Self-Calibration from Multiple Views with a Rotating Camera *

Richard I. Hartley

G.E. CRD, Schenectady, NY, 12301.
Email : hartley@crd.ge.com

Abstract. A new practical method is given for the self-calibration of a camera. In this method, at least three images are taken from the same point in space with different orientations of the camera and calibration is computed from an analysis of point matches between the images. The method requires no knowledge of the orientations of the camera. Calibration is based on the image correspondences only. This method differs fundamentally from previous results by Maybank and Faugeras on self-calibration using the epipolar structure of image pairs. In the method of this paper, there is no epipolar structure since all images are taken from the same point in space. Since the images are all taken from the same point in space, determination of point matches is considerably easier than for images taken with a moving camera, since problems of occlusion or change of aspect or illumination do not occur. The calibration method is evaluated on several sets of synthetic and real image data.

1 Introduction

The possibility of calibrating a camera based on the identification of matching points in several views of a scene taken by the same camera has been shown by Maybank and Faugeras ([9, 5]). Using techniques of Projective Geometry they showed that each pair of views of the scene can be used to provide two quadratic equations in the five unknown parameters of the camera. For this, it is necessary that the two views be taken from different viewpoints. Given three pairs of views, a method of directly solving these equations to obtain the camera calibration has been reported in [9, 5, 8] based on homotopy continuation. It has been reported however that this method requires extreme accuracy of computation, and seems not to be suitable for routine use. The applicability of these methods is further complicated by the problem of finding matched points in images taken from different viewpoints. This task can be difficult, because of occlusion, aspect changes and lighting changes that inevitably occur when the camera moves.

Recently several other papers on self-calibration have appeared ([3, 2, 4]). These papers all rely on known motions of the cameras. In [3] the motion of the camera is assumed to be purely translational. In [2, 4] rotational motions

* The research described in this paper has been supported by DARPA Contract #MDA972-91-C-0053

Lecture Notes in Computer Science, Vol. 800
Jan-Olof Eklundh (Ed.)
Computer Vision – ECCV '94

of the camera are considered, but the rotation must be through known angles. This simplifies the calibration task enormously. In this paper, on the other hand, calibration is carried out solely on the basis of image content, and without a priori assumptions of calibration values. Calibration can be carried out by finding point matches in as few as three images, though for best results, more images may be used. The method is based on analysis of the projective distortion that an image undergoes when the camera is rotated.

The calibration algorithm is demonstrated on real and synthetic data and is shown to perform robustly in the presence of noise.

2 The Camera Model

A commonly used model for perspective cameras is that of projective mapping from 3D projective space, $\mathcal{P}^3$, to 2D projective space, $\mathcal{P}^2$. This map may be represented by a 3×4 matrix, M of rank 3. The mapping from $\mathcal{P}^3$ to $\mathcal{P}^2$ takes the point $\mathbf{x} = (x, y, z, 1)^\top$ to $\mathbf{u} = M\mathbf{x}$ in homogeneous coordinates. (Note: the equality relation when applied to homogeneous vectors really means equality up to a non-zero scale factor).

Provided the camera centre is not located on the plane at infinity, the matrix M may be decomposed as $M = K(R|-R\mathbf{t})$, where $\mathbf{t}$ represents the location of the camera, R is a rotation matrix representing the orientation of the camera with respect to an absolute coordinate frame, and K is an upper triangular matrix called the *calibration matrix* of the camera. The matrix $(R| - R\mathbf{t})$ represents a rigid transformation (rotation and translation) of R^3. Given a matrix M it is a very simple matter to obtain this decomposition, using the QR-decomposition of matrices ([1, 10]).

The entries of the matrix K may be identified with certain physically meaningful quantities known as the *internal parameters* of the camera. Indeed, K may be written as

$$K = \begin{pmatrix} k_u & s & p_u \\ 0 & k_v & p_v \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

where

- k_u and k_v are the magnifications in the two coordinate directions,
- p_u and p_v are the coordinates of the principal point,
- s is a skew parameter corresponding to a skewing of the coordinate axes.

The purpose of this paper is to give a method for determining the matrix K of internal camera parameters. In the method to be described, the camera will be held in the same location in space and rotated to different orientations. For convenience, the common location of all the cameras will chosen to be the origin of the coordinate system. We will speak of several cameras each with its own camera matrix, whereas in fact the cameras will be the same camera, with the same interior parameters, differing only in their orientation. Thus, we consider a set of cameras with camera matrices $M_j = K(R_j \mid 0)$.

A point $\mathbf{x} = (x, y, z, 1)^\top$ is mapped by the camera M_j to the point $\mathbf{u} = K(R_j \mid 0)(x, y, z, 1)^\top = KR_j(x, y, z)^\top$. In other words, since the last column of M_j is always 0, the fourth coordinate of $\mathbf{x}$ is immaterial. Therefore, in this paper, we will drop the fourth column of the camera matrix, and write instead $M_j = KR_j$ where K is upper triangular, the same for all cameras, and R_j is a rotation matrix. This transformation sends points $\mathbf{x} = (x, y, z)^\top$ to $\mathbf{u} = KR_j\mathbf{x}$. Note that the points $k\mathbf{x}$, where k is a non-zero factor, are all mapped to the same point independent of the scale factor. Consequently, M_j represents a mapping between a two-dimensional projective object space with coordinates $(x, y, z)^\top$ and two-dimensional projective image space with coordinates $(u, v, w)^\top$. This situation has a very convenient feature, not shared by the usual 3D to 2D projective mapping, namely that the mapping M_j from object to image space is invertible.

3 Rotating the Camera

Now, we will consider what happens to an image taken by a camera when the camera is rotated. Thus, let $M = KR$ and $M' = KR'$ be two cameras, and let $\mathbf{u}_i = KR\mathbf{x}_i$ and $\mathbf{u}'_i = KR'\mathbf{x}_i$. From this it follows that

$$\mathbf{u}'_i = KR'R^{-1}K^{-1}\mathbf{u}_i$$

This simple observation gives the following important result

Proposition 1. *Given a pair of images taken by cameras with the same interior parameters from the same location, then there is a projective transformation P taking one image to the other. Furthermore, P is of the form $P = KRK^{-1}$ where R is a rotation matrix and K is the calibration matrix.*

In standard terminology, the relation $P = KRK^{-1}$ may be described by saying that P is a conjugate of a rotation matrix, K being the conjugating element.

Now, suppose we have several cameras with matrices $M_j = KR_j$ for $j = 0, \ldots, N$. For convenience, we assume that the coordinate axes are chosen to be aligned with the 0-th camera, so that $R_0 = I$, the identity matrix, and hence $M_0 = K$. Write $P_j = M_j M_0^{-1} = KR_jK^{-1}$. This gives the following proposition.

Proposition 2. *Given a set of images $J_0, \ldots J_N$ taken from the same location by cameras with the same calibration (or with the same camera), then there exist 2D projective transforms, represented by matrices P_j, taking image J_0 to image J_j. The matrix P_j may be written in the form*

$$P_j = KR_jK^{-1}$$

where K is the common calibration matrix of the cameras, and R_j represents the rotation of the $j-th$ camera with respect to the 0-th. The camera matrix for the j-th camera is $M_j = KR_j = P_jK$.

4 Algorithm Idea

The idea of the calibration algorithm will now be described. Suppose we are given a set of overlapping images $J_0, J_1, \ldots, J_N$ where $N \geq 2$, all taken from the same location with cameras with the same calibration (or the same camera). It is required to determine the common calibration matrix of the cameras. The steps of the algorithm are as follows.

1. Establish point correspondences between the images.
2. For each $j = 1, \ldots, N$ compute the 2D projective transformation P_j matching J_0 to J_j. Image-to-image projective transformations may be computed from as few as four point matches.
3. Find an upper triangular matrix K such that $K^{-1}P_jK = R_j$ is a rotation matrix for all $j > 0$. The matrix K is the calibration matrix of the cameras, and R_j represents the orientation of the $j - th$ camera with respect to the 0-th camera.
4. Refine the estimated camera matrix using Levenberg-Marquardt iterative techniques ([7]).

The main subject of this paper comprises step 3 of this algorithm, which will be described in section 5.

5 Determining the Calibration Matrix

We now suppose that transformations P_j are known for $j = 1, \ldots, N$. We wish to find the calibration matrix K, which will be an upper triangular matrix satisfying the condition that $K^{-1}P_jK = R_j$ is a rotation matrix for all j. For any non-singular matrix A, let $A^{-\top}$ be the inverse transpose of A. For a rotation matrix R, we have $R = R^{-\top}$. From the relation $R_j = K^{-1}P_jK$ it follows that $R_j = K^\top {P_j}^{-\top} K^{-\top}$. Equating the two expressions for R_j gives $K^\top {P_j}^{-\top} K^{-\top} = K^{-1}P_jK$, from which it follows that

$$(KK^\top){P_j}^{-\top} = P_j(KK^\top) \tag{2}$$

Given sufficiently many views and corresponding matrices P_j equation 2 may be used to solve for the entries of the matrix $KK^\top$. In particular, denoting $KK^\top$ by C and writing

$$C = KK^\top = \begin{pmatrix} a & b & c \\ b & d & e \\ c & e & f \end{pmatrix}$$

the equation (2) gives rise to a set of nine linear equations in the six independent entries of C. It may be seen that multiplying C by a constant factor does not have any effect on the equation (2). Consequently, C can only be solved up to a constant factor. It turns out (see Section 6) that because of redundancy, the nine equations derived from (2) for a single known transformation P_j are not sufficient to solve for C. However, if two or more such P_j are known, then we may solve the overconstrained system of equations to find a least-squares solution for C.

Once $C = KK^\top$ is found it is an easy matter to solve for K using the Choleski factorization ([1, 10]). The factorization is unique, provided that K is constrained to have positive diagonal entries. A solution for K is only possible when C is positive-definite. This is guaranteed for noise-free data, since by construction, C possesses such a factorization. With noisy input data, it is possible that the matrix C turns out not to be positive-definite, and so the calibration matrix can not be found. In practice this was found to happen only in the case of gross errors in the point matching.

6 Are Two Views Sufficient ?

We consider now what can be done with only two views. Two views are related via a projective transformation $P = KRK^{-1}$. The fact that P is a conjugate of a rotation matrix has the immediate consequence that P and R have the same eigenvalues. The eigenvalues of a rotation matrix are equal to 1, $\exp(i\theta)$ and $\exp(-i\theta)$, where θ is the angle of rotation. Therefore, by finding the eigenvalues of P, we are able to find the angle of rotation of R.

Any rotation is conjugate to a rotation about the x axis. Since P is conjugate to a rotation through angle θ, it is therefore conjugate to a rotation about the x axis through angle θ, denoted R_x^θ. Thus, one may write $P = HR_x^\theta H^{-1}$, and hence $PH = HR_x$. Knowing P and R_x^θ one obtains H by solving a set of linear equations. Now, using QR decomposition, we may obtain $H = KR$, where K is upper-triangular and R is a rotation. It follows that $P = KRR_x^\theta R^{-1}K^{-1} = K\hat{R}K^{-1}$ as required.

The matrix H found by solving $PH = HR_x$ is not unique, however. In fact, if $PH = HR_x$, and $\mathrm{diag}(\alpha, 1, 1)$ is a diagonal matrix, then $PH\mathrm{diag}(\alpha, 1, 1) = HR_x\mathrm{diag}(\alpha, 1, 1) = H\mathrm{diag}(\alpha, 1, 1)R_x$, since $\mathrm{diag}(\alpha, 1, 1)$ commutes with R_x. It follows that $H\mathrm{diag}(\alpha, 1, 1)$ is an alternative solution. In short, there exists a one-parameter family of solutions for H (ignoring constant multiples), and hence for K. However, with just one constraint on the calibration matrix it is possible to determine K. Since the skew s is usually very small, the assumption that $s = 0$ is a very reasonable one, commonly used by other authors ([2]). Alternatively, one may make other assumptions about the calibration, for instance that the camera has square pixels, $k_u = k_v$. Under either of these assumption it is possible to find K from only two views. Details are deferred to another paper.

7 Experimental Verification of the Algorithm

7.1 Tests with Synthetic Data

First of all, the calibration algorithm was carried out on synthetic data to determine its performance in the presence of noise. The synthetic data was created to simulate the images taken with a 35mm camera with a 50mm lens, and digitized with 20 pixels per mm. The field of view is approximately 38° × 26°. and the image measures 700 × 460 pixels. In this case, the magnification factors k_u and k_v both equal 1000. It was also assumed that $s = 0$, and image coordinates were taken to be centred at the principal point of the image, so that $p_u = p_v = 0.0$. A set of 100 random points were chosen and their computed image coordinates in a set of views were used to calibrate the camera.

Tables 1 and 2 summarize the results obtained with three views. Experiments with a larger number of views gave more accurate results. Experiments with real images indicate that images may be matched with an RMS error of about 0.5 pixels, which suggests that this is a realistic noise level. The results with synthetic data show that the algorithms are robust for noise levels well beyond this range.

Noise	k_u	k_v	p_u	p_v	skew
–	1000.0	1000.0	0.0	0.0	0.0
0.125	999.2	999.5	-0.2	-0.3	0.0
0.25	998.4	999.0	-0.4	-0.5	0.1
0.5	996.8	998.0	-0.7	-0.9	0.1
1.0	993.5	996.0	-1.5	-1.8	0.2
2.0	956.1	960.7	-7.5	19.1	0.8
4.0	946.0	955.3	-12.4	26.4	1.5
8.0	938.7	956.6	-15.8	23.6	3.7
16.0	1077.9	1108.7	-0.2	-13.7	5.1

Table 1. Calibration from three images in the presence of various degrees of noise with one run at each noise level. The size of the images was 700 × 460 pixels. The three view directions lie in a circle of radius 10°. The first row shows the expected parameter values, whereas subsequent rows show the effects of different levels of noise (measured in pixels). The table shows the calibration results after refinement using Levenberg-Marquardt iteration. Errors before the refinement were approximately twice as large.

7.2 Tests with Real Images

Calibration tests were carried out on two sets of real images. In the first set of images five images of the Capitol building in Washington were taken with a 35mm camera with a zoom lens. The focal length of the lens was approximately 40mm (though not known exactly, since it was a zoom lens). The images were printed, enlarged and digitized. The images were then scanned at 150 pixels per inch, resulting in images of size 776 × 536 pixels. Corresponding points were found between the images using STEREOSYS ([6]) and the calibration was carried out. A composite of the five images is shown in Fig 1. The calibration results are summarized in Table 3.

Noise	statistic	k_u	k_v	p_u	p_v	skew
1.0	Mean	1016.2	1016.4	5.6	-13.0	-0.2
	σ	29.1	29.2	7.5	14.7	0.9
2.0	Mean	979.4	976.1	18.5	-1.1	-4.2
	σ	44.0	45.2	15.2	2.8	7.5

Table 2. Result of 100 runs with 3 views, with varying random noise of 1 and 2 pixels. The parameters k_u and k_v were highly correlated, whereas other parameters showed little correlation. The table shows the results after iterative refinement. However, the results before refinement were not significantly worse.

	k_u	k_v	p_u	p_v	skew	residual pixel error
unrefined	964.4	966.4	392.8	282.0	-4.9	unknown
refined	956.8	959.3	392.0	281.4	-6.4	0.33

Table 3. Calibration results for five images of the Capitol with a 35mm camera. The results before and after iterative refinement are quite similar. The calibration seems very plausible, since the measured skew is small, magnification is almost the same in both directions and the principal point is near the centre of the image. The last column gives the difference in pixels between predicted image coordinates (given the calibration and reconstruction) and the measured values. A value of k_u or k_v of 960 corresponds to a focal length of approximately $35 \times 960/776 = 43.3$mm.

Fig. 1. A composite image constructed from five different views of the Capitol. The composite image shows very clearly the projective distortion necessary for matching the images. Analysis of this projective distortion provides the basis for the calibration algorithm.

A second set of 29 images were taken covering a region of about 48 × 22 degrees with a 105mm lens in a 35mm camera. The images were of size 470 × 320 pixels. The lens has a fairly small field of view, which increases the difficulty of calibration using the methods of this paper. Nevertheless, calibration results were satisfactory.

8 Conclusion

The self-calibration algorithm given here represents a practical approach to camera calibration, giving good accuracy, and showing graceful degradation in the presence of noise. The non-iterative algorithm based on Choleski factorization does not show markedly inferior results than the optimal Levenberg-Marquardt method, and should be preferred except where highest possible accuracy is needed.

The use of the iterative Levenberg-Marquardt method to refine the results allows the calibration problem to be cast as a general parameter fitting problem and allows the imposition of additional constraints, such as the known aspect ratio k_u/k_v, zero skew, or even known rotation angles for the various images.

References

1. K.E. Atkinson. *An Introduction to Numerical Analysis, 2nd Edition.* John Wiley and Sons, New York, 1989.
2. Anup Basu. Active calibration: Alternative strategy and analysis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* pages 495–500, 1993.
3. Lisa Dron. Dynamic camera self-calibration from controlled motion sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* pages 501–506, 1993.
4. Fenglei Du and Michael Brady. Self-calibration of the intrinsic parameters of cameras for active vision systems. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition,* pages 477–482, 1993.
5. O. D. Faugeras, Q.-T Luong, and S. J. Maybank. Camera self-calibration: Theory and experiments. In *Computer Vision - ECCV '92 ,LNCS-Series Vol. 588, Springer-Verlag,* pages 321 – 334, 1992.
6. M. J. Hannah. Bootstrap stereo. In *Proc. Image Understanding Workshop, College Park, MD,* pages 210–208, April 1980.
7. Richard I. Hartley. Euclidean reconstruction from uncalibrated views. In *Proc. of the Second Europe-US Workshop on Invariance, Ponta Delgada, Azores,* pages 187–202, October 1993.
8. Q.-T Luong. *Matrice Fondamentale et Calibration visuelle sur l'environnement.* PhD thesis, Universite de Paris-Sud, Centre D'Orsay, 1992.
9. S. J. Maybank and O. D. Faugeras. A theory of self-calibration of a moving camera. *International Journal of Computer Vision,* 8:2:123 – 151, 1992.
10. William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 1988.