Refining PTZ Camera Calibration

Imran N. Junejo*
IRISA/INRIA Rennes, France
imran.junejo@inria.fr

Hassan Foroosh
University of Central Florida, Orlando, U.S.A.
foroosh@cs.ucf.edu

Abstract

Due to the increased need for security and surveillance, PTZ cameras are now being widely used in many domains. Therefore, it is very important for the applications like video mosaic generation or automatic surveillance that these camera be accurately calibrated. In this paper, we address the problem of parameter refinement for such pan-tilt-zoom (PTZ) cameras. Use of bundle-adjustment for parameter refinement has widely been adopted in the computer vision field. However, as has been shown by researchers, in presence of noise, this Maximum Likelihood estimate looses its optimality. We propose a novel statistically optimal error function that is shown to experimentally outperform this ML estimate in presence of significant noise. We perform tests on synthetic as well as on real data to verify our method.

1 Introduction

PTZ cameras are now common tools with far reaching applications [6]. Earlier work on cameras of special motion was done by [8, 5]. [3] use known rotations to perform camera calibration of the rotating cameras and [10] use the inter-image homographies. However, the state of the art auto-calibration method for rotating and zooming cameras is that of Agapito et al. [1], who use the mapping of the image of the absolute conic (IAC) via the infinite homography to impose linear constraints on camera internal parameters using five or more images.

A typical method for estimating parameters for a PTZ camera involves estimating the homography $\mathbf{H}$ between the different views, which requires point correspondences. However, due to poor camera quality or tracking errors, there is some unwanted noise in estimating the correct corresponding points. As a consequence, the estimated parameters, in most of the methods described above, are used as initialization for an another step - generally known in the community as the bundle-adjustment [11]. The process of bundle-adjustment is basically an implementation of Maximum Likelihood Estimation (MLE). It has been shown that in the presence of noise, this usage of MLE looses its asymptotic optimality and poses potential problems [9].

In this work, we derive a statistically optimal geometric error function for refining the estimated camera parameters, that is similar in its role to epipolar constraint [13]. This error function is specific to the PTZ cameras, and although we do not provide any rigorous mathematical proof for its optimality, it is experimentally shown to out perform the MLE based bundle-adjustment as used by [1]. Experimental results demonstrate the superiority of the proposed geometric error in terms of accuracy of results and noise resilience. The rest of the paper is organized accordingly.

2 Classical Bundle-Adjustment

A 3D point $\mathbf{M} = [X\ Y\ Z]^T$ is projected by a PTZ camera to a point $\mathbf{m}$ on the image plane by a 3×3 matrix $\mathbf{P}$ as:

$$\mathbf{m} \sim \underbrace{\mathbf{K}\mathbf{R}}_{\mathbf{P}}\mathbf{M}, \quad \mathbf{K} = \begin{bmatrix} \lambda f & \gamma & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad (1)$$

where $\sim$ indicates equality up to multiplication by a non-zero scale factor, $\mathbf{R}$ is the rotation matrix, and $\mathbf{K}$ is a nonsingular 3×3 upper triangular matrix known as the camera calibration matrix including five parameters, i.e. the focal length f, the skew γ, the aspect ratio λ and the principal point at $(u_0,\ v_0)$.

Most practical auto-calibration methods comprise of two steps [4, 6, 1]: In the *first* step an initial solution is found by solving directly a set of algebraic constraints that are often linear. In the *second* step, the initial solution is refined by minimizing an error function, which preferably should reflect the geometry of the configuration [4, 6]. For pan-tilt-zoom (PTZ) cameras, an error functions based on minimizing geometric distances in the image plane is sought that usually involve minimizing the error between the measured and the estimated

*This is part of the author's work at the University of Central Florida, Orlando, U.S.A.

978-1-4244-2175-6/08/$25.00 ©2008 IEEE

reprojected image coordinates. Thus we seek a Maximum Likelihood (ML) solution assuming that the error in the measurement is Gaussian. This is generally referred to as the bundle adjustment [1, 10, 11]. Given n images and m corresponding points, the maximum likelihood estimate can be obtained by minimizing the following Euclidean distance error function:

$$\mathcal{C}_{ml} = \sum_{i=1}^{n}\sum_{j=1}^{m} \parallel \hat{\mathbf{x}}_{ij} - \mathbf{K}_i\mathbf{R}_i\bar{\mathbf{X}}_j \parallel^2 \quad (2)$$

Thus the squared error sum between the image measurement ($\hat{\mathbf{x}}_{ij}$) and the projection of the true image points for all points across all views is minimized. Minimizing (2) is a non-linear problem, which is solved by Levenberg-Marquardt iterative minimization method [6]. The optimal (ML) solution lies close to the initial solution. Thus it aims to change (or perturb) the estimated points and the camera parameters such that the cost function is minimized subject to the reprojection model defined by the homography relationship between the views. Therefore the probability of a true solution will follow a normal distribution. Formally, the measured location $\hat{\mathbf{x}}$ is related to the true location by a Gaussian additive noise η:

$$\hat{\mathbf{x}} = \mathbf{x} + \eta = \mathcal{F}(\mathbf{K}, \mathbf{R}) + \eta \quad (3)$$

where $\mathcal{F}(\mathbf{K}, \mathbf{R})$ is the reprojection model for the true values of the image points given an estimate of the parameters $\mathbf{K}$ and $\mathbf{R}$. Therefore the probability of the true solution is:

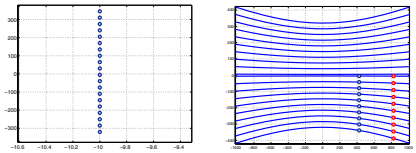
$$p(\hat{\mathbf{x}}|\mathbf{K}, \mathbf{R}, \sigma) = \mathcal{N}(\hat{\mathbf{x}}|\mathcal{F}(\mathbf{K}, \mathbf{R}), \sigma) \quad (4)$$

which one aims to maximize [1, 6].

3 Geometric Error Function

In most cases, the data is corrupted by noise. This is mainly due to the poor matching, changing lighting conditions or other tracking errors. As a consequence, in addition to the parameters that we are estimating, other hidden unknowns are involved in the problem. These unknown parameters are called as the *nuisance parameters*. It is generally believed in the vision community that bundle-adjustment, which is basically an implementation of MLE, is "optimal". However, as has been shown by Okatani and Deguchi [9], this is not the case and naive implementation of MLE for minimization of reprojection errors or other vision applications poses problems. In the situation when there is no noise or no nuisance parameters, and provided a sufficient number of data are give, MLE is theoretically guaranteed to provide an asymptotically optimal estimate. However, the nuisance parameters increase with the amount of data and MLE looses it optimality.

In contrast, we *propose* an error function that is shown to be experimentally optimized and consistently give better results than MLE. By *optimized* we mean a



$\mathbf{I}_1$ $\mathbf{I}_2$

Figure 1. Points in $\mathbf{I}_2$ corresponding to image points $\mathbf{x}_i$ in $\mathbf{I}_1$ lie on a conic in $\mathbf{I}_2$.

cost function tailored specifically to our special camera model i.e. pure rotation and zoom.

Pure Pan: For a panning PTZ camera, a point $\mathbf{x}$ in the first image $\mathbf{I}_1$ is related to the corresponding point $\mathbf{x}'$ in the second image $\mathbf{I}_2$ via the infinite homography:

$$\mathbf{x}' \sim \mathbf{K}_2\mathbf{R}_y\mathbf{K}_1^{-1}\mathbf{x} \quad (5)$$

where the rotation matrix $\mathbf{R}_y$ is parameterized as $\mathbf{R}_y = \begin{bmatrix} c & 0 & -s \\ 0 & 1 & 0 \\ s & 0 & c \end{bmatrix}$ where $c = \cos\theta_y$ and $s = \sin\theta_y$. Using the first two linear constraints given by

$$\mathbf{x}' \times (\mathbf{K}_2\mathbf{R}_y\mathbf{K}_1^{-1}\mathbf{x}) = \mathbf{0} \quad (6)$$

we then express c and s in terms of $\mathbf{K}_i$ and the feature points $\mathbf{x}$ and $\mathbf{x}'$. Upon substituting c and s into the Pythagorean identity

$$c^2 + s^2 - 1 = 0 \quad (7)$$

and rearranging, we get:

$$\mathbf{x}'^T\mathbf{Q}\mathbf{x}' = 0 \quad (8)$$

Note that the above equation is independent of the rotation angle. $\mathbf{Q}$ is a conic given by the 3×3 symmetric matrix,

$$\mathbf{Q} = \begin{bmatrix} a & b/2 & d/2 \\ b/2 & c & e/2 \\ d/2 & e/2 & f \end{bmatrix} \quad (9)$$

$$\begin{aligned}
\text{with } a &= (\mathbf{x}_y - v_0)^2 && (10)\\
b &= 0 && (11)\\
c &= -f_1^2 - (\mathbf{x}_x - u_0)^2 && (12)\\
d &= (4u_0v_0\mathbf{x}_y - 2u_0\mathbf{x}_y^2 - 2u_0v_0^2) && (13)\\
e &= (2v_0\mathbf{x}_x^2 - 4\mathbf{x}_x v_0 u_0 + 2v_0u_0^2 + 2v_0f_1^2) && (14)\\
f &= u_0^2\mathbf{x}_y^2 - 2v_0u_0^2\mathbf{x}_y + f_2^2v_0^2 - f_1^2v_0^2 - 2f_2^2v_0\mathbf{x}_y \\
&\quad + f_2^2\mathbf{x}_y^2 + 2v_0^2u_0\mathbf{x}_x - v_0^2\mathbf{x}_x^2 && (15)
\end{aligned}$$

where f_1 and f_2 are the camera focal lengths in views $\mathbf{I}_1$ and $\mathbf{I}_2$, respectively. The conic $\mathbf{Q}$, in addition to the camera parameters, is parameterized by the image point $\mathbf{x} = \begin{bmatrix} \mathbf{x}_x & \mathbf{x}_y & 1 \end{bmatrix}^T$. What equation (8) implies is that for every point $\mathbf{x}$ in $\mathbf{I}_1$, the corresponding point $\mathbf{x}'$ in $\mathbf{I}_2$ must lie on the conic $\mathbf{Q}$, which is defined by the camera parameters and the point $\mathbf{x}$. Similarly, for transformation from $\mathbf{I}_2$ to $\mathbf{I}_1$, it can be shown that for every point $\mathbf{x}'$ in $\mathbf{I}_2$, the corresponding point $\mathbf{x}$ in $\mathbf{I}_1$ must lie on a conic $\mathbf{Q}'$:

$$\mathbf{x}^T\mathbf{Q}'\mathbf{x} = 0 \quad (16)$$

where $\mathbf{Q}'$, in contrast to $\mathbf{Q}$, is defined by the camera parameters and the point $\mathbf{x}' = \begin{bmatrix} \mathbf{x}'_x & \mathbf{x}'_y & 1 \end{bmatrix}^T$ in $\mathbf{I}_2$.

In summary, as a camera pans the points in the image plane trace a conic trajectory. It can be readily verified from (10)-(12) that these conics are in fact hyperbolas. This is demonstrated in Figure 1. Points corresponding to $\mathbf{x}_i$ in view $\mathbf{I}_1$ lie on a hyperbolic trajectory in $\mathbf{I}_2$. Exactly where a corresponding point lies on the hyperbola depends on the rotation angle. As shown in the Figure 1(b), the blue dots are the corresponding points when the pan angle was $\theta_y = 20°$ whereas it was $\theta_y = 35°$ for the red dots. Therefore, instead of looking for the solution in the neighborhood of a points in *all* directions, we can minimize the orthogonal distance of points from the hyperbolic curves.

Derivation of the Cost function: Similar to the mapping of points to lines between two views by the Fundamental matrix, from the above discussion, a PTZ camera, undergoing pan motion defines quadratic curves for mapping of the corresponding image points $\mathbf{x} \leftrightarrow \mathbf{x}'$. Thus, instead of minimizing the distance of feature points to epipolar lines [13] (or finding points consistent with the homographies [2]), for pure rotation we can minimize the distance of points to conics.

The geometric distance $\mathcal{D}$ of a point $\mathbf{x}$ to a conic $\mathbf{Q}'$ can be obtained using Sampson's rule [6]

$$\mathcal{D} = \epsilon^T (\mathbf{J}\mathbf{J}^T)^{-1} \epsilon \tag{17}$$

where $\epsilon = \mathbf{x}^T \mathbf{Q}' \mathbf{x}$ is the cost associated with $\mathbf{x}$ and $\mathbf{J} = \left[\frac{\partial(\mathbf{x}^T\mathbf{Q}'\mathbf{x})}{\partial \mathbf{x}_x}, \frac{\partial(\mathbf{x}^T\mathbf{Q}'\mathbf{x})}{\partial \mathbf{x}_y} \right]$ is a matrix of partial derivatives. Using the chain rule, the elements of $\mathbf{J}$ are computed as:

$$\frac{\partial(\mathbf{x}^T\mathbf{Q}'\mathbf{x})}{\partial \mathbf{x}_x} = \frac{\partial(\mathbf{x}^T\mathbf{Q}'\mathbf{x})}{\partial \mathbf{x}_x} \frac{\partial \mathbf{x}}{\partial \mathbf{x}_x} = 2(\mathbf{Q}'\mathbf{x})_1$$

and similarly

$$\frac{\partial(\mathbf{x}^T\mathbf{Q}'\mathbf{x})}{\partial \mathbf{x}_y} = 2(\mathbf{Q}'\mathbf{x})_2$$

where the subscripts 1 and 2 denote the first and the second component of the vector, respectively.

Using (17), the distance of a point $\mathbf{x}$ to a conic $\mathbf{Q}'$ thus reduces to:

$$\mathcal{D} = \frac{(\mathbf{x}^T\mathbf{Q}'\mathbf{x})^2}{4((\mathbf{Q}'\mathbf{x})_1^2 + (\mathbf{Q}'\mathbf{x})_2^2)} \tag{18}$$

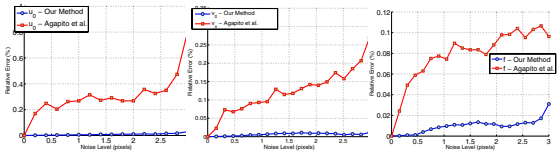
For symmetric error minimization, the cost function would then be of the form

$$\sum_{i=1}^{n} \left(\frac{(\mathbf{x}_i^T\mathbf{Q}_i'\mathbf{x}_i)^2}{4((\mathbf{Q}_i'\mathbf{x}_i)_1^2 + (\mathbf{Q}'_i\mathbf{x}_i)_2^2)} + \frac{(\mathbf{x}_i'^T\mathbf{Q}_i\mathbf{x}_i')^2}{4((\mathbf{Q}_i\mathbf{x}_i')_1^2 + (\mathbf{Q}_i\mathbf{x}_i')_2^2)} \right)$$

$$= \sum_{i=1}^{n} (\mathcal{D} + \mathcal{D}') \tag{19}$$

That is, the camera intrinsic and extrinsic parameters and the correct feature point locations must minimize the sum of distances to the conics. The minimum of this non-linear cost function is sought using the Levenberg-Marquardt algorithm. We have thus reduced the search space of true feature locations to quadratic curves.

Tilt Motion: The above discussion equally applies to pure tilt, or in fact to any single axis rotation.



(a) u_0 (b) v_0 (c) f

Figure 2. Performance vs. Noise Level: averaged over 1000 independent trials. Results after geometric optimization compared to ML-optimized Agapito et al.

Pan-Tilt Motion For a PTZ camera undergoing both pan and tilt motion, (5) is modified as:

$$\mathbf{x}' \sim \mathbf{K}_2\mathbf{R}_x\mathbf{R}_y\mathbf{K}_1^{-1}\mathbf{x} \tag{20}$$

where $\mathbf{R}_y$ is as defined above, and $\mathbf{R}_x$ defines rotation around the x-axis by θ_x. In principle, there are sufficient number of constraints to eliminate the two angles. However, due to non-linearity, this is not straightforward. Therefore, we parameterize $\mathbf{R}_y$ as before in terms of c and s, and also parameterize $\mathbf{R}_x$ by $c' = \cos\theta_x$ and $s' = \sin\theta_x$. Similar to pan case, we then express c and s in terms of feature points and the camera parameters to obtain a conic as defined in (8). The difference now is that the conic $\mathbf{Q}$ (and similarly $\mathbf{Q}'$) contains the tilt angle components c' and s', which are used as additional parameters in the cost function (19).

4 Experiments

Due to space limitations, we show results on Pan motion (tilt follows similarly) and pan-tilt case only.

Synthetic Data: We performed over 1000 independent trials. For this purpose, a point cloud of 1000 random points [1] was produced inside a unit cube to generate image point correspondences while arbitrarily selecting the rotation angles. Simulated camera has a focal length of 1000, aspect ratio of $\lambda = 1.5$, skew $\gamma = 0$, and the principal point at $(u_0, v_0) = (512, 384)$, for image size of 1024×768.

We used the method of Agapito et al. [1] to estimate camera parameters and compare our results with the ML estimate method proposed by [1]; our refinement approach consistently outperforms the classical ML refinement.

Pan Motion: For pure pan (or pure tilt), which is generally known to be a degenerate case of camera motion [1, 6], the results are shown in Figure 2. Figure 2(a) shows the relative error in u_0, which is found to be less than 0.2% for a noise of up to 3 pixels. Similarly, noise for the v_0 and f is also very low compared to MLE. The error in the proposed estimated method is comparably lower than the classical ML estimation method.

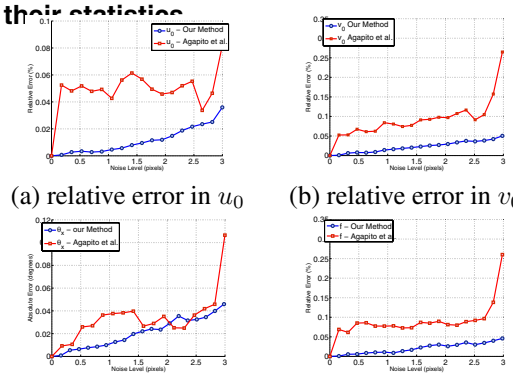
Pan-Tilt Motion: For the case when the camera is both panning and tilting, the error curves are shown in Figure 3. The error for the parameters u_0, v_0, f, and θ_x is lower than 0.04%,0.1%, 0.04% and $0.05°$, respectively.

(a)

Comb.	u_0	u_0'	v_0	v_0'	f	f'	θ_x	θ_x'
C_1	301.12	300.29	299.36	289.89	592.18	540.02	2.69	2.60
C_2	320.10	320.37	610.87	485.43	735.21	697.44	2.30	3.11
C_3	255.22	295.22	539.12	376.33	331.14	300.27	1.25	2.89
C_4	299.54	323.25	622.74	440.88	467.15	392.46	0.98	2.28
C_5	301.20	288.52	381.39	376.33	385.67	331.72	0.35	3.04
C_6	266.11	286.79	294.51	289.89	247.24	176.39	3.45	2.65
C_7	277.65	311.54	897.35	370.97	767.97	433.87	0.48	3.52
C_8	256.11	305.24	415.55	459.82	445.92	387.45	1.42	2.45
C_9	312.12	310.22	333.22	176.56	522.84	487.33	3.11	2.81
Mean	287.69	304.60	488.24	362.90	499.48	416.33	1.78	2.82
M. Dev.	20.56	10.02	121.04	83.49	125.04	92.19	0.94	0.24

(b)

Figure 4. (a) Sample images from pan-tilt sequence. (b) Estimated parameters and their statistics.



(a) relative error in u_0 (b) relative error in v_0

(c) Absolute Error in θ_x (d) Relative Error in f

Figure 3. Performance vs. Noise Level: averaged over 1000 independent trials for pan-tilt motion. Results after geometric optimization compared to ML-optimized Agapito et al.

The above experiments indicate that the minimization based on the geometric error function derived in this paper consistently give better results than the traditional ML estimate for the PTZ camera.

Real Data: The data was obtained by a SONY® SNC-RZ30N PTZ camera with an image resolution of 640×480, and deliberately keep f same between frames. Image features and correspondences are obtained by using the SIFT algorithm [7]. In order to evaluate our results, we use an approach similar to [12].

Pan-Tilt Motion: This sequence is taken while panning with $\theta_y = 2^\circ$ and tilting with $\theta_x = 2^\circ$, and keeping the focal length fixed for the camera. We apply the method to all the combinations of 9 images from the total of 10 images. The results are shown in Figure 4, where $'$ indicates results obtained from the proposed refinment. The the tilt angle θ_x was estimated as 1.78° by MLE and 2.82° by the proposed cost function. The median deviation [12] of the results obtained by the proposed cost function is smaller than that of MLE. The principal point is also estimated to be close to the center of the image. Notice that we are able to refine only one rotation angle for pan-tilt motion and no angle for the pan or tilt motion alone. This is primarily due to the fact that we remove one angle of rotation when we apply the Pythagorean identity in Eq (7) in Section 3.

References

[1] L. D. Agapito, E. Hayman, and I. Reid. Self-calibration of rotating and zooming cameras. *Int. J. Comput. Vision*, 45(2):107–127, 2001.

[2] O. Chum, T. Pajdla, and P. Sturm. The geometric error for homographies. *Computer Vision and Image Understanding*, 97(1):86–102, January 2005.

[3] J. Frahm and R. Koch. Camera calibration with known rotation. *Proc. IEEE ICCV*, pages 1418–1425, 2003.

[4] R. Hartley. Minimizing algebraic error in geometric estimation problems. *Proc. of ICCV*, pages 469–476, 1998.

[5] R. I. Hartley. Self-calibration of stationary cameras. *Int. J. Comput. Vision*, 22(1):5–23, 1997.

[6] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[7] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 6(2):91–110, 2004.

[8] T. Moons, L. Gool, M. Proesmans, and E. Pauwels. Affine reconstruction from perspective image pairs with a relative object-camera translation in between. *IEEE Trans. Pattern Anal. Mach. Intell.*, 18(1):77–83, 1996.

[9] T. Okatani and K. Deguchi. Toward a statistically optimal method for estimating geometric relations from noisy data: cases of linear relations. *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages I: 432–439, 2003.

[10] Y. Seo and K. Hong. About the self-calibration of a rotating and zooming camera: Theory and practice. *Proc. IEEE ICCV*, pages 183–189, 1999.

[11] B. Triggs, P. McLauchlan, R. I. Hartley, and A. Fitzgibbon. Bundle adjustment — a modern synthesis. *Vision Algorithms: Theory and Practice*, pages 298–373, 1999.

[12] Z. Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, 2000.

[13] Z. Zhang, R. Deriche, O. D. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1-2):87–119, 1995.