Auto Camera Calibration Method for Person Tracking Applications

Bruno Merven, Fred Nicolls, and Gerhard de Jager

Department of Electrical Engineering, University of Cape Town
Private Bag, Rondebosch, 7701, South Africa
{bruno,nicolls,gdj}@dip.ee.uct.ac.za

Abstract. Jones [1] recently presented a novel calibration procedure that uses a linearized model of the projection of the height of a person to recover the image-plane to local-ground- plane transformation with minimum expert intervention. In this paper we present an addition to the work of Jones [1], by further reducing the measurements needed to be made manually to define the image-plane to ground-plane homography. We also do an analysis of the sensitivity of the recovered transformation to errors in measurements and noise, and point out how to use the method to achieve best calibration results.

Keywords: Auto camera calibration, person tracking

1 Introduction

A good camera model for image-plane to ground-plane transformations is a useful thing to have when tracking people or moving objects within monitored scenes. It allows the tracking to take place in world coordinates rather than in image coordinates and this has several advantages:

1. Motion models are easier to construct in world coordinates;
2. Constraints on motion are more easily imposed;
3. Occlusions become easier to ressolve;
4. The definition of a common coordinate system in the case of multi-camera tracking configurations is made simpler.

The camera model as described by Tsai [2] is based on the pin hole model of perspective projection. This model is defined by a set of intrinsic parameters and extrinsic parameters.

Intrinsic parameters are the internal properties of the camera that describe how the camera forms an image. Tsai [2] includes the following internal parameters: *focal length, pixel dimensions, distortion coefficients and the pixel coordinates of the optical center.*

Extrinsic parameters describe the camera's pose: orientation and position in the world coordinate frame. The external parameters are simply the *rotation*

J. Bigun and T. Gustavsson (Eds.): SCIA 2003, LNCS 2749, pp. 91–100, 2003.

angles and the *translation components* for the transformation between world and camera coordinate frames.

Auto calibration methods aim to obtain those camera parameters without the need for manual time-consuming, labour intensive and skill-dependent procedures based on variations of the Tsai method [2, 3].

The method proposed by Jones [1] to recover the image to the local-ground-plane transformations uses a linear model of the projected height of objects in the scene in conjunction with world knowledge about the average person's height (assuming the person is standing) and the height of the camera above the ground. The camera parameters that are estimated using this method are the pixel width to focal length ratio in the y-direction and the look-down angle of the camera.

In section 2 we describe the camera model used by Jones. We describe how the linearized height model relates to the actual projected height model, and how it can be used to estimate the camera parameters. In section 3 we present our improvement on Jones' method, that allows a more robust estimation of the camera pose and does not require the prior knowledge of the camera's height. In section 4 we investigate the sensitivity of our method to noise and distortion and explain the conditions that have to be met for the method to be suitable for a given camera calibration problem.

2 The Auto Calibration Method

2.1 The Camera Model

The camera model used in this paper is a slightly simplified version of the one used by Tsai in that the optical center is assumed to be the center of the image and radial distortions and other camera aberrations are not taken into consideration. Establishing camera to ground homography requires that the ground plane coordinate system (GPCS) origin be chosen relative to the camera position and orientation, which defines the image plane coordinate system (IPCS). Figure 1 shows the relationship between the GPCS, the IPCS and the image plane. The GPCS has origin O_G and is defined as follows:

- The Y-axis $\hat{\mathrm{Y}}$, is the projection of the optical axis along the ground plane.
- The X-axis $\hat{\mathrm{X}}$, is the vector within the ground plane normal to the camera optical axis.
- The Z-axis $\hat{\mathrm{Z}}$, is the normal to the ground plane.

The position of the focal point, O_I is directly above the GCPS origin O_G at (0,0,L).

The IPCS has O_I as origin and is defined as follows:

- The x-axis $\hat{\mathrm{x}}$ is parallel to $\hat{\mathrm{X}}$ a distance L away.
- The y-axis $\hat{\mathrm{y}}$ is perpendicular to the optical axis.
- The z-axis $\hat{\mathrm{z}}$, coincides with the optical axis. It makes an angle θ with $\hat{\mathrm{Z}}$, called the look-down angle of the camera.

A point $\mathbf{x}$ (x, y, z) in the IPCS is mapped to $\mathbf{X}$ (X, Y, Z) in the GPCS by the transformation ($\mathbf{R}$, $\mathbf{t}$), where $\mathbf{R}$ is a rotation θ about $\hat{\mathrm{x}}$ and $\mathbf{t}$ a translation, in this case $[0; 0; L]^T$.

The image plane is parallel to the plane defined by $\hat{\mathrm{x}}$ and $\hat{\mathrm{y}}$ and is located a distance f (focal length of optical system for the camera) from it.

A point P on the image plane has coordinates: $\boldsymbol{x}_P = (x, y, -f)$. The pixel coordinate system (PCS) lies in the image plane and has its origin at the top left corner of the image, shown by O_P and is defined as follows:

- i is the row position of a pixel.
- j is the column position of a pixel.
- (i_0, j_0) is the optical center.

The PCS is related to the IPCS by: $x = \alpha_x(j - j_0)$ and $y = \alpha_y(i - i_0)$ where α_x and α_y are the horizontal and vertical inter-pixel widths. Thus

$$\mathbf{x}_P = (\alpha_x^f(j_0 - j), \alpha_y^f(i_0 - i), -1)f \tag{1}$$

where $\alpha_x^f = \alpha_x/f$ and $\alpha_y^f = \alpha_y/f$. Let l be the optical ray containing O and P. l can be described by: $\mathbf{x} = \mu\mathbf{x}_P$ where μ is a scaling factor. And let P be the image of Q, an object on the ground plane. Q is then the intersection of l with the ground plane.

To find Q in the GPCS, l must be mapped from the IPCS to the GPCS. l is then described by: $\mathbf{X} = \mu\mathbf{R}\mathbf{x}_P + \mathbf{t}$. $\mathbf{X}_Q$, the GPCS coordinates of Q in terms of i and j is found by solving:

$$\begin{bmatrix} X' \\ Y' \\ 0 \end{bmatrix} = \mu' \begin{bmatrix} \alpha_x^f(j_0 - j) \\ \alpha_y^f(i_0 - i)\cos\theta + \sin\theta \\ \alpha_y^f(i_0 - i)\sin\theta - \cos\theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ L \end{bmatrix} \tag{2}$$

[Note that the constant f in (1) is absorbed by μ']. This yields:

$$X' = \frac{L\alpha_x^f(j - j_0)}{\cos\theta - \alpha_y^f(i_0 - i)\sin\theta} \tag{3}$$

$$Y' = \frac{L(\sin\theta + \alpha_y^f(i_0 - i)\cos\theta}{\cos\theta - \alpha_y^f(i_0 - i)\sin\theta} \tag{4}$$

2.2 The Height Model

In this section we show how the projected height model can be used to infer depth. The GPCS coordinate of a person's head is the point of intersection of l with the plane parallel to the ground plane at a height $(L - H)$ below the camera, where H is the height of the person. Let the image coordinates of the feet and head of the person be (i_1, j_1) and (i_2, j_2) respectively. The image of the head appears h pixels 'above' that of the feet, so:

$$i_2 = h + i_1 \tag{5}$$

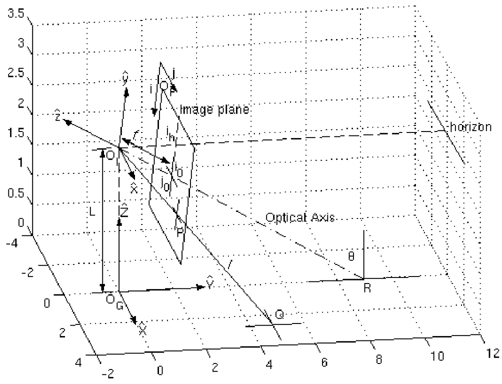


Fig. 1. The Camera, IPCS and GPCS

Let the ground-plane coordinates be (X'_1, Y'_1) and (X'_2, Y'_2). Then using (4) we get:

$$Y'_1 = \frac{L(\sin\theta + \alpha_y^f(i_0 - i_1)\cos\theta)}{\cos\theta - \alpha_y^f(i_0 - i_1)\sin\theta} \tag{6}$$

$$Y'_2 = \frac{(L - H)(\sin\theta + \alpha_y^f(i_0 - i_2)\cos\theta)}{\cos\theta - \alpha_y^f(i_0 - i_2)\sin\theta} \tag{7}$$

Since the head is above the feet in the GPCS we can let

$$Y'_1 = Y'_2 \tag{8}$$

and by substituting (5) in (8) and simplifying we get:

$$\alpha_y^f\, h = \frac{\cos\theta\sin\theta(1 - (\alpha_y^f(i_0 - i_1))^2) + \alpha_y^f(i_0 - i_1)(\cos^2\theta - \sin^2\theta)}{L/H + \alpha_y^f(i_0 - i_1)\cos\theta\sin\theta - \cos^2\theta} \tag{9}$$

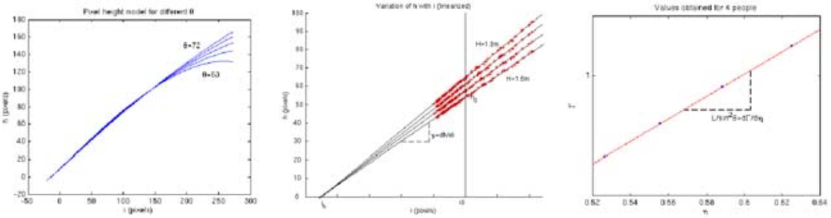
This gives us a projected height model that computes the height in pixels of the image of a person at image coordinate i_1, given that we know his real height H, the height of the camera above the ground L, the angle the camera makes with the vertical θ, and the pixel width to focal length ratio α_y^f.

2.3 The Linearized Height Model

Jones [1] goes on from here to assume the above relationship is linear, with some precautions about steep camera angles. Figure 2(a) shows us that indeed the relationship deviates more and more from linearity as θ is decreased. In section 4 we give a more quantitative analysis of the validity of this linearization.
This linear relationship is expressed as follows:

$$h = \gamma(i - i_h) \tag{10}$$

where γ is called the *height expansion rate* and i_h is the pixel row coordinate of the horizon. By recording how h varies with i over a number of frames, the values of γ and i_h can be recovered.



(a) h vs i for different θ.

(b) Measurements of (i,h) for 4 people of different heights.

(c) Calculated (η,Γ) for the 4 people.

Fig. 2.

2.4 Using the Linear Model to Perform Calibration

For a person of height H standing on the ground plane at point R (see figure 1), where the projection of the optical axis intersects the ground plane, $i_1 = i_0$ and using (5), $i_2 = h_0$.

Substituting in (9) simplifies to:

$$h_0 = \frac{H \cos\theta \sin\theta}{\alpha_y^f (L - H\cos^2\theta)} \tag{11}$$

h_0 is also found using the linearized model (10):

$$h_0 = \gamma(i_0 - i_h)\,. \tag{12}$$

The look-down angle θ is directly related to the horizon parameter i_h by:

$$(i_0 - i_h) = \frac{\cot\theta}{\alpha_y^f} \tag{13}$$

Substituting $(i_0 - i_h)$ from (13) in (12), equating to (11) and simplifying yields:

$$\sin^2\theta = \frac{\gamma(L - H)}{H(1-\gamma)} \tag{14}$$

Once a suitable number of (i,h) measurements are recorded, γ and i_h can be obtained. θ can be then be calculated using (14) given H and L. Once θ is known, α_y^f is calculated using (13).

3 Estimating the Camera Height

What was observed when trying to estimate the height of the camera for a set of observations of a person of known height H, was that there are several possible solutions of θ and L that fit the height model. In other words the problem is somewhat underconstrained. To impose enough constraint on the problem, one has to simply "watch" more than one person. For each person of different height H_p, collect data points and work out a different γ_p as shown in figure 2(b). If one defines

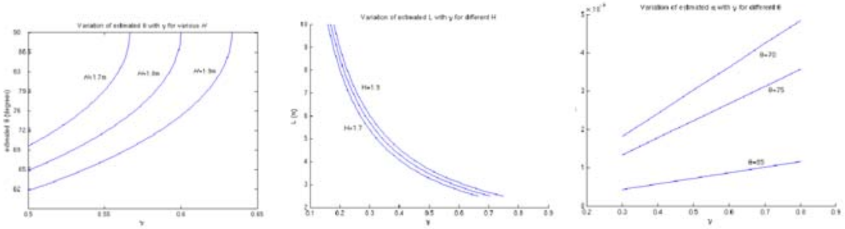
$$\Gamma_p = \frac{1-\gamma_p}{\gamma_p} \tag{15}$$

$$\eta_p = \frac{h_{0ref}}{H_{ref} \cdot h_{0p}} \tag{16}$$

where H_{ref} and h_{0ref} is the known height in m and corresponding height in pixels of one of the persons in the scene, equation (14) can bere-written as:

$$\Gamma = \frac{L}{\sin^2\theta} \cdot \eta - \frac{1}{\sin^2\theta} \tag{17}$$

By applying linear regression to the set η, Γ (see figure 2(c)), we can calculate θ and L.



(a) Sensitivity of θ to errors in γ and H_p

(b) Sensitivity of L to errors

(c) Sensitivity of α_y^f to errors

Fig. 3. Sensitivity to errors

4 Accuracy of Method and Sensitivity Analysis

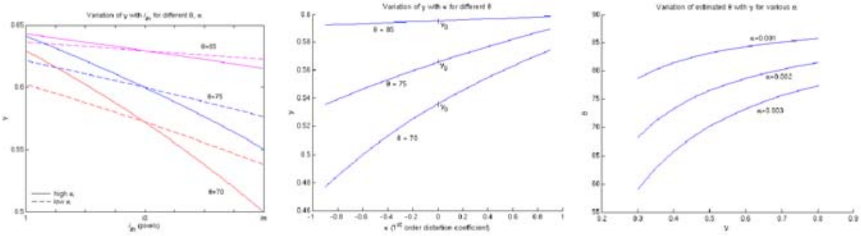
A calibration method is only useful if it yields reasonably accurate results. In this section we describe how to use our method successfully and what results should be expected given a calibration problem. We also make a few suggestions about how to get around some of the limitations of the method.

The camera parameters that the method estimates are θ, L, and α_y^f by observing how the height in pixels h_{calib} of people of height H_p, vary with the vertical pixel position i_{calib}, where $H_p = \frac{1}{\eta_p}$.

4.1 Sensitivity of Estimated θ to Errors

θ is calculated from γ and H_p using (17) or (14). If you look closely at the formulation for finding θ in (14) you will notice that $\sin^2\theta$ is quite sensitive to errors in γ. Figure 3(a) gives you a good idea of how sensitive it is to γ and H_p. The calibration method uses a linearized height model and as shown in the previous section, the actual height model deviates from linearity as θ decreases. This induces an error in i_h that gets worse as θ decreases. Although this error is small it is enough to affect $\theta_{estimated}$ significantly. We found that it is simpler to compensate for the error directly on θ than on i_h. The compensation factor for θ was found by comparing $\theta_{estimated}$ with θ_{actual}:

$$\theta_{actual} = 0.65\,\theta_{estimated} + 27(degrees) \tag{18}$$

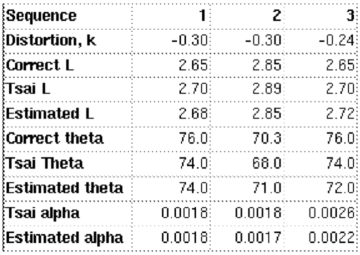


(a)

(b)

(c)

Fig. 4. Sensitivity of γ

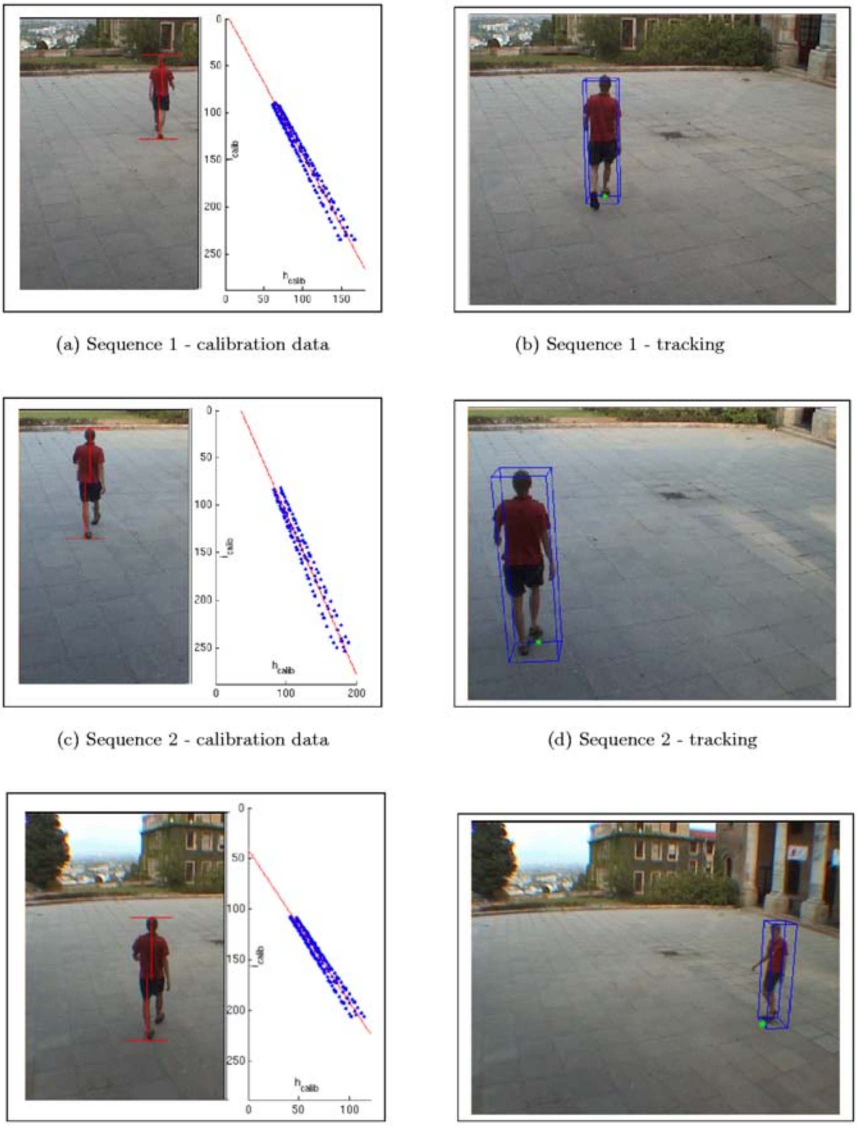


Sequence	1	2	3
Distortion, k	-0.30	-0.30	-0.24
Correct L	2.65	2.85	2.65
Tsai L	2.70	2.89	2.70
Estimated L	2.68	2.85	2.72
Correct theta	76.0	70.3	76.0
Tsai Theta	74.0	68.0	74.0
Estimated theta	74.0	71.0	72.0
Tsai alpha	0.0018	0.0018	0.0026
Estimated alpha	0.0018	0.0017	0.0022

Fig. 5. Calibration results

4.2 Sensitivity of Estimated L to Errors

L is also calculated from γ and H_p using (17). L is however less sensitive to errors in γ as seen in figure 3(b).



(a) Sequence 1 - calibration data

(b) Sequence 1 - tracking

(c) Sequence 2 - calibration data

(d) Sequence 2 - tracking

(e) Sequence 3 - calibration data

(f) Sequence 3 - tracking

Fig. 6. For each of the 3 sequences, the figure on the left shows the process of collecting calibration data (h, i). The figure on the right shows a simple 3D tracker in action that uses the calibration results for that configuration. A box of the same size, position and orientation (guessed from 2 consecutive images) as the person being tracked is generated in world coordinates and is mapped back to the image using those same calibration results.

4.3 Sensitivity of Estimated α_y^f to Errors

α_y^f is estimated from θ and i_h using (13). i_h would not typically be part of the calibration set. It is estimated from the linear model.

$$i_h = i_0 - \frac{h_0}{\gamma} \tag{19}$$

Figure 3(c) shows how α_y^f varies with γ for different θ.

4.4 γ, the Linearized Model Paramter

As seen above, all the estimated calibration parameters are calculated using γ, so it is very important that a good value of γ is obtained. Errors in γ are induced by the following factors:

1. *Not using a good calibration set.*
   The height model is linearized about i_0, (actually $\gamma = \frac{\mathrm{d}h}{\mathrm{d}i}|_{i=i_0}$), so the $mean(i_{calib})$ must be as close to i_0 as possible even if it means disguarding some calibration points. This constraint on the calibration set can easily be automated. Figure 4(a) shows how γ varies with $mean(i_{calib})$. Note that for smaller values of α_y^f, γ is less sensitive to errors. Hence, the higher α_y^f (or the lower the focal length), the more difficult it is to get accurate results.
2. *Lens Distortion effects.*
   Figure 4(b) shows how γ will be shifted from its correct value for different 1^{st} order radial distortion coefficients κ. It should be quite clear that the method will fail to give good results for cameras with high distortion if no compensation is made.
3. *Segmentation errors.*
   The calibration set is obtained typically by using dimensions and positions of bounding boxes around people-like moving blobs, which requires some form or another of segmentation. Random segmentation errors will be averaged out by using a large calibration set. However, segmentation errors, such as shadows, heads or feet 'chopped off', systematically occuring in certain parts of the image will bias γ. Care must be taken that no such errors occur during the calibration process.

4.5 Sensitivity for Extrinsic Parameters Only

The limitations of the method are made quite clear above. There are certain cases where the method just cannot practically be used to estimate both intrinsic and extrinsic parameters. However, the same linear model can be used, much more reliably, to estimate automatically the extrinsic parameters θ and L if intrinsic parameters are obtained prior to the installation of the camera.

With κ known, (i_{calib}, h_{calib}) can be compensated for to yield better γ. With α_y^f known, θ can be calculated from i_h using (13), and L using (14). Figure 4(c) show θ's sensitivity to γ. It is very clear that θ is much less sensitive in this case.

5 Results

In this section the method is evaluated. The test installations involve three different camera setups. As described in section 2 the projected height model for each camera setup is recovered by observing the variation of h_{calib} with i_{calib}.

Figure 5 shows comparisons of our methods and another method based on the traditional Tsai [2] and ground truth data. Those results show that for the first two setups, where α_y^f is small our method performs relatively well for finding both intrinsic and extrinsic parameters. However for the third setup where α_y^f is greater, the method yields less accurate results.

6 Conclusions

In this paper we have presented an automatic calibration system for fixed camera person tracking applications. When possible the method can automatically recover instrinsic parameter α_y^f and extrinsic parameters L and θ. We have also presented an extensive study of how sensitive the estimations are to errors, thus detailing some of the limitations of the calibration method. The results that are included show that provided conditions spelt out in section 4 are met, the method will yield results comparable to the classical calibration methods. Further work would entail the automatic estimation of the distortion parameter κ and possibly α_x^f, which in this paper was assumed to be equal to α_y^f.

References

[1] G.A. Jones, J. Renno and P. Remagnino. “Auto-calibration in Multiple-Camera Surveillance Environments”. *Proceedings 3rd IEEE Int. Workshop on PETS*, Copenhagen, June 1 2002

[2] Roger Y. Tsai. “A versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses”. *IEEE Journal of Robotics and Automation,* RA-3(4):323–344, August 1987

[3] Zhengyou Zhang. “A Flexible New Technique for Camera Calibration.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 22, No. 11, 1330–1334. November 2000