

# Interactive Calibration of a PTZ Camera for Surveillance Applications

Miroslav Trajković

*Philips Research USA, 345 Scarborough Rd., Briarcliff Manor, NY 10510, USA*

*Miroslav.Trajkovic@Philips.com*

## Abstract

*In this paper, we describe a novel method for easy and precise external and internal calibration of pan-tilt-zoom cameras for surveillance applications. The external calibration module assumes known height of the camera and allows an installer to determine camera position and orientation by pointing the camera at several points in the area and clicking on their respective position on the area map shown in the GUI. The only requirements for internal camera calibration are that the maximum zoom-out of the camera is known (this is typically provided by the manufacturer) and that the installer has pointed the camera to a texture-rich area. We compute not only focal length, pixel aspect ratio and principal point, but also, the relationship between camera zoom settings and the focal length. Our calibration method provides accurate and consistent results and is currently under commercial implementation.*

## 1. Introduction

Pan-tilt-zoom cameras (stationary, but rotating and zooming) are often used in surveillance applications. The main advantage of a PTZ camera is that one camera can be used for the surveillance of a large area, yet it can also be used to closely look at the points of interest.

As the layout of the surveillance areas is prone to changes, the installer often wants to move a camera from one position to another. In this case, it is important to him to have a simple procedure to determine camera position and orientation in reference to the surveillance area. The knowledge of camera position and orientation is crucial for geometric reasoning. This, in turn, enables the operator to use some useful functionalities, such where the operator clicks on the map, and the camera automatically points to this direction (or in the case of multiple cameras, the closest camera points to this

direction), displaying current and entire viewspace of the camera, etc.

Knowledge of internal camera calibration parameters is also important for a variety of useful tasks, including tracking with a rotating camera, obtaining metric measurements, knowing how much to zoom to achieve a desired view, etc. Again, it is of utmost importance to develop a simple procedure that will enable the installer with little or no technical training to perform calibration of all the cameras covering the surveillance area, or even better, to develop a method that will perform calibration entirely automatically.

There has been much work in the area of self-calibration, starting with the seminal work of Maybank and Faugeras [7], in which they have shown that the camera calibration parameters can be computed from the three snapshots of the environment, provided that sufficiently many point correspondences between each of the three image pairs can be established. In general, the self-calibration methods that deal with unconstrained camera motion require good initial values and the minimization of the complex cost function, and are, therefore, not always feasible. When some constraints on camera motion are imposed (i.e. purely translational [4], purely rotational [6], or purely rotational with known motion parameters [2, 5]), much simpler, and typically more precise procedures for camera calibration are obtained.

The subject of zoom-camera self-calibration has only recently received some attention. Agapito *et. al.* [1] proposed a linear algorithm for the self-calibration of a rotating and zooming camera, assuming zero skew (or more restrictive conditions of square pixels, known pixel aspect ratio, and known principal point), and allowing for the variable principal point. Their algorithm is linear and very rapid, but the principal point is very unstable (it varies over more than 200 pixels). The disadvantages of this and the other self-calibration methods are that they require precise image correspondences, and are very sensitive to noise. Also,

they do not model the camera focal length as a function of zoom settings.

Batista *et al.* [3] did not consider self-calibration, but they tried to model motorized zoom lenses. They used modeled focal length  $f$  and focused target depth  $D$  as an  $n^{\text{th}}$  order bivariate polynomial in camera zoom and focus settings. As shown in section 4, this model is not adequate, and a better model is proposed.

In this paper, we describe procedures for external and internal self-calibration of a PTZ camera. It is assumed that the user has positioned the camera over a texture-rich area with features or a calibration object and that the height of the camera is known. The procedure then determines:

- Camera position;
- Camera orientation (represented by Pan and Tilt bias angles);
- Camera Center or Principal Point; and
- Mapping from zoom settings (ticks) to focal length; The algorithm assumes that the camera principal point and the center of rotation of the pan and tilt units coincide. The distance between these two points is usually small and therefore this assumption can be made. We also assume that the skew factor is zero, hence, the calibration matrix is of the form:

$$Q = \begin{bmatrix} f_x & 0 & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathcal{O} & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

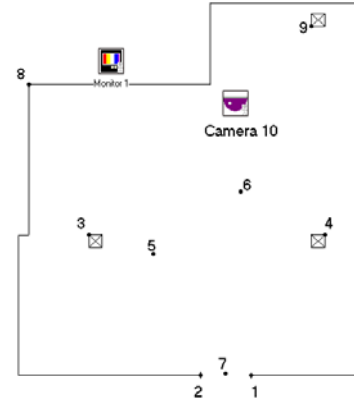
The remainder of the paper is organized as follows. Notation and background are given in section 2. The estimation of the external calibration parameters (pan and tilt bias and camera position) is developed in section 3. Internal camera calibration (estimation of principal point, focal lengths and mapping from zoom ticks to focal length) is addressed in section 4, along with some experimental results. Concluding remarks are given in section 5 and references are provided in section 6.

## 2. Notation

Let us suppose that the security operator is monitoring an area represented by the map in Figure 1.

Let  $OX_wY_wZ_w$  denote the three-dimensional coordinate system of the room and let  $C$  denote the location of the camera. We will refer to  $OX_wY_wZ_w$  as the *world coordinate system*. Let  $Cx'_c y'_c z'_c$  denote the camera coordinate system for the zero pan and tilt angles and let the  $x'_c$  axis coincide with the optical axes of the camera. If a user wants to point the camera toward point  $A$  it is necessary to rotate the camera around the pan and

tilt axes by the angles  $\alpha_m$  and  $\beta_m$  respectively. To compute these angles, one must know the world coordinates of the camera  $(X_C, Y_C, Z_C)$ , point  $A (X_i, Y_i, Z_i)$ , and the orientation of the camera in the world coordinate system, or more conveniently, the orientation of the camera in the normalized camera coordinate system  $Cx_c y_c z_c$ , obtained by translating the world coordinate system from  $O$  to  $C$ . The camera orientation can be represented by the angles  $\alpha_{\text{offset}}, \beta_{\text{offset}}$  and  $\gamma_{\text{offset}}$  and these angles will be called the pan, tilt and roll bias respectively. Instead of the tilt and roll bias it may be convenient to use substitutes  $\varphi_x = \beta \cos \gamma$  and  $\varphi_y = \beta \sin \gamma$ .



**Figure 1:** GUI used for CCTV surveillance. The operator can insert a camera at any position on the map and compute its position, orientation, and internal calibration using the simple user-friendly procedure presented in this paper.

Furthermore, we can define the tilt bias as a function of the pan angle and it will have the form:

$$\varphi(\alpha) = \varphi_x \cos \alpha + \varphi_y \sin \alpha. \quad (2)$$

For each camera setting, one obtains:

$$\begin{aligned} \alpha_i + \alpha_{\text{offset}} &= \text{atan2} \left( \frac{Y_{iC} - \varphi_y Z_{iC}}{X_{iC} - \varphi_x Z_{iC}} \right) \\ \tau_i + \varphi(\alpha_i) &= \text{atan} \left( \frac{Z_{iC}}{\sqrt{X_{iC}^2 + Y_{iC}^2}} \right). \end{aligned} \quad (3)$$

where

$$X_{iC} = X_i - X_C, \quad Y_{iC} = Y_i - Y_C, \quad Z_{iC} = Z_i - Z_C,$$

Given  $n$  camera settings ( $n \geq 3$ ), the camera calibration parameters may be estimated by minimizing the cost function corresponding to the equation (4):

$$\begin{aligned} f(\mathbf{P}_C, \boldsymbol{\omega}) &= \sum_{i=1}^n (\alpha_i + \alpha_{\text{off}} - \text{atan2} \frac{Y_{iC} - \varphi_y Z_{iC}}{X_{iC} - \varphi_x Z_{iC}})^2 \\ &+ \sum_{i=1}^n (\tau_i + \varphi(\alpha_i) - \text{atan} \frac{Z_{iC}}{\sqrt{X_{iC}^2 + Y_{iC}^2}})^2 \end{aligned} \quad (4)$$

where  $\mathbf{P}_C = (X_C, Y_C, Z_C)$  and  $\boldsymbol{\omega} = (\alpha_{\text{offset}}, \varphi_x, \varphi_y)$ .

### 3. External Calibration

In this section the algorithms for the computation of the camera position and orientation are presented. The algorithms are presented in increasing complexity. First, assuming that the camera position is known and that the tilt bias is approximately zero, we present an algorithm for the estimation of the pan bias. We then assume that the camera position is unknown, and present the algorithm for the estimation of camera position and the pan bias, assuming that tilt bias is approximately zero. Finally, we present an algorithm for the tilt bias estimation, assuming that the camera position and the pan bias are known.

The order of algorithms presented here follows our current implementation. Namely, we first use a set of at least three camera settings (for which the tilt bias can be neglected) to compute the pan bias and camera position. Then, knowing the pan bias and camera position, we use another set of at least three camera settings to compute the tilt bias.

#### 3.1. Pan bias estimation

There are numerous ways to estimate the pan bias from equations (3) and (4). The simplest scenario occurs if the camera position is exactly known and the tilt bias is assumed to be zero. In this case, the pan bias can be estimated directly from equation (3), and for  $n$  measurements, the least squares solution is given by:

$$\alpha_{\text{offset}} = \frac{1}{n} \sum_{i=1}^n (\text{atan2}(Y_{iC}, X_{iC}) - \alpha_i). \quad (5)$$

For better precision, it is advisable to choose reference world points to be at a height similar to that of the camera. In this case the term  $Z_{iC}$  in equation (3) will be close to zero, and as the tilt bias ( $\varphi_x$  and  $\varphi_y$ ) is usually close to zero, the terms  $\varphi_x Z_{iC}$  and  $\varphi_y Z_{iC}$  in equation (3) can be neglected.

#### 3.2 Camera position and pan bias estimation

Let us assume that only the camera height is known and that  $X_C$  and  $Y_C$  components of the camera position are only approximate. As before, it can be assumed that the tilt bias is zero, as explained in the previous section. The camera position and pan bias can now be computed using a linear algorithm. Let us consider the first equation in (3) assuming  $\varphi_x \approx \varphi_y = 0$ . This equation now becomes:

$$\alpha_i + \alpha_{\text{offset}} = \text{atan2} \left( \frac{Y_i - Y_C}{X_i - X_C} \right). \quad (6a)$$

After applying the tan operation to the both sides of equation (6a) and rearrangement, it can be written as:

$$m_0(X_i + Y_i t_i) - m_1 - m_2 t_i - (Y_i - t_i X_i) = 0, \quad (6b)$$

where:

$$\begin{aligned} t_i &= \tan \alpha_i, m_0 = \tan \alpha, \\ m_1 &= tX_C - Y_C, m_2 = X_C + tY_C \end{aligned}$$

Given three or more measurements ( $\alpha_i, X_i, Y_i$ ), vector  $\mathbf{m}$  can be determined using least squares. Once  $\mathbf{m}$  is computed, the camera position and pan bias can be easily found.

This linear algorithm usually produces quite good results, but since it doesn't minimize a geometrically meaningful criterion (it minimizes a cost function associated with equation (6b), which is different from the optimal cost function associated with equation (6a)), it doesn't produce the optimal result.

The optimal camera position and pan bias can be estimated by minimizing the cost function (7) associated with the first equation in (3):

$$\begin{aligned} f(X_C, Y_C, \alpha_{\text{offset}}) = \\ \sum_{i=1}^n (\alpha_i + \alpha_{\text{offset}} - \text{atan2}(Y_{iC}, X_{iC}))^2 \end{aligned} \quad (7)$$

As this is a nonlinear function, the solution has to be found numerically. In our implementation we have used conjugate gradients to minimize the cost function, and the solution with the precision of 0.0001 is typically found in three to four iterations. As initial values we use the solution obtained from the linear algorithm.

#### 3.3. Tilt bias estimation

In the current implementation, the tilt bias is estimated after the camera position and the pan bias have been computed. Then the tilt bias can be estimated from the second equation of (3). However, we have experimentally found that better results can be obtained using the following empirical model instead of (3):

$$\varphi(\alpha) = \varphi_0 + \varphi_x \cos \alpha + \varphi_y \sin \alpha. \quad (8)$$

The factor  $\varphi_0$  in equation (8) accounts for the mechanical imperfection of the tilt mechanism and the fact that the camera may be unable to perform the zero tilt. The experimental results have justified the introduction of this factor and the prediction error was significantly reduced.

By substituting (8) into the second equation of (3), we obtain:

$$\begin{aligned} & \varphi_0 + \varphi_x \cos \alpha + \varphi_y \sin \alpha = \\ & \operatorname{atan}\left(\frac{Z_{iC}}{\sqrt{X_{iC}^2 + Y_{iC}^2}}\right) - \tau_i, \quad i=1, \dots, n. \end{aligned} \quad (9)$$

Equation (9) is linear in  $\boldsymbol{\varphi} = (\varphi_0, \varphi_x, \varphi_y)$  and the tilt bias parameters can be estimated using the Least Squares and solving a system of three linear equations in  $\boldsymbol{\varphi}$ . The minimum number of points required is  $n = 3$ .

In order to obtain the estimate of  $Z_{iC}$  it is necessary to choose the world points with known heights. Typically these points are either on the ceiling or on the floor. If the points are chosen on the ceiling, then the term  $\operatorname{atan}\left(\frac{Z_{iC}}{\sqrt{X_{iC}^2 + Y_{iC}^2}}\right) - \tau_i$  becomes unreliable, so these

points should not be used. It is also possible to obtain the tilt bias by minimizing the cost function (4) assuming that the camera position and the pan bias are known. However, our experiments suggest that this would not give stable and reliable results and should not be used.

## 4. Internal calibration

In this section we give algorithms for the estimation of the principal point, focal length, pixel aspect ratio and mapping from zoom ticks to focal length.

The principal point is estimated first, as it can be estimated independently of the focal length. Once the principal point is estimated, the focal length and pixel aspect ratio are estimated for several zoom settings. Finally, the mapping between zoom settings and focal length is computed, taking into account the nature of the problem.

### 4.1. Principal point estimation

The principal point is estimated using images collected at minimum and maximum zoom settings ( $z_1$  &  $z_2$ ) and at fixed pan and tilt angles. It is assumed that the ratio between maximum and minimum zoom-in is known and is obtained from camera specifications. It is also assumed that the principal point does not change with the zoom, for the justification, please refer to [8] (the authors found that the principal point changes very little with the zoom and that it has weak influence on calibration results).

Let  $s$  denote the scale factor  $f_1 / f_2$  (Note that  $s = f_{x1}/f_{x2} = f_{y1}/f_{y2}$ ). The positions of the point  $\mathbf{P}$  in two consecutive images are given as:

$$x_1 = f_{x1} \frac{\mathbf{r}_1^T \mathbf{P}}{\mathbf{r}_3^T \mathbf{P}} + x_0 \quad y_1 = f_{y1} \frac{\mathbf{r}_2^T \mathbf{P}}{\mathbf{r}_3^T \mathbf{P}} + y_0 \quad (10)$$

$$x_2 = f_{x2} \frac{\mathbf{r}_1^T \mathbf{P}}{\mathbf{r}_3^T \mathbf{P}} + x_0 \quad y_2 = f_{y2} \frac{\mathbf{r}_2^T \mathbf{P}}{\mathbf{r}_3^T \mathbf{P}} + y_0, \quad (11)$$

where  $R = [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3]^T$  denotes the rotation (*i.e.* orientation) matrix. Combining equations (10) and (11) we obtain:

$$\begin{aligned} x_2 &= \frac{f_{x2}}{f_{x1}} f_{x1} \frac{\mathbf{r}_1^T \mathbf{P}}{\mathbf{r}_3^T \mathbf{P}} + \frac{f_{x2}}{f_{x1}} x_0 + x_0 - \frac{f_{x2}}{f_{x1}} x_0 \\ &= \frac{f_{x2}}{f_{x1}} \left( f_{x1} \frac{\mathbf{r}_1^T \mathbf{P}}{\mathbf{r}_3^T \mathbf{P}} + x_0 \right) + x_0 \left( 1 - \frac{f_{x2}}{f_{x1}} \right) \\ &= s x_1 + x_0 (1 - s) \\ y_2 &= s y_1 + y_0 (1 - s) \end{aligned} \quad (12)$$

Equation (12) may be written as:

$$\begin{aligned} x_2 &= s(x_1 - x_0) + x_0 \\ y_2 &= s(y_1 - y_0) + y_0 \end{aligned} \quad (13)$$

From equation (13) it may be concluded that the second image may be obtained from the first one by expanding it radially from the point  $(x_0, y_0)$ . Note that the camera center is invariant under this transformation (*i.e.*  $f(x_0, y_0) = (x_0, y_0)$ ).

Using the above facts, the principal point may be estimated in the following manner (without loss of generality, we will assume that  $s > 1$ ):

1. Create a template  $T$  by reducing the size of the second image by the factor  $s$ .
2. Find the best match for the second template in the first image. The position of the best match corresponds to the camera center (due to its invariance to scaling).

### 4.2. Focal length estimation

For a particular zoom setting, estimation of the focal length is performed by taking two images at fixed pan and different tilt settings and finding the displacement of the principal point  $d$ . The focal length is then computed as a function of  $d$  and the tilt difference ( $\alpha$ ) between two settings, as shown below.

Let  $A$  be an arbitrary point in the world, and let  $\mathbf{P}$  and  $\mathbf{P}'$  denote its world coordinates in the coordinate systems of the camera with different tilt settings. It may be shown that:

$$\begin{aligned} X' &= X \\ Y' &= Y \cos \alpha - Z \sin \alpha \\ Z' &= Y \sin \alpha + Z \cos \alpha \end{aligned} \quad (14)$$

Using similar reasoning as for equations (10) and (11), the positions of the points in two consecutive frames are given by:

$$x = f_x \frac{X}{Z} + x_0 \quad y = f_y \frac{Y}{Z} + y_0 \quad (15)$$

$$x' = f \frac{X'}{Z'} + x_0 \quad y' = f \frac{Y'}{Z'} + y_0 \quad (16)$$

By introducing new variables  $x_n = x - x_0$  and  $y_n = y - y_0$ , from equation (9) we have

$$\frac{X}{Z} = \frac{x_n}{f} \quad \frac{Y}{Z} = \frac{y_n}{f}. \quad (17)$$

In a similar way:

$$x'_n = f \frac{x_n}{y_n \sin \alpha + f \cos \alpha} \quad (18)$$

$$y'_n = f \frac{y_n \cos \alpha - f \sin \alpha}{y_n \sin \alpha + f \cos \alpha} \quad (19)$$

The coordinates of the principal point in the first image are given by  $(x_n, y_n) = (0, 0)$ . The coordinates of its correspondence in the second image can be computed from (18) and (19) and we obtain

$$\begin{aligned} x'_n &= 0 \\ y'_n &= -f \tan \alpha \end{aligned} \quad (20)$$

From (20) it may be concluded that the projection of the principal point will move along the  $y$  axis only and that this displacement can be easily found using template matching. Once the displacement is found, the focal length can be computed as

$$f = -\frac{d}{\tan \alpha} \quad (21)$$

### 4.3. Estimation of the pixel aspect ratio

As opposed to the estimation of the focal length, where the camera has performed pan rotation only, for the estimation of the pixel aspect ratio (or equivalently  $f_x$ ) when the focal length and principal point are known, any known camera rotation may be considered.

Let  $R = [r_{ij}]_{3 \times 3} = R(\alpha, \beta)$  denote a known rotation of the camera. Using notation introduced in section 4.1, we have

$$\mathbf{P}' = \mathbf{R}\mathbf{P},$$

and using a similar derivation, we obtain

$$\begin{aligned} x'_n &= f_x \frac{f'_{11}x_n + f_x r_{12}y_n + f_x f'_{13}}{f'_{31}x_n + f_x r_{32}y_n + f_x f'_{33}} \\ y'_n &= f \frac{f'_{21}x_n + f_x r_{22}y_n + f_x f'_{23}}{f'_{31}x_n + f_x r_{32}y_n + f_x f'_{33}}. \end{aligned}$$

Having in mind that the coordinates of the principal point are (0,0), the coordinates of its correspondence in the second image can be computed as

$$x'_p = f_x \frac{r_{13}}{r_{33}}, \quad y'_p = f \frac{r_{23}}{r_{33}}$$

From (20) it may be concluded that the projection of the principal point will move along the  $y = y'_p$  only and this displacement can be easily found using template matching. Once  $x'_p$  is found,  $f_x$  can be computed as

$$f_x = x'_p \frac{r_{33}}{r_{13}}.$$

### 4.4. Focal length fitting

Given the focal length estimated for the several zoom settings, our goal is to find mapping between the zoom setting and the focal length. In this section, we will first propose a mapping function, based on the analogy with the multi-lens system. We will then show that this form has desirable numerical properties (stability and linear computation) and finally, we will show how to compute the coefficients of the mapping function.

As known from Newton's law, the combined focal length from the system of two lenses with the focal lengths  $f_1$  and  $f_2$ , at distance  $d$ , is given by:

$$\frac{1}{f} = \frac{1}{f_1} + \frac{1}{f_2} - \frac{d}{f_1 f_2},$$

or equivalently,

$$f(d) = \frac{f_1 f_2}{f_1 + f_2 - d}$$

As motorized lenses are more complex than the ideal two lens system, we propose the following model:

$$f(t) = \frac{a_0}{1 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + \dots} \quad (22)$$

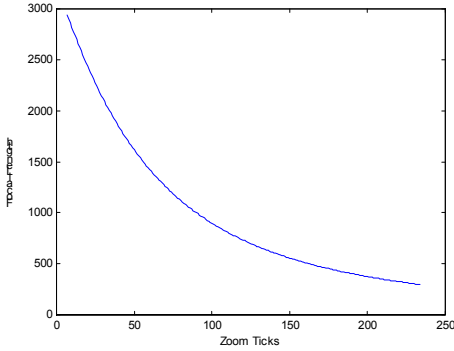
where  $t$  denotes zoom setting, typically given in ticks. The order  $n$  of the polynomial in the denominator is generally unknown, but our experiments have shown that this order should be 2. One way of finding the optimal  $n$  is to compute the coefficients for the different values of  $n$ , and then compute the ratio between focal lengths obtained by the model for the maximum and minimum zoom settings and compare it with the zoom power given by the manufacturer. It is this experiment, that gave us value of  $n = 2$ . One example of a curve representing (22) is given in Figure 2.

Coefficients  $a_0$ ,  $a_1$  and  $a_2$  can be directly estimated by minimizing the objective function

$$C(\mathbf{a}) = \sum_{i=1}^n \left( f(t_i) - \frac{a_0}{1 + a_1 t_i + a_2 t_i^2} \right)^2, \quad (23)$$

*i.e.* by fitting  $\mathbf{a}$  directly to the measurements of focal length. This direct approach poses two problems:

1. The objective function is nonlinear and an iterative method for minimization has to be used;
2. The computation of the focal length is much less reliable for low zoom ticks (high focal length) than for high zoom ticks. Therefore, the objective function (23) gives higher weights to worse estimates, and thus the estimates of  $a_0$ ,  $a_1$  and  $a_2$  will deteriorate.



**Figure 2:** Typical curve showing focal length as a function of zoom ticks.

As will be shown below, both of these problems may be overcome by using *lens power* rather than focal length of the lenses. Lens power is defined as the inverse of focal length

$$p(t) = \frac{1}{f(t)}$$

and, by substituting it in equation (22) (for  $n = 2$ ), we obtain

$$p(t) = b_0 + b_1 t + b_2 t^2 \quad (24)$$

where  $b_0 = 1/a_0$ ,  $b_1 = a_1/a_0$  and  $b_2 = a_2/a_0$ .

The corresponding objective function is now of the form

$$C(\mathbf{a}) = \sum_{i=1}^n \left( p(t_i) - (b_0 + b_1 t_i + b_2 t_i^2) \right)^2 \quad (25)$$

and this function overcomes both shortcomings of the objective function (23). Its minimization is linear, and the less reliable measurements (low lens power) are given lower weight (as the absolute variation in measurements is low, although relative variation remains higher). Moreover, we can employ the fact that the ratio between minimum and maximum zoom-in is known ( $s$ ), which can be written in terms of lens power and  $t_{min}$  and  $t_{max}$  (min and max zoom ticks) as:

$$sp(t_{min}) - p(t_{max}) = 0. \quad (26)$$

This constraint can be enforced either through the Lagrange multipliers, or, more easily, by expressing one of the coefficients  $b_0$ ,  $b_1$ ,  $b_2$  as the function of other two, using (26). As  $b_0 > b_1 > b_2$ , the best way (numerically) is to express  $b_2$  as a function of  $b_0$  and  $b_1$ , leading to set of linear equations in  $b_0$  and  $b_1$ .

Pixel aspect ratio  $\sigma$  can be estimated in a similar manner, although the exact solution will require solving a fourth order polynomial in  $\sigma$ .

## 5. Experimental results

To verify the validity of our calibration procedure we have performed following set of experiments.

1. Measure position and orientation (pan and tilt bias) of the camera. The validity of this is verified by clicking at some location of the map and check how close to this location is camera directed. This is evaluated only subjectively.
2. Measure internal camera calibration by choosing several textured regions in the image as starting points and comparing the results.
3. Use camera to measure height of the object in the image at the different positions in the room, and for different zoom setting.

Results are presented for the camera labeled as Camera 10, shown in the Figure 1.

### 5.1 Camera position and orientation

For the computation of position and pan bias, we have used the points labeled 1,2,3 and 4, measured as close to the ceiling as possible, while for the computation of the tilt bias we used points 1, 5, 2, and 3 on the floor, and obtained the following results:

**Camera position:** (5.51m, 0.31m) camera position coordinates are given in the room coordinate system with origin in point 8, and are determined with the error of about 10cm (<2%). **Pan bias:**  $\alpha_{\text{offset}} = 3.394$ ; and **Tilt bias:**  $\varphi = [.003748 \ -0.050125 \ -0.057554]^T$ .

Generally, the external calibration error is a consequence of the fact that camera will rarely point exactly to the point at the map that the operator has selected, and these errors are of the order of few centimeters.

The validity of the external calibration is confirmed by clicking at an arbitrary point on the map and having camera automatically point at this point.

### 5.2. Internal calibration

The internal camera calibration parameters have been measured for camera pointed at three different positions

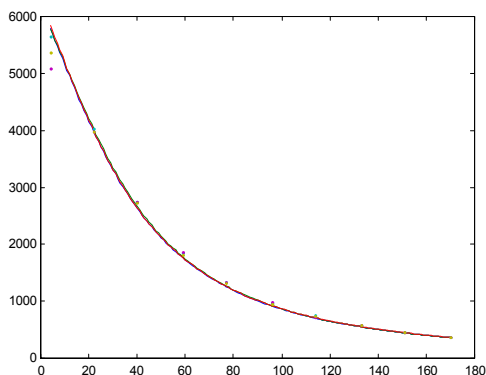
in the room, each with different texture patterns, and the obtained results are presented in Table 1. As we can see from Table 1, the Principal point is computed very consistently. The different value for  $x_0$  in the third measurement is the consequence of the imprecision of template matching, which does not always provide consistent results. The same is true for the slight inconsistency in other parameters, and we can see from the table that they are very consistently computed.

	$(x_0, y_0)$	$\sigma$	focal length (a)
1	167.47, 119.47	0.9624	6168.5, 0.0148, 4.69e-4
2	167.47, 119.47	0.9706	6129.0, 0.0134, 4.74e-4
3	168.53, 119.47	0.9704	6253.8, 0.0156, 4.66e-4

**Table 1:** Internal calibration results for camera at different pan and tilt settings pointing at various texture regions in the room.

Maximum relative difference in measurements of  $\sigma$  is lower than 1%, and maximum error in focal length for any zoom setting is less than 1.5%.

Figure 3 shows the estimated focal lengths for various zoom ticks for different camera positions (given by dots), and the focal length mappings for the all zoom settings (4 to 170) using focal length polynomials from Table 1.



**Figure 3:** Focal length measurements (shown by dots) and focal length polynomials (lines).

As it can be seen from Figure 3, the focal length measurements are almost identical, except for the lowest zoom settings (high focal length) which are unreliable. The reason for unreliability is twofold. First, from equation (21), we can see that the focal length is proportional to  $1/\tan(\alpha)$ , and  $d$ . For high zoom, in order to see the same scene in both images,  $\alpha$  has to be small, and then  $1/\tan(\alpha)$  is large. Hence, even a small imprecision in  $d$  (obtained by template matching) will result in a high error in focal length. On the other hand, with high zoom there is typically less texture in the

scene, and therefore the precision in the template matching is lower.

### 5.3. Height measurement

Finally, to get the estimate of the overall performance of the system, we have measured the height of several objects / people at different locations in a room.

First, we have measured a height of the door shown in Figure 4a. The distance from the camera to the door is about 10 m. The true height of the door is 206.1 cm, while the height that we obtained from our camera was 203.6 cm, which is an error of about 1.2%.



**Figure 4:** (a) The door and (b) the person whose height has been estimated from the PTZ camera calibrated using procedure described in this paper. The person is standing at point 6 shown at Figure 1. The points on the person and the door have been manually selected.

The person's height was measured at several positions in the room (4,5,6 and 7 in Figure 1), and the results obtained are shown in Table, along with the ground truth:

Ground Truth	Position			
	4	5	6	7
175.9	172.2	<b>181.6</b>	179.2	175.1

**Table 2:** Measurements of the person height when the person is standing at various locations in the room.

As we can see from Table 2, the height is determined within 3.3% error, which is quite acceptable for surveillance applications. The height error has several causes: an external calibration error, an internal calibration error and an image error (*i.e* the error in determining exact pixel coordinates of desired points in the image). Since we determined pixel positions of the door and the person manually, the image error is small, and does not have significant effect on the height estimate. As we can notice from Table 1, the estimation of the height varies with the camera pan and tilt settings (positions 4,5,6 and 7 correspond to different camera settings). This leads us to the conclusion, that the external calibration error, *i.e* tilt bias error, contributes more to the height error than the internal calibration error. It can be seen that the height error is lowest

around the point 7. It may be explained by the fact that we used points 1 and 2 to compute tilt bias, so the tilt bias at point 7 is more precise, than the tilt bias at other points.

## 6. Conclusion

In this paper we have presented an algorithm for the calibration of a PTZ camera. For external camera calibration (estimation of camera position and orientation), the user has to point the camera to at least three points having a similar height as the camera, and at least three points on the floor. The algorithm then automatically determines the position of the camera, as well as the pan and tilt biases. Since this pointing is not very precise (there is almost certainly error, of an order of  $1^\circ$ ), we might expect similar errors in the estimation of camera position and orientation. The errors are typically small, and do not affect the performance and functionality of the visual surveillance system significantly. The algorithm for the internal camera calibration is very simple and efficient, and requires only one point correspondence at a time. The procedure is user-friendly, the only requirement being that the user has to point the camera at a texture-rich area. Experimental results suggest that it has a very good performance.

## 7. References

- [1] L. de Agapito, R. Hartley and E. Hayman, Linear self-calibration of a rotating and zooming camera, *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 15–21, 1999.
- [2] A. Basu, "Active calibration of cameras: theory and implementation", *IEEE Trans. on PAMI*, vol. 25, no. 2, pp. 256–265, 1995.
- [3] J. Batista, P. Peixoto and H. Araújo, "Real time active visual surveillance by integrating peripheral motion detection", *IEEE Workshop on Visual Surveillance*, pp. 18-25, 1998.
- [4] L. Dron, "Dynamic camera self-calibration from controlled motion sequences", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp.501–506, 1993.
- [5] F. Du and M. Brady, "Self-calibration of the intrinsic parameters of cameras for active vision systems", *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 477–482, 1993.
- [6] R. Hartley, "Self-calibration of stationary cameras", *Intl. Journal Comp. Vision*, vol. 22, pp.5-23, 1997.
- [7] S. Maybank and O. Faugeras. "A theory of a self-calibration of a moving camera". *Intl Journal of Computer Vision*, 8(2):123-152, 1992.
- [8] M. X. Li and J.-M. Lavest, "Some Aspects of Zoom-Lens Camera Calibration", *IEEE Trans. on PAMI*, pp.1105-1110, Nov., 1996.