneously estimating the lens distortion coefficient and the parameters of a homography induced by pure panning.

Consider a feature correspondence $\mathbf{x_u} \leftrightarrow \mathbf{x'_u}$ between two ideal, undistorted images taken by a PTZ camera undergoing pure panning at the lowest zoom scale. Since we assume the camera center to be constant, the feature match is related by a homography, represented as a 3×3 matrix $\mathbf{H}$ that acts on homogeneous image coordinates, $\mathbf{x'_u} \sim \mathbf{H}\mathbf{x_u}$. This can also be expressed as

$$\mathbf{x'_u} \times \mathbf{H}\mathbf{x_u} = 0 \tag{9}$$

We note that for pure panning, $\mathbf{H}$ only has 5 nonzero elements, since $h_{12} = h_{21} = h_{23} = h_{32} = 0$. Now we consider a correspondence between distorted images $\mathbf{x_d} \leftrightarrow \mathbf{x'_d}$; combining (9) with the division model (3) gives

$$(\mathbf{x'_d} + \kappa(0)\mathbf{z'_d}) \times \mathbf{H}\,(\mathbf{x_d} + \kappa(0)\mathbf{z_d}) = 0 \tag{10}$$

where $\mathbf{z_d} = [0,\ 0,\ \|\mathbf{x_d}\|^2]^\top$ and $\mathbf{z'_d}$ is defined similarly. Expanding (10),

$$\begin{aligned} \mathbf{x'_d} \times \mathbf{H}\mathbf{x_d} + \kappa(0)\,(\mathbf{z'_d} \times \mathbf{H}\mathbf{x_d} + \mathbf{x'_d} \times \mathbf{H}\mathbf{z_d}) \\ + \kappa(0)^2\,(\mathbf{z'_d} \times \mathbf{H}\mathbf{z_d}) = 0 \end{aligned} \tag{11}$$

or

$$\left(\mathbf{M_1} + \kappa(0)\mathbf{M_2} + \kappa(0)^2\mathbf{M_3}\right)\mathbf{h} = 0 \tag{12}$$

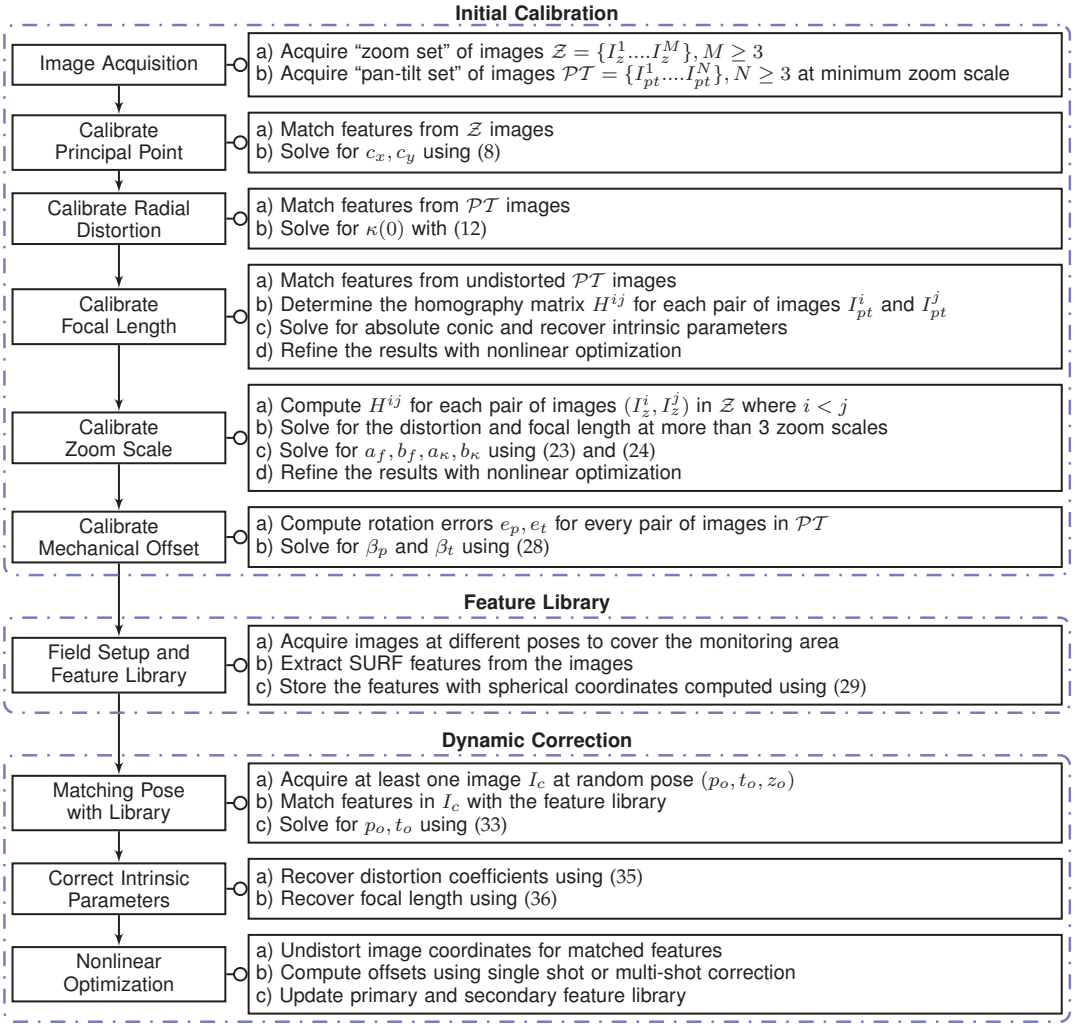


Fig. 7. Outline of the proposed algorithm.

where $\mathbf{h} \in \mathbb{R}^5$ collects the non-zero parameters of $\mathbf{H}$ into a unit-length column vector, and $M_1, M_2, M_3 \in \mathbb{R}^{2\times 5}$ are given by:

$$M_1 = \begin{bmatrix} 0 & 0 & -y' & yx' & y \\ x' & 1 & 0 & -xx' & -x \end{bmatrix}$$
$$M_2 = \begin{bmatrix} 0 & 0 & -sy' & 0 & ys' \\ sx' & s'+s & 0 & 0 & -xs' \end{bmatrix}$$
$$M_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & ss' & 0 & 0 & 0 \end{bmatrix}$$

where $s = \|\mathbf{x_d}\|$ and $s' = \|\mathbf{x'_d}\|$. Thus, every correspondence results in 2 equations in the 5 unknowns of $\mathbf{H}$ and the unknown lens distortion parameter $\kappa(0)$. A large set of correspondences between the image pair results in a polynomial eigenvalue problem in the form of (12). This problem can be solved using Matlab's `polyeig` function applied to 5 of the 6 equations resulting from 3 correspondences. After removing infinite or imaginary eigenvalues, we initially estimate $\kappa(0)$ as the eigenvalue corresponding to the minimizer of $\|\mathbf{x'_u} \times \mathbf{H_i x_u}\|$, where $\mathbf{H_i}$ is formed by reshaping the entries of the corresponding eigenvector. Nonlinear minimization of the objective function

$$F(\kappa(0), \mathbf{h}) = \sum_{n=1}^{N} \left\| \frac{\mathbf{x'_d}^n}{1+\kappa(0)\|\mathbf{x'_d}^n\|} \times \mathbf{H_i} \frac{\mathbf{x_d}^n}{1+\kappa(0)\|\mathbf{x_d}^n\|} \right\| \quad (13)$$

produces a refined estimate of $\kappa(0)$ for the next step.

5.3 Focal Length

Next, we use the set $\mathcal{PT}$ to estimate the focal length at the lowest zoom scale, $f(0)$. In the previous step, we estimated the lens distortion parameter $\kappa(0)$, so we assume that we can generate feature correspondences between undistorted versions of the images in $\mathcal{PT}$. Each such pair is related by a homography, which we can explicitly compute in terms of the internal and external parameters:

$$\mathbf{H} \sim \mathbf{K}(0)\mathbf{R}\mathbf{K}(0)^{-1} \quad (14)$$

where $\mathbf{K}(0)$ is the calibration matrix for the undistorted image at the lowest zoom scale, and $\mathbf{R}$ is the relative

3D rotation of the camera from the home position. Note that all the image coordinates used to compute $\mathbf{H}$ are normalized with respect to the principal point. If we denote the image of the absolute conic [8] at the lowest scale as

$$\omega(\mathbf{0}) = \left(\mathbf{K}(0)\mathbf{K}(0)^\top\right)^{-1} \tag{15}$$

then combining (14) and (15) yields

$$\omega(\mathbf{0}) = \mathbf{H}^{-\top}\omega(0)\mathbf{H}^{-1} \tag{16}$$

Since in this case, $\mathbf{K}(0)$ and $\omega(0)$ are diagonal matrices, we can write (16) in the form $\mathbf{Aw} = \mathbf{0}$ where $\mathbf{A}$ is a 3×3 matrix and $\mathbf{w}$ contains the diagonal elements of $\omega(0)$ rearranged as a 3×1 vector. If we obtain homographies corresponding to several pairs of images in $\mathcal{PT}$, then the system $\mathbf{Aw} = \mathbf{0}$ is overdetermined. The vector $\mathbf{w}$ (and hence $\omega(0)$) can be obtained using the Direct Linear Transform [8]. Then we can estimate

$$f_x(0) = \sqrt{|w_3/w_1|} \quad f_y(0) = \sqrt{|w_3/w_2|} \tag{17}$$

in which w_i is the i^{th} element of $\mathbf{w}$.

5.4 Zoom Scale Dependence

At this point, we have reasonable estimates for the lens distortion parameter $\kappa(0)$ and the camera calibration matrix $\mathbf{K}(0)$ at the lowest zoom scale. The final initial estimation problem is to compute the parameters $\{a_f, b_f, a_\kappa, b_\kappa\}$ from (4) and (5) that define the variation of the intrinsic parameters over the entire zoom scale. For this problem, we use images and feature correspondences from the set $\mathcal{Z}$.

First, we estimate the lens distortion parameters at each scale, $\{\kappa(z_1), \ldots \kappa(z_M)\}$. We consider each image in $\mathcal{Z}$ independently; since the zoom scale z_0 is in this set, the images with zoom scales z_0 and z_i are related by a homography $\mathbf{H}_i$. Since we already have estimated $\kappa(0)$, we have a relationship between correspondences in the *undistorted* image corresponding to z_0 and the *distorted* image corresponding to z_i that resembles (10):

$$\mathbf{x_i^d} \times \mathbf{H}\mathbf{x_0^u} + \kappa(z_i)\mathbf{z_i^d} \times \mathbf{H}\mathbf{x_0^u} = 0 \tag{18}$$

where $\mathbf{x_0^u}(\text{undistorted}) \leftrightarrow \mathbf{x_i^d}(\text{distorted})$. This corresponds to a polynomial eigenvalue problem

$$(\mathbf{M_1} + \kappa(z_i)\mathbf{M_2})\mathbf{h} = 0 \tag{19}$$

where

$$\mathbf{M_1} = \begin{bmatrix} 0 & -y_i & y_0 \\ x_i & 0 & -x_0 \end{bmatrix} \quad \mathbf{M_2} = \begin{bmatrix} 0 & -s_i y_0 & 0 \\ s_i x_0 & 0 & 0 \end{bmatrix} \tag{20}$$

with similar notation to (13). Since the camera undergoes pure zooming, and all the image coordinates are normalized with respect to the principal point, the unknown homography again has only 3 nonzero elements, with $h_{12} = h_{13} = h_{21} = h_{23} = h_{31} = h_{32} = 0$. We can again solve the eigenvalue problem to obtain an estimate of the homography $\mathbf{H}_i$ and the lens distortion coefficient $\kappa(z_i)$.

It is straightforward to show that the camera calibration matrix $\mathbf{K_i}$ for each undistorted image in $\mathcal{Z}$ is related to the camera calibration matrix $\mathbf{K}(\mathbf{0})$ at zoom scale z_0 by

$$\mathbf{K_i} = \mathbf{H_i}\mathbf{K}(\mathbf{0}) \tag{21}$$

After estimating the lens distortion and underlying homography in the previous step, the right-hand side of (21) is entirely determined, and we immediately obtain an estimate of the camera calibration matrix $\mathbf{K_i} = \text{diag}(k_1, k_2, k_3)$, from which we can extract the focal length $f_x(z_i) = \frac{k_1}{k_3}$ and $f_y(z_i) = \frac{k_2}{k_3}$. The aspect ratio α can be estimated as

$$\alpha = \frac{1}{|\mathcal{Z}|}\sum_{i=0}^{|\mathcal{Z}|}\frac{f_y(z_i)}{f_x(z_i)} \tag{22}$$

Since α is fixed, from this point forward, we only discuss one focal length parameter $f(z) = f_x(z)$, assuming α can be used to compute $f_y(z)$.

Finally, we use all the estimated $f(z_i)$ to estimate the parameters $\{a_f, b_f\}$ of the focal length model by solving the linear least-squares problem:

$$\begin{bmatrix} z_1 & z_1^2 \\ \vdots & \vdots \\ z_n & z_n^2 \end{bmatrix}\begin{bmatrix} a_f \\ b_f \end{bmatrix} = \begin{bmatrix} f(z_1) - f(0) \\ \vdots \\ f(z_n) - f(0) \end{bmatrix} \tag{23}$$

We similarly use all the estimated $\kappa(z_i)$ to estimate the parameters $\{a_\kappa, b_\kappa\}$ of the lens distortion model by solving the linear least-squares problem

$$\begin{bmatrix} 1 & -\left(\kappa\left(z_1\right) - \kappa\left(0\right)\right)^{\frac{1}{2}} \\ \vdots & \vdots \\ 1 & -\left(\kappa\left(z_n\right) - \kappa\left(0\right)\right)^{\frac{1}{2}} \end{bmatrix}\begin{bmatrix} a_\kappa^{\frac{1}{2}} \\ b_\kappa \end{bmatrix} = \begin{bmatrix} f(z_1)\left(\kappa\left(z_1\right) - \kappa\left(0\right)\right)^{\frac{1}{2}} \\ \vdots \\ f(z_n)\left(\kappa\left(z_n\right) - \kappa\left(0\right)\right)^{\frac{1}{2}} \end{bmatrix} \tag{24}$$

We note that a relatively small number of images in $\mathcal{Z}$ (typically 5–10) are required to calibrate the complete PTZ model, compared to the number of samples that may be required for the linear interpolation model proposed in [22].

5.5 Refinement with Nonlinear Optimization

The parameters obtained up to this point were all obtained independently using simple linear least-squares problems. These serve as good initializations for a final nonlinear joint parameter estimation over all the images in $\mathcal{Z}$. That is, we exploit our knowledge of the explicit form of the homography matrix $\mathbf{H_{ij}}$ for a pair of undistorted images with the same (pan, tilt) and varying zoom scales z_i and z_j:

$$\mathbf{H_{ij}} = \begin{bmatrix} \frac{f(z_i)}{f(z_j)} & 0 & c_x(1 - \frac{f(z_i)}{f(z_j)}) \\ 0 & \frac{f(z_i)}{f(z_j)} & c_y(1 - \frac{f(z_i)}{f(z_j)}) \\ 0 & 0 & 1 \end{bmatrix} \tag{25}$$

in which $\frac{f(z_i)}{f(z_j)} = 1 + \frac{a_f z_i + b_f z_i^2}{f(z_j)}$.

Using this explicit parametrization, we minimize the reprojection error summed over every pair of images in $\mathcal{Z}$ and every correspondence in the image pair:

$$F(a_f, b_f, f(0), a_\kappa, b_\kappa, \kappa(0)) = \sum_{(i,j)\in\mathcal{Z}} \sum_{\{x_{ij}^k, x_{ij}^{k'}\}} \|\mathbf{H_{ij}} x_{ij}^k - x_{ij}^{k'}\|^2 \tag{26}$$

Here, $\{x_{ij}^k, x_{ij}^{k'}\}$ are undistorted using (3) and (5) with respect to the updated $\{a_\kappa, b_\kappa, \kappa(0)\}$. This sum-of-squares cost function can be minimized using the Levenberg-Marquardt algorithm.

5.6 Calibrating Mechanical Error

In the previous section, no information from the PTZ camera control interface is used in the calibration, since the reported pan and tilt parameters are considered to be inaccurate. Based on the observations in Section 3, we propose a linear compensation model for the mechanical error in the PTZ camera as:

$$p_m = \beta_p p \quad t_m = \beta_t t \tag{27}$$

where (p_m, t_m) are the measured (reported) pan and tilt, and (p, t) are the actual pan and tilt. Via (1), we compute the errors (e_p, e_t) at each position in the set $\mathcal{PT}$, leading to the estimates

$$\beta_p = 1 + \sum_{i=1}^{N} \frac{e_p^i}{p_m^i} \quad , \quad \beta_t = 1 + \sum_{i=1}^{N} \frac{e_t^i}{t_m^i} \tag{28}$$

From this point forward, all the (pan, tilt) parameters are compensated using (27).

6 Dynamic Correction for a PTZ Camera

We assume that the algorithms in Section 5 are carried out at the time the PTZ camera is mounted on-site. However, it's unreasonable to expect PTZ cameras to be frequently recalibrated, especially in a highly trafficked environment with many cameras. As demonstrated in Section 3, the pan and tilt parameters reported by the camera become increasingly unreliable; therefore, we desire a dynamic correction method that ensures the same field of view is captured every time the user inputs an absolute (pan, tilt, zoom) coordinate, even after hundreds of hours of operation. Our approach is to build a feature library of the camera's environment at the time of mounting, and use matches between the on-line images and this library as the basis for online correction. That is, the user inputs a (pan, tilt, zoom) directive to the camera, and the camera compensates "behind the scenes" to make sure the correct field of view is returned.

6.1 Feature Library of the Scene

Figure 8 illustrates the concept of the feature library for a PTZ camera, which we build in undistorted (pan, tilt) coordinates. We set the zoom scale to its minimum value, and control the camera to sweep around the environment at discrete pan and tilt positions sufficient to cover the entire scene. At each position, we acquire an image and extract all the SURF features except at the image borders. We prefer SURF features due to their use of efficient box-filter approximations instead of Gaussian partial derivatives as in SIFT [16], enabling very fast detection and matching [17].

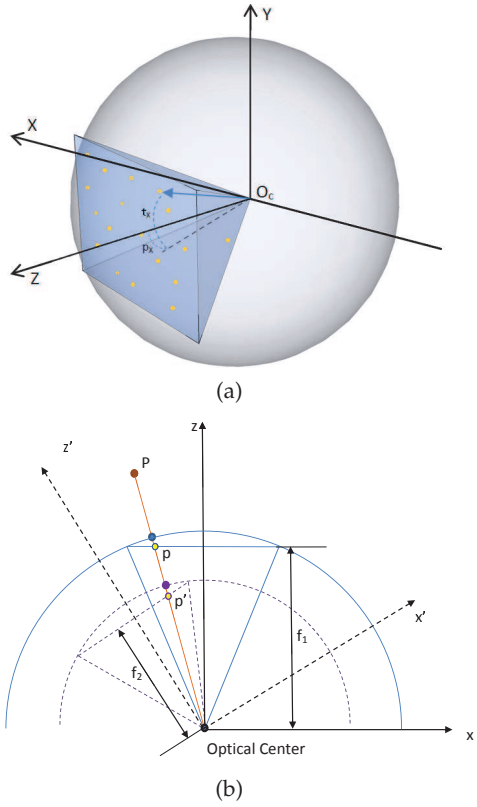


Fig. 8. The feature library for a PTZ camera. (a) The features for the PTZ camera are stored in spherical coordinates. (b) Two projections of the same 3D point from images with different (pan, tilt, zoom) coordinates will have the same (θ, ϕ) values in the feature library.

The library stores each feature i, its SURF descriptor, and its location in undistorted (pan, tilt) coordinates (θ_i, ϕ_i), computed as

$$(\theta_i, \phi_i) = \left(p + \arctan\frac{x_i}{f(0)}, t + \arctan\frac{y_i}{\alpha f(0)}\right)$$

where (x_i, y_i) is the image coordinate of the feature on the undistorted image plane, and (p, t) are the (pan, tilt) coordinates of the camera for this image. We note that all the intrinsic parameters of the camera, including the lens distortion coefficient, are available and reliable at this stage, since it is performed immediately after the calibration process from Section 5. Features from different images that highly overlap in both descriptor and position are merged, so that each feature appears only once in the library.

Figure 8b illustrates how two projections of the same

3D point from images with different (pan, tilt, zoom) coordinates have the same (θ, ϕ) values using this scheme, suggesting its value for providing accurate pose estimation reference. We note that Sinha and Pollefeys [22] mentioned a related possibility of using a calibrated panorama for closed-loop control of a PTZ camera; however, the pose is retrieved by aligning the current image to an image generated by the panorama with the same pose configuration, which would introduce serious errors at the edges due to distortion. Furthermore, the method depends on the assumption that the internal parameters of the camera and the zoom control are stable. Finally, this method can only correct small errors in pose, since with large errors, the two images will fail to match.

Since the scene features may change over long periods of time, we constantly update the feature library with features that are repeatedly detected but unmatched as the camera operates.

6.2 Online Correction

In this section, we assume that the camera is in an unknown (pan, tilt, zoom) true pose (p, t, z) that must be estimated. It reports its pose as (p_m, t_m, z_m), which we assume are incorrect due to the accumulation of errors discussed in Section 3.

The correction process involves the feature library $\mathcal{L}$ created at the time of mounting, and the set of SURF features $\mathcal{S}$ acquired in the current image. We try to match each feature in $\mathcal{S}$ to $\mathcal{L}$, resulting in a set of putative matches $\{(x_i, y_i) \leftrightarrow (\theta_i, \phi_i), i = 1, \ldots, N\}$, where (x_i, y_i) is the normalized (with respect to principal point) feature location in the (distorted) current image, and (θ_i, ϕ_i) is the true (pan, tilt) location of a feature from $\mathcal{L}$. The feature matches are initially computed using nearest-neighbor matching between SURF descriptors.

The problem of estimating the true pose (p, t, z) proceeds in several steps. We begin by noting that

$$\alpha \left(\frac{x_i}{\tan(\theta_i - p)} \right) - \frac{y_i}{\tan(\phi_i - t)} = 0 \tag{29}$$

Considering (29) for any two features i and j, we can compute an estimate for (p, t) by solving the independent pair of quadratic equations:

$$\begin{aligned} &(x_j \tan\theta_j - x_i \tan\theta_i) \tan^2 p \\ &\quad + (x_i - x_j)(\tan\theta_i \tan\theta_j - 1) \tan p \\ &\quad + \tan\theta_j x_i - \tan\theta_i x_j = 0 \\ &(y_j \tan\phi_j - y_i \tan\phi_i) \tan^2 t \\ &\quad + (y_i - y_j)(\tan\phi_i \tan\phi_j - 1) \tan t \\ &\quad + \tan\phi_j y_i - \tan\phi_i y_j = 0 \end{aligned} \tag{30}$$

We use the criteria $\text{sign}(\theta_i - p) = \text{sign}(x_i)$ and $\text{sign}(\phi_i - t) = \text{sign}(y_i)$ to choose the correct solution. Then we can obtain an initial guess for (p, t) by removing the outliers with two-parameter RANSAC, which computes (p, t) for randomly selected feature pairs with (30) and evaluates the fit with (29). This can effectively remove the mismatched feature pairs. In order to make the feature matching robust to large changes in the scene, we use the following criteria to evaluate the result:

$$\text{std}(\mathbf{E}_{in}) < \varepsilon, \ |\mathbb{S}_{in}| > \tau \tag{31}$$

in which $\mathbb{S}_{in}$ is the set of τ inliers found after RANSAC and $\text{std}(\mathbf{E}_{in})$ is the standard deviation of the error in fitting (29) with each inlying feature pair. In practice, we used $(\varepsilon, \tau) = (1.5 \text{ pixels}, 20)$. If these criteria cannot be met, the camera should be directed to another random position to capture another image for correction.

Expanding and rearranging (29) gives

$$w_i = \frac{\alpha x_i}{y_i} = \frac{\tan(\theta_i - p)}{\tan(\phi_i - t)} = \frac{(\tan\theta_i - \tan p)(1 + \tan\phi_i \tan t)}{(1 + \tan\theta_i \tan p)(\tan\phi_i - \tan t)} \tag{32}$$

We note that this relationship holds even though (x_i, y_i) are in distorted image coordinates, since by (3), the ratio w_i is invariant to the unknown value $\kappa(z)$. Thus, for N matches, we minimize the nonlinear cost function

$$F_1(p, t) = \left\| \mathbf{A}^\top \mathbf{v} - \mathbf{b} \right\|^2 \tag{33}$$

in which

$$\begin{aligned} \mathbf{A} &= \begin{bmatrix} w_1 \tan\theta_1 \tan\phi_1 + 1 & \cdots & w_N \tan\theta_N \tan\phi_N + 1 \\ w_1 - \tan\theta_1 \tan\phi_1 & \cdots & w_N - \tan\theta_N \tan\phi_N \\ \tan\phi_1 - w_1 \tan\theta_1 & \cdots & \tan\phi_N - w_N \tan\theta_N \end{bmatrix} \\ \mathbf{b} &= \begin{bmatrix} \tan\theta_1 - w_1 \tan\phi_1 \\ \vdots \\ \tan\theta_N - w_N \tan\phi_N \end{bmatrix}, \ \mathbf{v} = \begin{bmatrix} \tan p \\ \tan t \\ \tan p \tan t \end{bmatrix} \end{aligned} \tag{34}$$

The minimization can be accomplished using the LM algorithm with the initial guess from the output of RANSAC. Once the pose (p, t) is determined, we can ideally compute the lens distortion coefficient $\kappa(z)$ using any two features i and j in either of the two forms

$$\begin{aligned} \kappa(z) &= \frac{\tan(\theta_i - p)x_j - \tan(\theta_j - p)x_i}{\tan(\theta_j - p)x_i r_j - \tan(\theta_i - p)x_j r_i} \\ &= \frac{\tan(\phi_i - t)y_j - \tan(\phi_j - t)y_i}{\tan(\phi_j - t)y_i r_j - \tan(\phi_i - t)y_j r_i} \end{aligned} \tag{35}$$

where $r_i = x_i^2 + y_i^2$. An estimate of $\kappa(z)$ can be computed as the average of all $2N(N-1)$ pairwise estimates from (35).

Similarly, we can ideally compute the focal length $f(z)$ from any feature i using

$$f(z) = \frac{(\alpha x_i \tan(\phi_i - t) + y_i \tan(\theta_i - p))}{2\alpha(1 + \kappa(z) r_i) \tan(\theta_i - p) \tan(\phi_i - t)} \tag{36}$$

and an estimate for $f(z)$ can be computed as the average of all N such estimates.

In most situations, we can assume the internal model is stable and we only need to correct the offsets of pan, tilt and zoom, which can be done with a *single-shot process* as follows. We first perform a joint nonlinear minimization to refine the estimates of $(p, t, f(z))$ by

minimizing

$$F_2(p,t,f(z)) = \sum_{i=1}^{N} \left\| \begin{array}{c} f(z)\tan(\theta_i - p) - \frac{x_i}{(1+\kappa(z)r_i)} \\ \alpha f(z)\tan(\phi_i - t) - \frac{y_i}{(1+\kappa(z)r_i)} \end{array} \right\| \quad (37)$$

in which $\kappa(z)$ is computed by (5). Then we can retrieve the true z by

$$z = (2b_f)^{-1}\left[-a_f + \left(a_f^2 - 4b_f\left[f(0) - f(z)\right]\right)^{-1}\right] \quad (38)$$

The offsets for correcting a measured pose setting can be stored as $\Delta p = p - p_m$, $\Delta t = t - t_m$ and $\Delta z = z - z_m$. We can now direct the camera to user-requested pose settings (p_r, t_r, z_r) by sending the pose instruction $(p_r + \Delta p, t_r + \Delta t, z_r + \Delta z)$ to the camera.

In rare cases, for low-quality cameras and lenses, the internal parameters of the camera should be occasionally re-calibrated completely, using *multi-shot correction*. That is, several images are acquired in order to re-estimate the relationship between the focal length/lens distortion and zoom scale. In our experiments, this happens infrequently, only after hundreds of hours of continuous operation and repetitive power switching.

7 Experiments

We conducted experiments on both simulated data and real data to evaluate the proposed models and algorithms.

7.1 Simulated Data

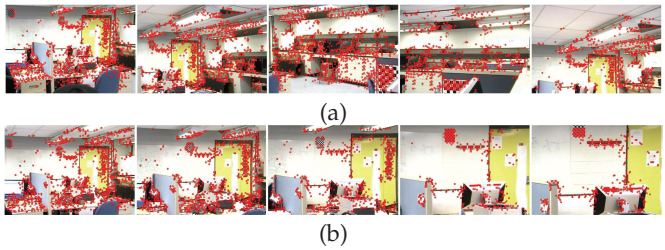
In this section, we discuss experimental results obtained using simulated data. We first generated parameters for a PTZ camera using the model in Section 4 using the observations in Section 3. The resulting parameters are: $f_x(z) = 500 + 0.1z + 0.3\times10^{-5}z^2$, $\alpha = 0.95$, $\kappa(z) = -0.15 + \frac{1\times10^4}{(f_x(z)+200)^2}$, $(c_x, c_y) = (320, 240)$. Next, we generated 5 images for the set $\mathcal{PT}$ and 5 images for the set $\mathcal{Z}$, used to estimate the parameters of the model. We generated 1000 points in 3D and projected these points into each of the $\mathcal{PT}$ and $\mathcal{Z}$ images to generate correspondences between the (distorted) image planes. Since real-world matching isn't ideal, we also added zero-mean, variance σ^2 isotropic Gaussian random noise to each projected image location, where σ ranged from 0 to 3.

This noisy, simulated data was input to the series of steps outlined in Section 5 to estimate the parameters of the model; the calibration errors for the intrinsic parameters are shown in Figure 9. The reported error for each estimated parameter is computed as $\left|\frac{\text{param}_{\text{est}} - \text{param}_{\text{actual}}}{\text{param}_{\text{actual}}}\right|$ except for the error in κ, which is computed as $|\text{param}_{\text{est}} - \text{param}_{\text{actual}}|$.

From the results it can be seen that the calibration algorithm is effective and able to maintain reasonable performance with the presence of noise. The algorithm seems to perform slightly better at higher zoom scales. We observed that even in the worst case of 3-pixel noise, the estimated focal length errors are between 6% and 8%.

7.2 Real Data

We also tested the proposed model and calibration method on an Axis AX213 PTZ camera with a resolution of 704×480 and $35\times$ optical zoom. Sample images used in the calibration are illustrated in Figure 10. We used 5 images in each of the sets $\mathcal{PT}$ and $\mathcal{Z}$.



(a) (b)

Fig. 10. (a) Example images from the set $\mathcal{PT}$, with a subset of SURF matches. (b) Example images from the set $\mathcal{Z}$, with a subset of SURF matches.

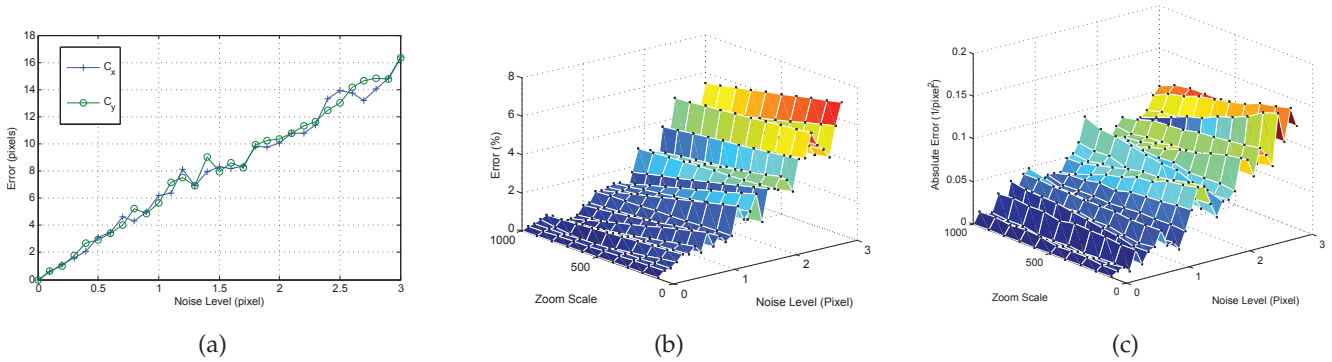
In order to evaluate the performance of the proposed method, we calibrated the camera at each fixed (pan, tilt, zoom) setting with Zhang's method [27], which we consider as ground truth. We also compare our calibration method to that of Sinha and Pollefeys [22], using 100 images from different zoom scales and poses. We call this alternate method "discrete zoom calibration" (DZC). Note that we use the division model of lens distortion for all the methods.

Figure 11 compares the results of the proposed method to discrete zoom calibration, as a function of zoom scale. The reported error for each estimated parameter is computed as $\left|\frac{\text{param}_{\text{est}} - \text{param}_{\text{zhang}}}{\text{param}_{\text{zhang}}}\right|$ except for the error in κ, which is computed as $|\text{param}_{\text{est}} - \text{param}_{\text{zhang}}|$. We can see that the proposed method outperforms DZC for all the parameters, especially at higher zoom scales. Figure 11d shows the average error in $f(z_i)$ as the number of images used in both algorithms increases. We observe that DZC requires at least 80 images in order to achieve reasonable results, which the proposed method can achieve using only 10 images, and that the proposed method is relatively insensitive to the number of images used.

Since a minimum of 5 images is enough for the calibration, all the nonlinear optimization routines converge rapidly, and the calibration process is very efficient. The whole process (including image acquisition) can be done within 30 seconds by our C# program on an Intel Core2 Duo 2.66GHz desktop with 3GB Memory.

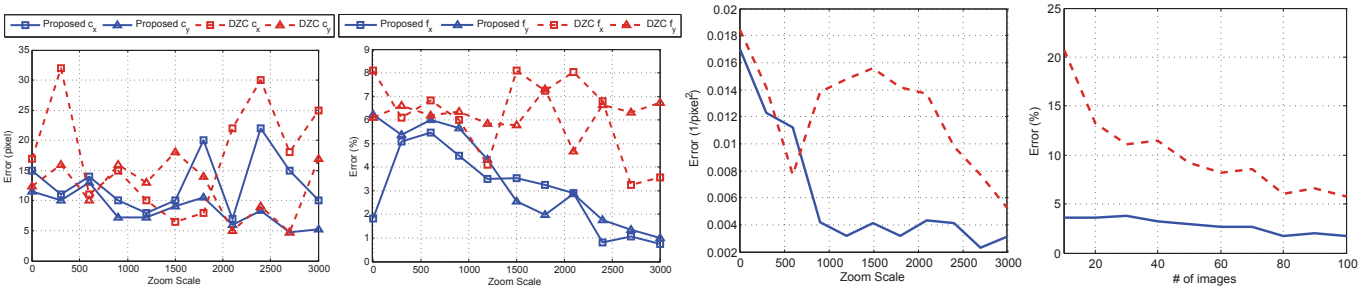
7.3 Dynamic Correction

After the initially calibrating the real PTZ camera as described above, we mounted it in a lab setting. As described in Section 6.1, we built the feature library at the lowest zoom scale, as shown in Figure 12a. Figure 12b illustrates a sample image acquired after 100 hours of continuous PTZ operation and several power cycles,



(a) (b) (c)

Fig. 9. Calibration results for the simulation experiment. Each error surface is a function of the noise standard deviation σ and the zoom scale z. (a) The error in the principal point (c_x, c_y). (b) The error in the focal length $f(z)$. (c) The error in the lens distortion coefficient $\kappa(z)$.



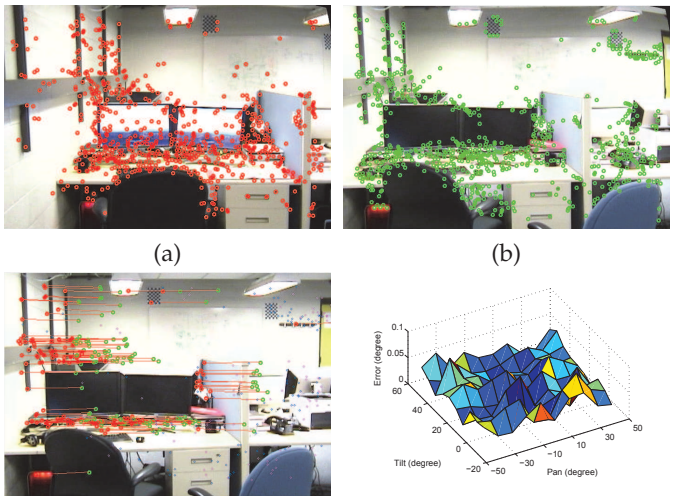
(a) (b) (c) (d)

Fig. 11. Calibration results for the real experiment. The error in each parameter is plotted as a function of the zoom scale z. (a) The error in the principal point (c_x, c_y). (b) The error in the focal length $f(z)$. (c) The error in the lens distortion coefficient $\kappa(z)$. The solid lines show the error for the proposed method using 10 input images, and the dotted lines show the error for DZC using 100 images. (d) The average error in $f(z)$ as the number of images used changes.

and the corresponding SURF features. Note that changes have occurred in the scene. Figure 12c illustrates the subset of online features matched to the library. Here, the library features are rendered at image coordinates corresponding to the reported (pan, tilt) of the camera, which we can see is quite erroneous. That is, after 100 hours of operation, the (pan, tilt) reported by the camera is quite unreliable.

In order to quantify the performance of dynamic correction, we conducted an experiment immediately after building the feature library. The camera is directed to several different positions in the pan range $[-45, 45]$ and tilt range $[-10, 50]$. For each position, the pose parameters computed using the feature library are compared to the parameters read from the camera, which are considered to be ground truth. Figure 12d shows the sum of squared error in pan and tilt for each position. We observed that the error is less than 0.10 degree for all positions with a mean of 0.03 degrees.

Next, we repeated the experiment from Figure 1, now incorporating dynamic correction; the results are illustrated in Figure 13. We can see that the targets of interest are now successfully located, suggesting that the proposed method can significantly improve the performance of PTZ cameras in video surveillance.



(a) (b) (c) (d)

Fig. 12. (a) Example features from the feature library at the lowest zoom scale. (b) A sample image acquired after 100 hours of continuous PTZ operation and several power cycles, and the corresponding SURF features. (c) The subset of features in (b) matched to the library (after outlier rejection). The lines indicate the distance to the feature library rendered at the (pan, tilt) reported by the camera, which we can see is quite erroneous. (d) The correction error for different poses.

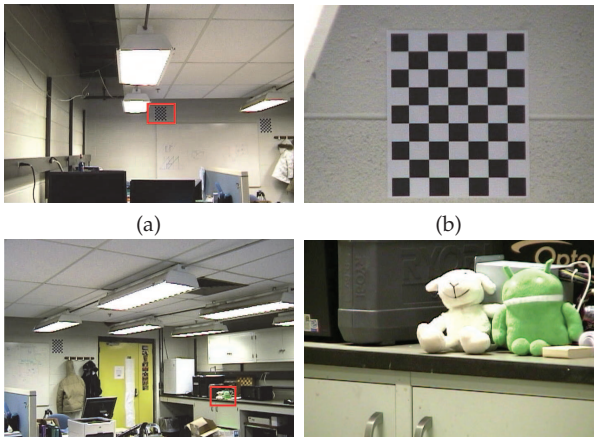


Fig. 13. The 36-hour experiment from Figure 1, repeated with our proposed dynamic correction algorithm. (a,c) The red rectangles indicate the image after dynamic correction. (b,d) Close-ups of the images after dynamic correction. Compare these images to Figure 1(b)-(c).

We also repeated the longer 200-hour experiment from Figure 5, now incorporating dynamic correction; the results are illustrated in Figure 14. We can see that the error is significantly reduced for all monitored positions. In addition, there is no increasing trend in the error over long periods of random motion. Even after restarting the camera, all four monitored positions are well-recovered using dynamic correction (bottom row of Figure 14).

The dynamic correction algorithm is efficient in practice. With the same hardware configuration as for the initial calibration, a single-shot correction process (including image acquisition) can be done within 3 seconds. With a GPU implementation (NVIDIA GeForce GTX680 with 4G memory), the time for single-shot correction can be reduced to 60 ms, which makes dynamic correction feasible for every movement of the PTZ camera.

We also conducted a similar experiment in an outdoor environment. We chose five (pan, tilt, zoom) locations to monitor in the outdoor environment, shown in Figure 15. The PTZ camera randomly rotated for 120 hours. Every half an hour, the camera captured images from all five monitored locations using both no correction and the proposed dynamic correction algorithm. We computed the average distance between matched SURF features in the reference and online images as a function of time, shown in the bottom row of Figure 15. We arrive at the same conclusion as in the indoor experiment: the dynamic correction effectively removes accumulated error and reduces the average error in (pan, tilt) estimation. We can see that the errors using dynamic correction are somewhat higher than in the indoor experiment. This is due to the much larger field of view and increased disturbances (e.g., changes in luminance and background), as well as the high zoom scales.

However, we note that the (pan, tilt, zoom) parameters computed after dynamic correction are sufficiently accurate in an absolute sense that we can use them to build panoramas without further registration. That is, we create a planar panorama by pointing the camera to a (pan, tilt) position (which is dynamically corrected by our algorithm) and simply rendering the pixels of the obtained image onto the panorama canvas at the ideal locations. Two example panoramas are illustrated in Figure 16. We can see that edges in the images line up precisely. This suggests that the proposed dynamic correction algorithm is immediately useful for algorithms like change detection, in which online and reference images must be accurately aligned.

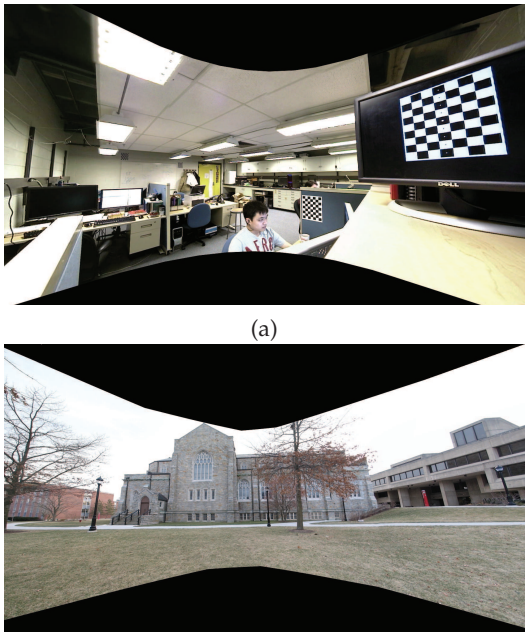
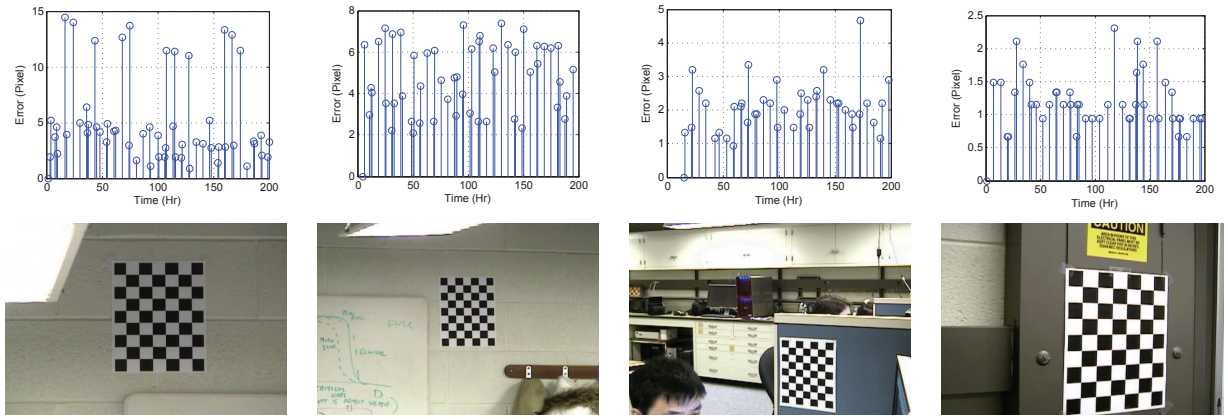


Fig. 16. Two panoramas obtained by the PTZ camera with dynamic correction. (a) Lab from images at 50 positions. (b) VCC from images at 20 positions.

8 Conclusion

We proposed a complete PTZ camera model, and presented an automatic calibration method based on this model. Using only matched image features extracted from a few images of the scene at different poses and zoom scales, the complete model for the PTZ camera can be recovered. Furthermore, we presented a fast dynamic correction method for keeping a PTZ camera calibrated, using a feature library built at the time the PTZ camera is mounted. Experiments using both simulated and real data show that the calibration methods are fast, accurate, and effective. The proposed PTZ camera model enables dynamic correction using only one image, which is not possible using previous methods.

The accurate estimation of the principal point is crucial to our calibration method, especially the assumption that it coincides with the distortion center and zooming center. However, this may not be the case for some cameras [14]. Also, while we assumed that the PTZ camera



(a) Pos.A (1,0,8000) (b) Pos.B (16,-2,7000) (c) Pos.C (60,-10,1000) (d) Pos.D (-70,-7,0)

Fig. 14. The 200-hour experiment from Figure 5, repeated with our proposed dynamic correction algorithm. The first row shows the error in pixels as a function of time over the 200-hour experiment. The error is much lower than without dynamic correction. The second row shows the monitored positions after power-cycling the camera, indicating that the serious errors without dynamic correction have been fixed. Compare these images to Figure 5.

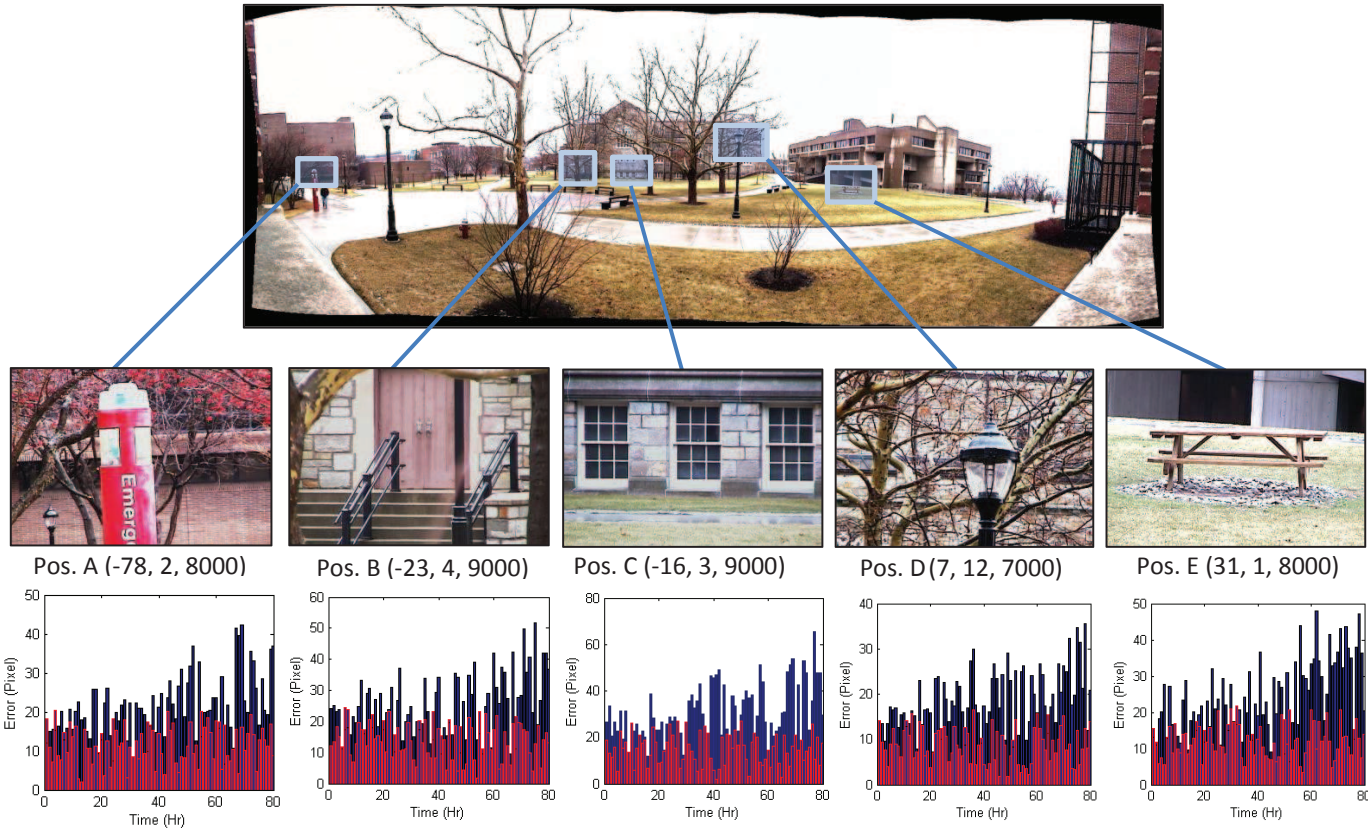


Fig. 15. A repeatability experiment in an outdoor environment. The feature library is built at the minimum zoom scale (1000), and all the test positions are at high zoom scales (7000–9000). The bottom row gives the average distance between matched SURF features in the reference and online images as a function of time; red bars are errors using dynamic correction while blue bars are errors without correction.

was purely rotating, in practice the effects of possible translation of the camera center cannot be ignored, as observed by several researchers [9], [10]. Finally, we observed that the auto-focusing of the camera has a non-negligible influence on its internal parameters. We plan to investigate all these issues, in order to make the PTZ camera model more accurate and comprehensive.

We observed that the algorithm may fail when a large component in the scene was moved or when the background is moving slowly. Since the scene features may change frequently for some applications, we plan to investigate approaches for keeping the feature library up to date in dynamic scenes. Finally, we plan to incorporate the proposed model and algorithms into real video surveillance applications, to improve both 2D and 3D tracking and localization performance.

REFERENCES

[1] N. Ashraf and H. Foroosh. Robust auto-calibration of a PTZ camera with non-overlapping FOV. In *International Conference on Pattern Recognition*, Dec. 2008.

[2] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3):346–359, 2008.

[3] R. T. Collins and Y. Tsin. Calibration of an outdoor active camera system. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1999.

[4] J. Davis and X. Chen. Calibrating Pan-Tilt cameras in wide-area surveillance networks. In *IEEE International Conference on Computer Vision*, 2003.

[5] L. de Agapito, R. Hartley, and E. Hayman. Linear Calibration of a Rotating and Zooming Camera. In *International Conference on Computer Vision and Pattern Recognition*, 1999.

[6] L. de Agapito, E. Hayman, and I. Reid. Self-Calibration of Rotating and Zooming Cameras. *International Journal of Computer Vision*, 45(2):107–127, Nov. 2001.

[7] A. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[8] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[9] R. I. Hartley. Self-Calibration of Stationary Cameras. *International Journal of Computer Vision*, 22(1):5–23, Feb. 1997.

[10] E. Hayman and D. Murray. The effects of translational misalignment when self-calibrating rotating and zooming cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(8):1015–1020, Aug. 2003.

[11] B. He and Y. Li. Camera calibration with lens distortion and from vanishing points. *Optical Engineering*, 48(1):013603, Jan. 2009.

[12] Z. Kukelova and T. Pajdla. A Minimal Solution to Radial Distortion Autocalibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2410–2422, Apr. 2011.

[13] R. Lenz and R. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology. In *IEEE International Conference on Robotics and Automation*, 1987.

[14] M. Li and J.-M. Lavest. Some aspects of zoom lens camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(11):1105–1110, 1996.

[15] S.-N. Lim, A. Elgammal, and L. Davis. Image-based pan-tilt camera control in a multi-camera surveillance environment. In *International Conference on Multimedia and Expo*, 2003.

[16] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov. 2004.

[17] J. Luo and O. Gwun. A comparison of SIFT, PCA-SIFT and SURF. *International Journal of Image Processing*, 3(4):143, Oct. 2009.

[18] R. Nevatia. Camera calibration from video of a walking human. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1513–1518, Sept. 2006.

[19] E. Rosten and R. Loveland. Camera distortion self-calibration using the plumb-line constraint and minimal Hough entropy. *Machine Vision and Applications*, 22(1):77–85, Apr. 2009.

[20] M. Sarkis, C. Senft, and K. Diepold. Calibrating an Automatic Zoom Camera With Moving Least Squares. *IEEE Transactions on Automation Science and Engineering*, 6(3):492–503, July 2009.

[21] T. Schoepflin and D. Dailey. Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation. *IEEE Trans. Intelligent Transportation Systems*, 4(2):90–98, June 2003.

[22] S. N. Sinha and M. Pollefeys. Pan-tilt-zoom camera calibration and high-resolution mosaic generation. *Computer Vision and Image Understanding*, 103(3):170–183, Sept. 2006.

[23] 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, Cybernetics*, 36(5):1091–103, Oct. 2006.

[24] R. Steele, C. Jaynes, A. Leonardis, H. Bischof, and A. Pinz. Overconstrained linear estimation of radial distortion and multi-view geometry. In *European Conference on Computer Vision*, 2006.

[25] B. Tordoff and D. Murray. The impact of radial distortion on the self-calibration of rotating cameras. *Computer Vision and Image Understanding*, 96(1):17–34, Oct. 2004.

[26] Z. Wu and R. J. Radke. Using scene features to improve wide-area video surveillance. In *IEEE Workshop on Camera Networks and Wide Area Scene Analysis*, 2012.

[27] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, 2000.

Ziyan Wu Ziyan Wu is currently working towards a Ph.D. degree in Computer and Systems Engineering in the Department of Electrical, Computer, and Systems Engineering at Rensselaer Polytechnic Institute. He received a B.S. degree in Electrical Engineering and Automation and an M.S. degree in Measurement Technology and Instruments, both from Beihang University in Beijing, China in 2006 and 2009 respectively. He received a Honeywell Innovators Award in 2008 and worked as a system engineer in Honeywell Technology Solution Labs in Shanghai, China in 2009. He is a graduate student affiliated with the DHS Center of Excellence on Explosives Detection, Mitigation and Response (ALERT). His research interests include camera calibration, multi-object tracking, anomaly detection and human re-identification with camera networks.

Richard J. Radke Richard J. Radke joined the Electrical, Computer, and Systems Engineering department at Rensselaer Polytechnic Institute in 2001, where he is now an Associate Professor. He has B.A. and M.A. degrees in computational and applied mathematics from Rice University, and M.A. and Ph.D. degrees in electrical engineering from Princeton University. His current research interests include computer vision problems related to modeling 3D environments with visual and range imagery, designing and analyzing large camera networks, and machine learning problems for radiotherapy applications. Dr. Radke is affiliated with the NSF Engineering Research Centers for Subsurface Sensing and Imaging Systems (CenSSIS) and Smart Lighting, the DHS Center of Excellence on Explosives Detection, Mitigation and Response (ALERT), and Rensselaers Experimental Media and Performing Arts Center (EMPAC). He received an NSF CAREER award in March 2003 and was a member of the 2007 DARPA Computer Science Study Group. Dr. Radke is a Senior Member of the IEEE and an Associate Editor of *IEEE Transactions on Image Processing*. His textbook *Computer Vision for Visual Effects* was published by Cambridge University Press in 2012.