Using Stationary-Dynamic Camera Assemblies for Wide-area Video Surveillance and Selective Attention

Ankur Jain, Dan Kopell, Kyle Kakligian, and Yuan-Fang Wang
Department of Computer Science, University of California, Santa Barbara, CA 93106
{ankurj,dkopell, smallart, yfwang}@cs.ucsb.edu

Abstract

In this paper, we present a prototype video surveillance system that uses stationary-dynamic (or master-slave) camera assemblies to achieve wide-area surveillance and selective focus-of-attention. We address two critical issues in deploying such camera assemblies in real-world applications: off-line camera calibration and on-line selective focus-of-attention. Our contributions over existing techniques are twofold: (1) in terms of camera calibration, our technique calibrates all degrees-of-freedom (DOFs) of both stationary and dynamic cameras, using a closed-form solution that is both efficient and accurate, and (2) in terms of selective focus-of-attention, our technique correctly handles dynamic changes in the scene and varying object depths. This is a significant improvement over existing techniques that use an expensive and non-adaptable table-look-up process.

1. Introduction

To achieve effective surveillance, a large number of cameras are often used for wide-area coverage. Once suspicious persons/activities have been identified through video analysis, selected cameras ought to be able to obtain close-up views of these suspicious subjects for further scrutiny and identification (e.g., to obtain a close-up view of the license plate of a car or the face of a person). These two requirements, a large field-of-view and selective focus-of-attention, place conflicting constraints on the system configurations and camera parameters.

The proposed solution is to cover an extended surveillance area by multiple stationary (or master) cameras with wide fields-of-view. The stationary cameras perform a global, wide field-of-view analysis of the motion patterns in the surveillance zone. Based on some pre-specified criteria, the stationary cameras identify suspicious behaviors or subjects that need further attention. The stationary cameras will guide the dynamic cameras to focus on the region of interest (e.g., the license plate of a car or the face of a person) for selective attention and analysis.

This paper addresses two specific problems that present unique challenges to using stationary-dynamic cameras for video surveillance: (1) off-line calibration of both stationary and dynamic cameras, and (2) on-line selective focus-of-attention by cooperative stationary-dynamic sensing.

2. Background and Contribution

We contrast our approaches with the state-of-the-art in off-line calibration and on-line selective focus-of-attention—issues that are critical to the use of stationary-dynamic camera assemblies for video surveillance.

Davis and Chen [3], presented a technique for calibrating a pan-tilt camera off-line. This technique adopted a general camera model that did not assume that the rotational axes were orthogonal or that they were aligned with the camera's imaging optics. Furthermore, they argued that the traditional methods of calibrating stationary cameras using a fixed calibration stand were impractical for calibrating dynamic cameras, because a dynamic camera had a much larger working volume. Instead, a novel technique was adopted to generate virtual calibration landmarks using a moving LED. The 3D positions of the LED were inferred, via stereo triangulation, from multiple stationary cameras placed in the environment. To solve for the camera parameters, an iterative minimization technique was proposed.

Zhou *et al.* [5], presented a technique to achieve selective focus-of-attention on-line using a stationary-dynamic camera pair. The procedure involved identifying, off-line, pixel locations in the stationary camera where a surveillance subject could later appear. The dynamic camera was manually moved to center on the subject. The pan and tilt angles of the dynamic camera were recorded in a look-up table indexed by the pixel coordinates in the stationary camera. The pan and tilt angles needed for maneuvering the dynamic camera to focus on objects that appeared at intermediate pixels in the stationary camera were obtained by interpolation. At run time, the centering maneuver of the dynamic camera was accomplished by a simple table-look-up process, based on the locations of the subject in the stationary camera and the pre-recorded pan-and-tilt maneuvers.

Compared to the state-of-the-art methods surveyed above, our contributions are twofold: In terms of off-line camera calibration:

1. It is well known that three pieces of information are needed to uniquely define a rotation (e.g., pan and tilt): position of the rotation axis, orientation of the axis, and rotation angle. Although [3] assumes this general model, it explicitly calibrates only the position and orientation of the axis. Our technique calibrates all these DOFs.

2. As will be shown in our experimental results (Section 4), the technique of [3] uses an iterative minimization procedure that is computationally expensive and does not guarantee convergence. Our technique solves for all intrinsic and extrinsic camera parameters for both stationary and dynamic cameras using a *closed-form* solution that is both efficient and accurate.

3. While the virtual landmark approach is interesting, we will show in Sec. IV that such a technique is less accurate than the traditional techniques using a small calibration pattern (e.g., a checkerboard). We will argue that traditional techniques can also provide large angular ranges for calibrating pan and tilt DOFs effectively.

In terms of on-line selective focus-of-attention:

1. For the procedure proposed in [5] to work, surveillance subjects must appear at the same depth each time they appear at a particular pixel location in the stationary camera. This assumption is unrealistic in real-world applications. Our technique allows surveillance subjects to appear freely in the environment with varying depths.

2. Manually building a table of pan and tilt angles is a time-consuming process. Furthermore, the process needs to be repeated at each surveillance location, and it will fail if the environmental layout changes later. Our technique adapts automatically to different locales.

3. Our techniques are applicable even with high and varying camera zoom settings and poorly aligned pan and tilt axes.

3. Technical Rationales

In this section, we present the technical details our mathematical formulations pertaining to (1) off-line PTZ camera calibration and, (2) on-line selective focus of attention.

3.1. Off-line calibration

Stationary Cameras: Because the setting of a master camera is held stationary, its calibration is performed only once, off-line. Many calibration algorithms are available and several public-domain packages and free software, such as OpenCV [1] , have routines for calibration. We will not discuss them here.

Dynamic Cameras: Calibrating a pan-tilt-zoom (PTZ) camera is more difficult, as there are many DOFs, and the choice of a certain DOF, e.g., zoom, affects the others.

The pan and tilt DOFs correspond to rotations, specified by the location of the rotation axis, the axis direction, and the angle of rotation. Some simplifications can make the calibration problems slightly easier, but at the expense of a less accurate solution. The simplifications are (1) collocation of the optical center on the axes of pan and tilt, (2) parallelism of the pan and tilt axes with the height (y) and width (x) dimensions of the CCD, and (3) the requested and realized angles of rotation match, or the angle of rotation does not require calibration. For example, [3] assumes that (3) is true and calibrates only the location and orientation of the axes relative to the optical center. In contrast, we adopt a general formulation that does not make any of the above simplifications. We show that simplifications are unnecessary, and that assuming a general configuration does not unduly increase the solution complexity.

The equation relating a 3D world coordinate and a 2D camera coordinate for a *pan-tilt* PTZ camera is [3]:

$$\begin{aligned}\mathbf{P}_r &= \mathbf{M}_{r\leftarrow i}(f)\mathbf{M}_{i\leftarrow c} \\ &\quad \mathbf{T}_\mathbf{t}^{-1}(f)\mathbf{R}_{\mathbf{n_t}}(\phi)\mathbf{T}_\mathbf{t}(f)\mathbf{T}_\mathbf{p}^{-1}(f)\mathbf{R}_{\mathbf{n_p}}(\theta)\mathbf{T}_\mathbf{p}(f)\mathbf{M}_{c\leftarrow w}\mathbf{P}_w \\ &\quad = \mathbf{M}_{r\leftarrow w}(f,\theta,\phi)\mathbf{P_w} \end{aligned} \quad (1)$$

which projects a 3D point ($\mathbf{P_w}$) into a 2D point ($\mathbf{P_r}$), in the camera's CCD array. For stationary cameras, the projection can be decomposed into three parts: a world-to-camera transform ($\mathbf{M}_{c\leftarrow w}$), an ideal projection ($M_{i\leftarrow c}$), and an ideal image-to-real CCD transform ($M_{r\leftarrow i}$). For dynamic cameras, two more DOFs must be considered: pan and tilt. In Eq. 1, θ denotes the pan angle and ϕ the tilt angle. $\mathbf{n_p}$ and $\mathbf{n_t}$ denote the orientations of the pan and tilt axes, respectively. To execute the pan and tilt DOFs, a translation ($\mathbf{T_p}$ and $\mathbf{T_t}$) from the optical center to the respective center of rotation is executed first, followed by a rotation around the respective axis, and then followed by a translation back to the optical center for the ensuing projection[1]. $\mathbf{T_p}$ and $\mathbf{T_t}$ are expressed as functions of the camera zoom (f), because zoom moves the optical center and alters the distances between the optical center and the rotation axes.

To calibrate a PTZ camera, two steps are needed: (1) calibrating the location and orientation of the rotation axes, and (2) calibrating the rotation angles so that the *realized* angles of rotation ($\hat{\theta}$ and $\hat{\phi}$) are as close to the *requested* ones (θ and ϕ). The procedure comprises two nested loops.

- In the inner loop, we execute a wide range of pan (or tilt) movements with a fixed camera zoom. We determine how faithfully these requested pan (or tilt) angles are realized by the camera unit. We construct functions $\hat{\theta} = g(\theta)$ and $\hat{\phi} = h(\phi)$ through interpolation. We also calibrate the rotation axis's location and orientation.

[1] Mathematically speaking, only the components of $\mathbf{T_p}$ and $\mathbf{T_t}$ that are perpendicular to $\mathbf{n_p}$ and $\mathbf{n_t}$ can be determined. The components parallel to $\mathbf{n_p}$ and $\mathbf{n_t}$ are not affected by the rotation, and hence will cancel out in the back- and-forth translations.

• In the outer loop, we vary the zoom setting of the camera and determine, for each selected zoom setting, the movement of the optical center and hence, the relative positions ($\mathbf{T_p}$ and $\mathbf{T_t}$) between the optical center and the rotation axes as functions of zoom. Again, we construct functions $\mathbf{T_p}(f)$ and $\mathbf{T_t}(f)$ through interpolation.

The loop body comprises the following basic steps (illustrated here for calibrating the pan angles):

1. First, holding $\theta = \phi = 0$ (or some selected angles), calibrate the dynamic camera using any stationary camera calibration procedure. Denote the world-to-camera matrix thus obtained as $\mathbf{M}_{c\leftarrow w}(0, f)$.

2. Moving θ to some known angle (but keeping ϕ fixed), calibrate the dynamic camera again using the previous procedure. Denote the world-to-camera matrix thus obtained as $\mathbf{M}_{c\leftarrow w}(\theta, f)$. Then it is easily shown that:

$$\mathbf{M}_{c\leftarrow w}(\hat{\theta}, f) = \mathbf{T_p}^{-1}(f)\mathbf{R}_{\mathbf{n_p}}(\hat{\theta})\mathbf{T_p}(f)\mathbf{M}_{c\leftarrow w}(0, f) \qquad (2)$$

Inverting the world-to-camera matrix on the right hand side of Eq. 2, we have [4]:

$$\begin{aligned}
&\mathbf{T_p}^{-1}(f)\mathbf{R}_{\mathbf{n_p}}(\hat{\theta})\mathbf{T_p}(f) = \mathbf{M}_{c\leftarrow w}(\hat{\theta}, f)\mathbf{M}_{w\leftarrow c}(0, f),\\
&= \begin{bmatrix} 1 & 0 & 0 & T_x \\ 0 & 1 & 0 & T_y \\ 0 & 0 & 1 & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{r}_1^T & 0 \\ \mathbf{r}_2^T & 0 \\ \mathbf{r}_3^T & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -T_x \\ 0 & 1 & 0 & -T_y \\ 0 & 0 & 1 & -T_z \\ 0 & 0 & 0 & 1 \end{bmatrix},\\
&= \begin{bmatrix} \mathbf{r}_1^T & -\mathbf{T_p}\cdot\mathbf{r}_1 + T_x \\ \mathbf{r}_2^T & -\mathbf{T_p}\cdot\mathbf{r}_2 + T_y \\ \mathbf{r}_3^T & -\mathbf{T_p}\cdot\mathbf{r}_3 + T_z \\ 0 & 1 \end{bmatrix}.
\end{aligned}$$

Using $\mathbf{M}_{c\leftarrow w}(\hat{\theta}, f)\mathbf{M}_{w\leftarrow c}(0, f) = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$,

we have:

$$n_{p_x} = \frac{m_{32} - m_{23}}{4w\sqrt{1-w^2}},\ n_{p_y} = \frac{m_{13} - m_{31}}{4w\sqrt{1-w^2}},\ n_{p_z} = \frac{m_{21} - m_{12}}{4w\sqrt{1-w^2}},$$

where, $\hat{\theta} = 2arccos(w)$ and $w = \sqrt{\frac{\sum_{i=1}^{4} m_{ii}}{4}}$. The translation matrix $\mathbf{T_p}$ can be solved using a system of three linear equations $-\mathbf{T_p}\cdot\mathbf{r}_1 + T_x = m_{14}, -\mathbf{T_p}\cdot\mathbf{r}_2 + T_y = m_{24}$ and $-\mathbf{T_p}\cdot\mathbf{r}_3 + T_z = m_{34}$.

In general, this calibration procedure should be carried out multiple times with different θ settings. The axis of rotation and the center of location should be obtained by averaging of multiple calibration trials. The relationship of the requested angle of rotation and the executed angle of rotation, i.e., $\hat{\theta} = g(\theta)$ can be interpolated from multiple trials using a suitable interpolation function f (e.g., a linear, quadratic, or sigmoid function).

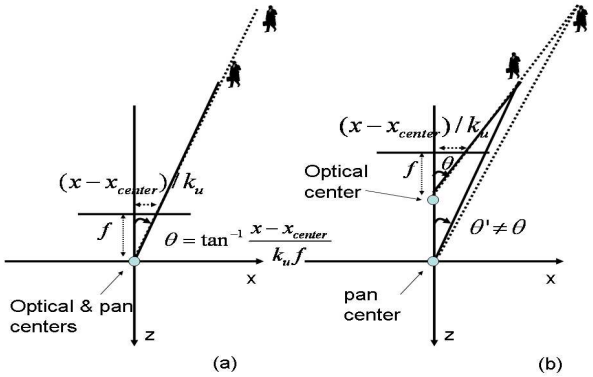


Figure 1. Computing the correct pan DOF (a) if optical and pan centers are collocated, and (b) if they are not.

3.2. On-line selective focus-of-attention

Once a potential suspect has been identified in a stationary camera, the next step is often to relay discriminative visual traits of the suspect (RGB and texture statistics, position and trajectory, etc.) from the stationary camera to a dynamic camera. The dynamic camera then uses its pan, tilt, and zoom capabilities for a closer scrutiny.

To accomplish the selective focus-of-attention feat, we must be able to (1) identify the suspect in the field-of-view of the dynamic camera, and (2) manipulate the camera's pan, tilt, and zoom mechanisms to continuously center upon and present a suitably sized image of the subject.

The first requirement is often treated as a correspondence problem, solved by matching regions in the stationary and dynamic cameras based on similarity of color and texture traits, congruency of motion trajectory, and affirmation of geometrical epipolar constraints. Inasmuch as solutions to the problem are well known in the CV community, we will not address that problem in this paper.

As to the second requirement, there is a trivial solution if the optical center of the PTZ camera is located on the axes of pan and tilt, and if the axes are aligned with the width and height of the CCD. Figure 1(a) illustrates this trivial solution for the pan DOF. Here, we show a cross section of the 3D space that is perpendicular to the pan axis. Assume that the pan axis is the y (vertical) axis. Then the cross section corresponds to the $z - x$ plane in the camera's frame of reference, with the optical center located at the origin. The x-coordinate of the tracked object can then be used for calculating the pan angle as $\theta = arctan((x - x_c)/(k_x f))$, where x_c is the x-coordinate of the center of the image plane, and k_x is the scale factor to convert real-world unit into pixel. As can be seen from Figure 1(a) the collocation of the optical center and the pan axis ensures that the camera pan will not move the optical center. In this case, we achieve the desired centering effect without needing to know the depth of the tracked object (a homography).

In reality, however, the optical center is not located on the rotation axis. As illustrated in Figure 1(b), even when the axes are aligned with the CCD, the pan angle we computed above (θ) will not be the correct rotation angle (θ'). A moment's thought should reveal the impossibility of computing the correct θ' without knowing the depth of the subject. This is illustrated in Figure 1(b), where the pan angle is shown to be a function of the depth of the subject.

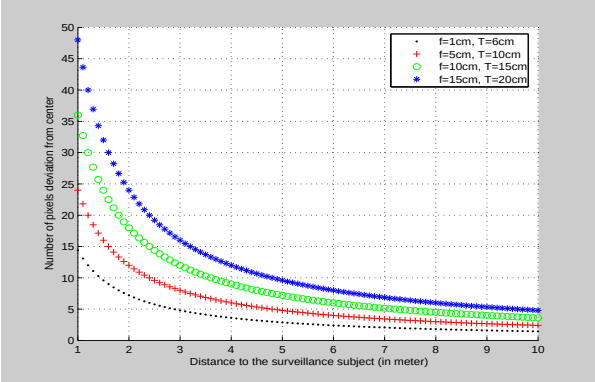


Figure 2. Error in centering assuming computational collocation of the optical center on the rotation axis.

In more detail, if we assume that the optical center and the centers of pan and tilt are collocated, and the axes align with the CCD as in Figure 1(a), we calculate the pan angle as $arctan((x - x_c)/(k_x f))$ to center the object in the dynamic camera. In reality, however, the optical center and the centers of pan and tilt may not be collocated, and if so, the angle thus calculated will not be entirely correct as shown in Figure 1(b). Executing the rotation maneuver will therefore not center the object. But how large can the error $(\theta - \theta')$ be, and how does that translate into real-world pixel error?

Figure 2 shows the pixel centering error as a function of the object distance for four different settings of focal length (f) and distance from the optical center to the pan (or tilt) axis ($\mathbf{T_p}$). In the simulation, we use real-world camera parameters of our Sony PTZ camera [2], where the CCD array size is $1/3''$ with about 480 pixels per scan line. The object can be as far as 10 meters, or as close as 1 meter (m), from the camera. The focal length can be as short as 1 centimeter (cm) (with $> 50^o$ wide fields-of-view) or as long as 15 cm (with 5^o narrow fields-of-view). Inasmuch as the location of the CCD array is fixed, changing the focal length will displace the optical center, thus altering the distance between the optical center and the axes. We assume a fixed displacement from the rotation axes to the CCD array to be about 4 cm, which corresponds to the real-world value for the Sony PTZ cameras. As can be seen, the centering error is small (less than 5 pixels but never zero) when the object is sufficiently far away. The centering error becomes unacceptable (> 20 pixels) when the object is getting closer (around 3 m) even with a modest zoom setting. Obviously, a much more accurate centering algorithm is needed.

It might seem that the centering problem could be solved if we either (1) adopt a mechanical design that ensures collocation of the optical center on the rotation axes, or failing that, (2) infer the depth of the subject to compute the rotation angle correctly. However, both solutions turn out to be infeasible because:

- In reality, it is often impossible to design a pan-tilt platform mechanically to ensure that the optical center falls on the rotation axes. To name a few reasons: (1) The mechanical designs have separate pan and tilt mechanisms, and the rotation axes are displaced with respect to each other. The optical center cannot lie on both axes at the same time. (2) A less accurate approach is to use a ball (or a socket) joint. Ball joints are not very desirable, and we are not aware of any commercial powered PTZ cameras that adopts this particular design, because of potential mechanical slippage and free play that degrade pan-and-tilt accuracy. (3) Finally, even if it were possible to use a ball joint and position the optical center optimally for a particular zoom setting, different zoom settings could displace the optical center.
- Depth information is critical for computing the correct pan- and-tilt angles. However, such information is only a necessary, not a sufficient condition. Although the pan angle can be uniquely determined from the x displacement and object depth in the simple configuration in Figure 1(b), generally nonzero pan- and tilt-angles will affect both x and y image coordinates. This is because when a pan-tilt camera is assembled, some non-zero deviation is likely in the orientation of the axes with respect to the camera's CCD. Mathematically, one can verify this coupling by multiplying the terms in Eq 1 and noting that θ and ϕ *have* each appeared in both of the (decidedly nonlinear) expressions for x and y coordinates.

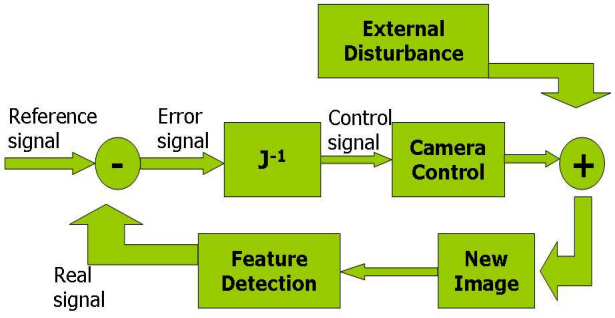


Figure 3. Selective focus-of-attention as a visual servo problem.

Instead, we formulate this selective, purposeful focus-of-attention problem as one of visual servo. Our visual servo

framework is modeled as a feedback control loop shown in Figure 3. This servo loop is repeated over time. As mentioned, the stationary cameras perform visual analysis to identify the current state (RGB, texture, position, and velocity) of the suspicious persons/vehicles. A similar analysis is performed by the dynamic cameras under the guidance of the stationary camera. Image features of the subjects (e.g., position and size of a car license plate or the face of a person) are computed and then serve as the input to the servo algorithm (the *real* signals). The real signals are compared with the *reference* signals, which specify the desired position (e.g., at the center of the image plane) and size (e.g., covering 80% of the image plane) of the image features. Deviation between the real and reference signals generates an error signal that is used to compute a camera control signal (i.e., desired changes in the pan, tilt, and zoom DOFs). Executing these recommended changes to the camera's DOFs will train and zoom the camera to minimize the discrepancy between the reference and real signals (i.e., to center the subject with a good size). Finally, as we have no control over the movements of the surveillance subjects, such movements must be considered external disturbance (noise) in the system.

In this paper, we do not address the video analysis and feature extraction processes, as there are many standard video analysis, tracking, and localization algorithms that can accomplish these. Instead, we discuss the detail of how to generate the camera control signals below.

Visual servo is based on Eq. 1 that relates the image coordinate to the world coordinate for the PTZ cameras. Assume that we sample at the video frame rate (30 frames/second) and at a particular instant we observe the tracked object at a location in the dynamic camera. Then, the questions we address here are:

1. Generally, what is the effect of changing the camera's DOF (f, θ, ϕ) on the tracked object's 2D image location?

2. Specifically, how can we manipulate the camera's DOFs to center the object?

One can expect, by a cursory examination of Eq. 1, that the relationship between image coordinates and the camera's DOFs to be fairly complicated and highly nonlinear. Hence, a closed-form solution to the above two questions is not likely. Instead, we linearize the problem by rearranging terms in Eq. 1 and taking the partial derivative of the resulting expressions with respect to the control variables (f, θ, ϕ). Taking partial derivatives of $x_r = fk_x\frac{x_i}{z_i} + x_o$ and $y_r = fk_y\frac{y_i}{z_i} + y_o$, where x_o and y_o are the corresponding projection centers in the CCD we have:

$$dx_r = k_x\frac{x_i}{z_i}df + fk_x\left(\frac{\partial(x_i/z_i)}{\partial\theta}\right)d\theta + fk_x\left(\frac{\partial(x_i/z_i)}{\partial\phi}\right)d\phi$$
$$dy_r = k_y\frac{y_i}{z_i}df + fk_y\left(\frac{\partial(y_i/z_i)}{\partial\theta}\right)d\theta + fk_y\left(\frac{\partial(y_i/z_i)}{\partial\phi}\right)d\phi \quad (3)$$

Eq. 3 can be represented in matrix form as follows:

$$\begin{bmatrix} dx_r \\ dy_r \end{bmatrix} = \begin{bmatrix} k_x\frac{x_i}{z_i} & fk_x\left(\frac{\partial(x_i/z_i)}{\partial\theta}\right) & fk_x\left(\frac{\partial(x_i/z_i)}{\partial\phi}\right) \\ k_y\frac{y_i}{z_i} & fk_y\left(\frac{\partial(y_i/z_i))}{\partial\theta}\right) & fk_y\left(\frac{\partial(y_i/z_i)}{\partial\phi}\right) \end{bmatrix}\begin{bmatrix} df \\ d\theta \\ d\phi \end{bmatrix},$$
$$= \mathbf{J}\begin{bmatrix} df \\ d\theta \\ d\phi \end{bmatrix}. \quad (4)$$

where k_x and k_y are the scale factors to convert real-world units into pixel in the x and y directions respectively. The expression of the Jacobian matrix $\mathbf{J}$ is a complicated and is not presented due to space constraints. However, it is a simple mathematic exercise to figure it out. The expression in Eq. 4 answers the first question we posed above. The answer to the second question is then obvious: we substitute $[x_c - x, y_c - y]^T$ for $[dx_r, dy_r]^T$ in Eq. 4 because that is the desired centering movement.

However, as Eq. 4 represents a linearized version of the original nonlinear problem (or its first-order Taylor series expansion), iterations are needed to converge to the true solution. Actually, the need for iterations does not present a problem, since computation is efficient and convergence is fast even with the simple Newton's method. In our experiments, convergence is always achieved within four iterations with $\approx 1/10,000$ of a pixel precision. Two final points worth mentioning are:

1. First, there are two equations (in terms of x and y displacements) and three (f, θ, ϕ) variables. Hence, it is not possible to obtain a unique solution. Our formulation manipulates (θ, ϕ) to control (x, y) to achieve the desired centering results. Once the object is centered, we use (f) to control the change in the object's size. That way, we have two DOFs with two equations to center a tracked object:

$$\begin{bmatrix} dx_r \\ dy_r \end{bmatrix} = \begin{bmatrix} fk_x\left(\frac{\partial(x_i/z_i)}{\partial\theta}\right) & fk_x\left(\frac{\partial(x_i/z_i)}{\partial\phi}\right) \\ fk_y\left(\frac{\partial(y_i/z_i))}{\partial\theta}\right) & fk_y\left(\frac{\partial(y_i/z_i)}{\partial\phi}\right) \end{bmatrix}\begin{bmatrix} d\theta \\ d\phi \end{bmatrix},$$
$$= \mathbf{J}_c\begin{bmatrix} d\theta \\ d\phi \end{bmatrix}. \quad (5)$$

It is easy to verify that Jacobian $\mathbf{J}_c$ is well conditioned and invertible using an intuitive argument. This is because the two columns of the Jacobian represent the instantaneous image velocities of the tracked point due to a change in the pan (θ) and tilt (ϕ) angles, respectively. As long as the instantaneous velocities are not collinear, $\mathbf{J}_c$ has independent columns and is therefore invertible. It is well known that degeneracy can occur only if a "gimbal lock" condition [4] occurs that reduces one DOF. For pan-tilt cameras, this occurs only when the camera is pointing straight up. In that case, the pan DOF reduces to a self- rotation of the camera body, which can make some image points move in a way similar to that under a tilt maneuver. This condition

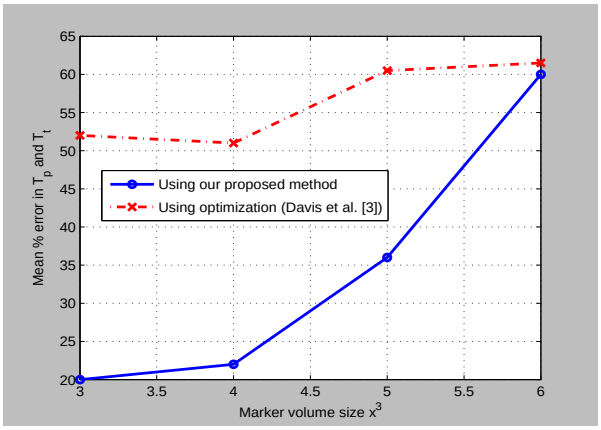


Figure 4. Comparison of calibration accuracy as a function of experimental setup (using a CCD of 300×300 pixels).

rarely occurs; in fact, it is not even possible for Sony PTZ cameras because the limited range of tilt does not allow the camera to point straight up.

2. Second, to uniquely specify the Jacobian, it is necessary to know the depth of the object. With the collaboration of stationary and dynamic cameras, it is possible to use standard stereo triangulation algorithms to obtain at least a rough estimate of the object depth.

4. Experimental Results

Off-line calibration: We compare the performance of our algorithm to that used in [3] to illustrate:

1. Theoretically, under the same simulation conditions, our method produces more accurate results without failure, while convergence cannot be guaranteed in [3].

2. Practically, our experimental set up using a traditional calibration mark placed near to the camera produces more reliable results than the virtual landmark approach of [3], regardless of the calibration procedure used.

The second claim deserves some explanation. We adopt the traditional method of using a planar checkerboard pattern placed at different depths before the camera to supply 3D calibration landmarks. While [3] advocates a different method of generating virtual 3D landmarks by moving an LED around in the environment. The argument used in [3] to support the virtual landmark approach is the need of a large working space to fully calibrate the pan and tilt DOFs. However, $\theta \approx r/d$, a large angular range can be achieved by either (1) placing a small calibration stand (small r) nearby (small d) or (2) using dispersed landmarks (large r) placed far away (large d).

Our reason of using a small volume is that to calibrate $\mathbf{T_p}$ and $\mathbf{T_t}$ accurately, we want their effects to be as pronounced as possible in image coordinates. This makes a near-field approach better than a far-field approach. As seen in Figure 2, when the object distance gets larger, whether or not the optical center is collocated with the axes of pan and tilt

Measurement Metric	Ref [3]	Ours
Mean % error in axis positions	51.76%	35.48%
Mean error in axis orientation	1.54 *rad*	0.22 *rad*

Table 1. Comparison of calibration accuracy. For [3], 51% **simulation runs failed to converge. If the simulation did converge,** 85 **iterations were needed in average.**

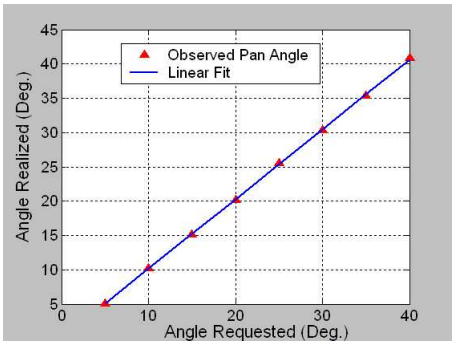
($\mathbf{T_p}$ and $\mathbf{T_t}$) becomes less consequential. Coupled with the localization errors in 3D and 2D landmarks, this makes it extremely difficult to calibrate $\mathbf{T_p}$ and $\mathbf{T_t}$ accurately using the approach presented in [3].

Another reason is that to provide the same angular calibration range, using the same focal length and CCD, would imply that the CCD's fixed and limited spatial resolution is used to cover either a small spatial range (r) in a near field or a large spatial range in a far field. Hence, the spatial resolution power necessarily becomes poorer when the calibration markers are placed afar. Figure 4 verifies the calibration error as a function of the volume occupied by the 3D calibration marks. As we shrank down the volume in front of the camera, the error in calibrating $\mathbf{T_p}$ and $\mathbf{T_t}$ dropped for both techniques as expected.

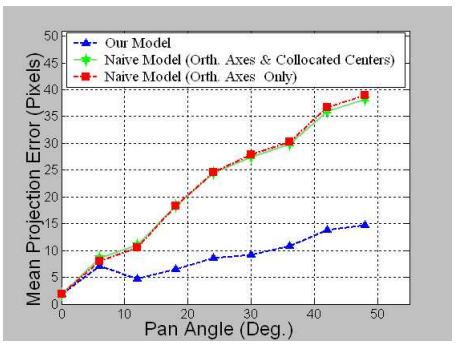
We validated the first claim this way: We conducted 100 synthetic experiments. In each experiment, we generated 50 3D landmarks randomly in an volume (similar to the one used in [3]). We projected these 50 landmarks using a synthetic camera that closely mimicked the real-world Sony EVI-D30 camera. We then applied both calibration procedures to estimate the pan and tilt camera parameters using these 50 2D and 3D coordinates. Because we did not have the codes of [3], we used Matlab's built-in nonlinear optimization function `fmincon` instead. In all simulation runs, we had chosen the initial guess of $\mathbf{T_p}$ and $\mathbf{T_t}$ to be zero, and $\mathbf{n_p}$ and $\mathbf{n_t}$ to be parallel to the CCD's y and x axes. We report the errors in calculating both the axis position and orientation in Table 1, averaged over these 100 runs.

For [3], we also recorded the percentage of times the algorithm failed to converge, and if it did converge, the number of iterations needed. As can be easily seen in Table 1 that under the same experimental conditions, our algorithm obtained more accurate results and did not suffer from convergence problem.

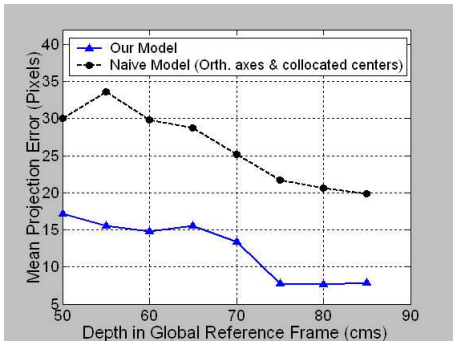
Experimental results using real images are summarized in Figure 5. We use Sony EVI-D30 cameras in our experiments [2]. The image size used is 768×480 pixels. Figure 5(a) shows the calibration results for $\hat{\theta} = g(\theta)$ and the best linear fit. As can be seen, the realized pan angles match well with the requested angles even for large rotations. Similar good results are obtained for $\hat{\phi} = g(\phi)$ (not shown here). To estimate $\mathbf{T_p}(f)$ we repeat the calibration procedures for a wide range of pan angles ($\theta = 5^o$ to 40^o in 5^o increment). The final $\mathbf{T_p}$ values are obtained by averaging the $\mathbf{T_p}$ values



(a) Relation between requested and realized angle of rotation for Sony PTZ camera.

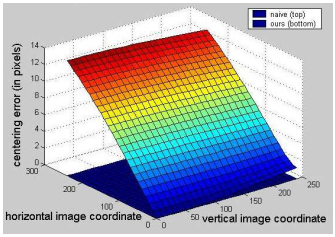


(b) Mean projection error as a function of pan angle

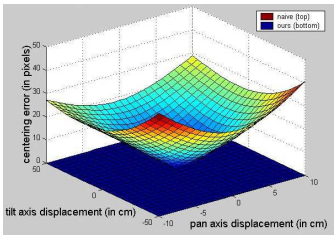


(c) Mean projection error as a function of depth for our model and naïve models.

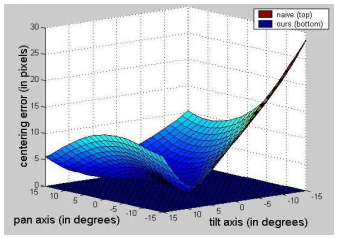
Figure 5. Off-line Calibration



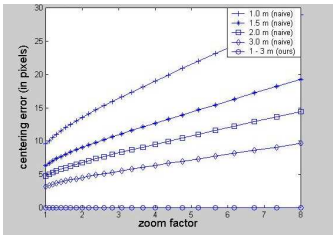
(a) Centering error for non-zero displacement of pan/tilt axis.



(b) Centering error as a function of displacement of pan/tilt axis.



(c) Centering error as a function of misalignment of pan/tilt axis.



(d) Effect of zoom on centering error.

Figure 6. Centering errors under various experimental conditions

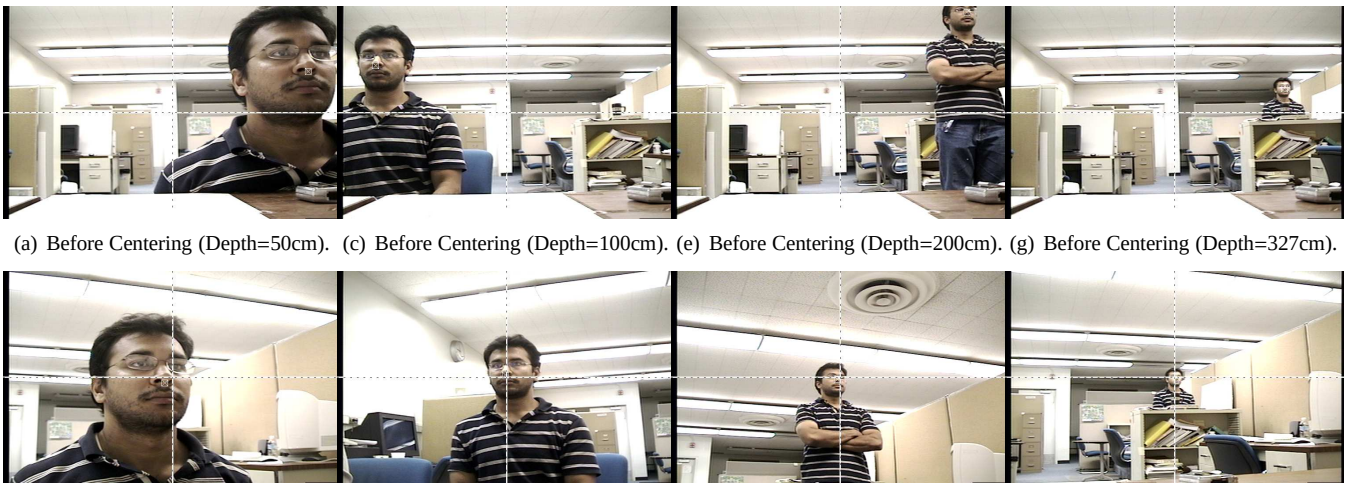
for different pan angles. The values for $\mathbf{T_t}$ are obtained in a similar fashion. For Sony cameras, our results show that the axes are well aligned with the camera's CCD. It enables the use of less expensive and less accurate motorized mounts for maneuvering dynamic cameras.

In Figures 5(c) and 5(b) we compare our model with two naïve models: one assuming collocated centers (i.e., the optical center is located on the pan and tilt axes) and orthogonal axes (i.e., the axes are aligned with the CCD), and the other assuming independent centers but orthogonal axes. The mean projection errors in Figures 5(c) and 5(b) are obtained for all three models (ours and two naïve ones) by (1) mathematically projecting 3D calibration landmarks onto the image plane using the camera parameters computed based on these three models, and (2) comparing the observed landmark positions with the mathematic predictions and averaging the deviations. Figure 5(c) shows the mean projection error as a function of pan angle. As seen in Figure 5(c), the performance of the naïve models is much worse than ours. Figure 5(c) suggests that the naïve models could completely lose track of an object at high zoom. At a pan angle of 45^o the projection error of the naïve models can be as large as 38 pixels at $\sim$ 1 m!

In Figure 5(b) we show the mean projection error as the depth of the calibration landmarks from the camera changes. As evident the projection error gets smaller as the object moves further away from the camera. However even at large distances the performance of our model is much better than that of the naïve one. It could be argued that a correct model like ours is important only when the object is close to the camera. However, surveillance cameras often have to zoom at a high value (as high as 20x). In such cases the object appears very close to the camera, and the projection error becomes unacceptable for the naïve models.

On-line focus-of-attention: We conducted experiments using both synthesized and real data. For synthesized data, we compared the accuracy of our algorithm with that of a naïve centering algorithm. The naïve algorithm makes the following assumptions: (1) the optical center is collocated on the axes of pan and tilt, and (2) the pan DOF affects only the x coordinates whereas the tilt DOF affects only the y coordinates. While those assumptions are not generally valid, algorithms making those assumptions can and do serve as good baselines because they are easy to implement, and give reasonable approximations for far-field applications.

In more detail the naïve algorithm works as follows: Assume that a tracked object appears in the dynamic camera at location $p_i = [x_i/z_i, y_i/z_i, 1]^T$ as defined in Eq. 3. We apply pan rotation in the y direction as $\mathbf{p}'_i = \mathbf{R}_y(\theta)\mathbf{p}_i$, and tilt rotation in the x direction as $\mathbf{p}''_i = \mathbf{R}_x(\phi)\mathbf{p}'_i$, where $\theta = arctan((x - x_c)/(k_x f))$ and $\phi = arctan((y' - y_c)/(k_y f))$. To make the simulation realistic, we use the parameters of Sony PTZ cameras. Figure 6(a) compares the centering error (in pixels) of our method and the naïve method with different starting image positions. Here we assume that the distances from the optical center to the panning and tilt axes



(a) Before Centering (Depth=50cm). (c) Before Centering (Depth=100cm). (e) Before Centering (Depth=200cm). (g) Before Centering (Depth=327cm).

(b) After Centering (Depth=50cm). (d) After Centering (Depth=100cm). (f) After Centering (Depth=200cm). (h) After Centering (Depth=327cm).

Figure 7. Focus-of-attention experiments using real video

are 5 cm ($\mathbf{T_p}$) and 2.5 cm ($\mathbf{T_t}$), respectively. These values are chosen to be similar to those of Sony cameras. A depth of $\sim$ 1m is assumed. The error of our method is less than 0.01 pixels for any starting point and convergence is always achieved in 4 iterations or less. By contrast, the naïve method, which does not take into account the displacement of the pan/tilt axes from the optical center, can be seen to center the point inaccurately, with errors as large as 14 pixels. Figure 6(b) shows similar results as 6(a), except that a single starting location is assumed (the upper-left corner of the image) and the centering error is displayed as a function of the displacements of the pan and tilt axes.

Another source of error of the naïve method involves the possible misalignment of the pan/tilt axes. While the naïve method assumes that the axes are perfectly aligned with the CCD, in reality some deviation should be expected. Figure 6(c) compares the accuracy of our method and the naïve method when axes are not perfectly aligned. We use a single starting location and plots centering error as a function of the misalignment of the pan and tilt axes. Again, our method gives almost perfect results while the naïve algorithm's results are sensitive to error in axes alignment.

Figure 6(d) exhibits the effect of introducing zoom. Intuitively, it makes sense that increasing zoom, for a given object depth, will cause the error of the naïve method to increase. This can be understood as zoom causing an object's *effective* depth to decrease. A smaller effective depth means that the effect of a non-zero pan/tilt axis displacement will be more significant to the centering problem. In this graph, the centering error is plotted as a function of zoom factor for various depths, and, as anticipated, increasing zoom lowers the accuracy of the naïve centering algorithm. Thus, it is clear that when the camera exploits its zoom capabilities (as is typically the case for surveillance), the use of a precise centering algorithm becomes even more critical.

We test the performance of the centering algorithm on real data as well. A person is made to stand in front of the camera at an arbitrary position. The centering algorithm then centers the tip of the nose of the person. Figure 7 shows the centering achieved using our proposed method. The center of the screen has been marked by dotted white lines and the nose tip by the white square, both before and after centering. The centering error reduces as the object gets farther away from the camera; however, the centering results are good even when the object (the face) gets as close as 50 cm to the camera.

5. Conclusions

We have presented algorithms for off-line calibration and on-line selective focus-of-attention using stationary-dynamic camera assemblies. We have supported our claims of robustness and accuracy through extensive validation.

References

[1] OpenCV, http://www.intel.com/research/mrl/research/opencv/.

[2] Sony Co-op. Sony EVI-D30 pan/tilt/zoom video camera user manual, 2003.

[3] J. Davis and X. Chen. Calibrating pan-tilt cameras in wide-area surveillance networks. In *ICCV '03: Proceedings of the Ninth IEEE International Conference on Computer Vision*, page 144, Washington, DC, USA, 2003. IEEE Computer Society.

[4] A. Watt and M. Watt. *Advanced Animation and Rendering Techniques: Theory and Practice*. Addison-Wesley, 1992.

[5] X. Zhou, R. T. Collins, T. Kanade, and P. Metes. A master-slave system to acquire biometric imagery of humans at distance. In *IWVS '03: First ACM SIGMM international workshop on Video surveillance*, pages 113–120, New York, NY, USA, 2003. ACM Press.