Self-Calibration from Two Views *

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, two 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 two images. The method requires no knowledge of the orientations of the camera, nor the geometry of the scene. 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, and so Maybank and Faugeras's method does not apply. 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.

Keywords : camera calibration, self-calibration, projective transformation, camera matrix.

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 ([7, 4]). 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 solving these equations to obtain the camera calibration has been reported in [7, 4, 6] based on directly solving these quadratic equations using homotopy continuation. An alternative algorithm for calibration of a moving camera has been given in [5], which works for any number of views.

The applicability of these methods is 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.

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

Recently several other papers on self-calibration have appeared ([2, 1, 3]). These papers all rely on known motions of the cameras. In [2] the motion of the camera is assumed to be purely translational. In [1, 3] rotational motions of the camera are considered, but the rotation must be through known angles. This simplifies the calibration task enormously. For instance, in this case, the focal length of the camera can be estimated simply as a ratio of feature displacement to incremental angle of rotation ([3]). In addition, the methods of [1, 3] require tracing features in the image through many frames. In [1] an approximate guess at the location of the principal point is also necessary. In an other paper [**?**], a method of calibration of cameras has been reported which requires three or more views taken from the same point in space. Calibration is carried out solely on the basis of image content.

In this paper, it is shown that if certain natural common assumptions are made about the camera, then it is possible to calibrate the camera from only two views. The two views must be taken from the same location in space, with the camera rotated between views. The calibration is carried out only on the basis of matched points in the two images without any prior knowledge of the scene being viewed, and with only one simple condition on the camera calibration. The method is based on analysis of the projective distortion that an image undergoes when the camera is rotated, but is otherwise quite different from the algorithm used with three or more views ([**?**]). Though verified in practice, these results seem to be quite unexpected, and even counter-intuitive – subject to common assumptions about the calibration, it is possible to calibrate a camera from only two views of an unknown scene. It has generally been tacitly assumed in the past that epipolar geometry is necessary for self-calibration. This paper shows that assumption to be false, since for views with a rotating camera there is not epipolar information.

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.

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

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

where

- k_u is the magnification in the u coordinate direction
- k_v is the magnification in the v coordinate direction
- 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.

Note that K is non-singular. This follows from the requirement that M should have rank 3.

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 but with two different orientations for the two views. For convenience, the common location of the cameras is chosen to be the origin of the coordinate system. We will speak of two cameras each with its own camera matrix, whereas in fact the cameras will be the same, with the same interior parameters, differing only in their orientation. Thus, we consider two cameras with camera matrices $M_1 = K(R_1 \mid 0)$ and $M_2 = K(R_2 \mid 0)$. Further, by orienting the coordinate frame with the first camera, it may be assumed that R_1 is the identity matrix, and hence R_2 is the rotation of the second camera with respect to the first. Often, we will identify a camera with its transformation matrix.

A point $\mathbf{x} = (x, y, z, 1)^\top$ is mapped by the camera M_j to the point $\mathbf{u} = K(R_j \mid 0)(x, y, z, 1)^\top = KR_j(x, y, z)$. 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 both 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 $M_1 = KR_1$ and $M_2 = KR_2$ be two cameras, and let $\mathbf{u}_i^1 = KR_1\mathbf{x}_i$ and $\mathbf{u}_i^2 = KR_2\mathbf{x}_i$. From this it follows that

$$\mathbf{u}_i^2 = KR_2R_1^{-1}K^{-1}\mathbf{u}_i^1$$

This simple observation gives the following important result

Proposition 3.1. *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.

4 Algorithm Idea

The idea of the calibration algorithm will now be described. Suppose we are given two overlapping images J_1 and J_2 both 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. Compute the 2D projective transformation P matching J_1 to J_2. (Section 5)
3. Find an upper triangular matrix K such that $K^{-1}PK = R$ is a rotation matrix for all $j > 0$. The matrix K is the calibration matrix of the cameras, and R represents the orientation of the second camera with respect to the first. Matrix K will not be uniquely determined by this condition. Uniqueness is assured by placing one extra constraint on the calibration matrix. (Section 6)
4. Refine the estimated camera matrix using Levenberg-Marquardt iterative techniques. (Section 7)

The steps of this algorithm will be described in detail in subsequent sections of this paper, as indicated. The main subject of this paper comprises step 3 of this algorithm. The steps of finding point correspondences, computing the transformation P and Levenberg-Marquart refinement of the solution were described in greater detail in [**?**].

5 Determination of the Transformation

Consider a set of matched points $\mathbf{u}_i^1 \leftrightarrow \mathbf{u}_i^2$. It is required to find a two-dimensional projectivity, P mapping each $\mathbf{u}_i^1$ to $\mathbf{u}_i^2$. With four matched points in general position it is well known that a projective transformation P taking each $\mathbf{u}_i^1$ to $\mathbf{u}_i^2$ may be computed. With more than four matched points a least-squares solution may be found that minimizes the matching error. More details may be found in [**?**].

An alternative method to finding point matches and subsequently computing the transform P is to find the P that matches the two images in a more global sense. In particular the transformation P may be adjusted in order to maximize the correlation between the second image and the transformed first image. This technique though promising was not used in the implementation used for evaluating this algorithm.

6 Solving for the calibration matrix

The constraint that the transformation matrix P must be the conjugate of a rotation is not sufficient to determine the conjugating element K exactly. Nevertheless, with just one additional constraint on the calibration matrix it is possible to determine K uniquely. For instance, it will be shown that under the assumption that the skew parameter $s = 0$, calibration matrix K is uniquely determined, and it is possible to calibrate from only two views. Since s is usually very small, the assumption that $s = 0$ is a very reasonable one, commonly used by other authors ([1]). Alternatively, one may make other assumptions about the calibration, for instance that the camera has square pixels, $k_u = k_v$.

According to Proposition 3.1, given two views the transformation taking one image to the other is of the form $P = KRK^{-1}$ where K is the calibration matrix and R is a rotation representing the relative orientation of the two cameras. Matrix P may be normalized so that its determinant $\det P = 1$. Given such a P, it will next be shown how to find an upper-triangular matrix K such that $P = KRK^{-1}$. It will turn out that there exist many such K (in fact a one-parameter family), but for now, we will concentrate on how to find just one of them. Later it will be shown how to find such a K with given desired properties (such as zero skew).

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. Furthermore, it is possible to find a matrix K' such that $P = K'\mathrm{diag}(1, \exp(i\theta), \exp(-i\theta))K'^{-1}$. The columns of K' are the eigenvectors of P. Since the eigenvectors are defined only up to multiplication by a non-zero factor, so are the columns of K'. Multiplying the columns of K' by independent factors preserves the condition that $P = K'\mathrm{diag}(1, exp(i\theta), exp(-i\theta))K'^{-1}$. One could continue this line or reasoning to determine the required calibration matrix, but this involves computations using complex numbers. Instead, we proceed slightly differently.

Any rotation is conjugate to a rotation about the x axis. Since P is conjugate to a rotation, it is therefore conjugate to a rotation about the x axis. From the eigenvalues of P one may determine the angle of rotation, θ. Then one may write $P = HR_xH^{-1}$, and hence $PH = HR_x$. We write

$$R_x = \begin{pmatrix} 1 & & \\ & c & -s \\ & s & c \end{pmatrix}$$

where $c = \cos(\theta)$ and $s = \sin(\theta)$. Further, write $H = (\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3)$ where $\mathbf{h}_i$ is the i-th column of H. Then from $PH = HR_x$ we obtain equations

$$\begin{aligned} P\mathbf{h}_1 &= \mathbf{h}_1 \\ P\mathbf{h}_2 &= c\mathbf{h}_2 + s\mathbf{h}_3 \\ P\mathbf{h}_3 &= -s\mathbf{h}_2 + c\mathbf{h}_3 \end{aligned}$$

This gives rise to a pair of equations

$$(P - I)\mathbf{h}_1 = 0 \tag{2}$$

and

$$\begin{pmatrix} P - cI & -sI \\ sI & P - cI \end{pmatrix} \begin{pmatrix} \mathbf{h}_2 \\ \mathbf{h}_3 \end{pmatrix} = 0 \ . \tag{3}$$

Because of the choice of c and s, the matrices in (2) and (3) will be singular. Consequently, we can solve (2) to find $\mathbf{h}_1$ and (3) to find $\mathbf{h}_2$ and $\mathbf{h}_3$. In the presence of noise, P will not be exactly equal to a conjugate of a rotation. In this case, the equations (2) and (3) will not have an exact solution. The least-squares solution is to be used. From the $\mathbf{h}_i$ we may resassemble a matrix H. This matrix will satisfy $P = HR_xH^{-1}$. Now, using QR decomposition, we may obtain $H = KR$, where K is upper-triangular and R is a rotation. It follows that $P = KRR_xR^{-1}K^{-1} = K\hat{R}K^{-1}$ as required.

It was shown above how to find a matrix H such that $HR_xH^{-1} = P$. Such an H is not unique, and so we now inquire how other solutions may be found. Suppose that $HR_xH^{-1} = P = H'R_xH'^{-1}$. It follows that $(H^{-1}H')R_x = R_x(H^{-1}H')$, in other words, $H^{-1}H'$ commutes with R_x. It may be shown by direct symbolic manipulation that if R_x is not a rotation through 0 or π radians, then $H^{-1}H' = \mathrm{diag}(\alpha_1, \alpha_2, \alpha_2)R'_x$ where R'_x is some other rotation about the x axis. Hence, $H' = H\mathrm{diag}(\alpha_1, \alpha_2, \alpha_2)R'_x$. Since we are only concerned with finding H up to a non-zero scale factor, we may assume that $H' = H\mathrm{diag}(\alpha, 1, 1)R'_x$. Now, if $H = KR$, and $R\mathrm{diag}(\alpha, 1, 1)$ has QR decomposition $K''R''$, then

$$H' = H\mathrm{diag}(\alpha, 1, 1)R'_x = KR\mathrm{diag}(\alpha, 1, 1)R'_x = KK''R''R'_x \ .$$

The foregoing discussion may be summarized in the following proposition.

Proposition 6.2. *Let P be a 2D projective transformation matching two images taken from the same location with the same camera. Let $P = HR_xH^{-1}$ where R_x is a rotation about the x axis. Further, let $H = KR$ be the QR decomposition of H. Then K is a calibration matrix for the camera, consistent with the transformation P. Any other calibration matrix K' consistent with P is of the form $K' = KK''$ where $K''R''$ is the QR decomposition of $R\mathrm{diag}(\alpha, 1, 1)$ for some α.*

This shows that the set of calibration matrices K corresponding to a given transformation matrix P is a one-parameter family. To find a unique calibration matrix, one extra constraint is necessary.

We next turn to the problem of finding a calibration matrix K satisfying additional constraints. To do this, we investigate the QR decomposition of a matrix $R\mathrm{diag}(\alpha, 1, 1)$. Let (r_{ij}) be the entries of the matrix R. The QR decomposition may be computed explicitly. Indeed, it may be verified after some computation that $R\mathrm{diag}(\alpha, 1, 1) = K''R''$ with K'' defined by

$$K'' = \frac{1}{\sqrt{AB}} \begin{pmatrix} \alpha\sqrt{A} & (\alpha^2 - 1)r_{11}r_{21} & (\alpha^2 - 1)r_{11}r_{31}\sqrt{B} \\ 0 & B & (\alpha^2 - 1)r_{21}r_{31}\sqrt{B} \\ 0 & 0 & A\sqrt{B} \end{pmatrix} \tag{4}$$

where $A = (1 - r_{31}^2) + \alpha^2 r_{31}^2$ and $B = r_{11}^2 + \alpha^2(1 - r_{11}^2)$.

There seems to be no pretty way of demonstrating the truth of this formula, and so it must be done by algebraic manipulation. The best way is probably to verify that $K''K''^\top = I + (\alpha^2 - 1)\mathbf{r}_1\mathbf{r}_1{}^\top = R\mathrm{diag}(\alpha, 1, 1)(R\mathrm{diag}(\alpha, 1, 1))^\top$ where $\mathbf{r}_1$ is the first column of R. From this it follows that $R\mathrm{diag}(\alpha, 1, 1) = K''R''$ for some rotation R'' as required. This formula leads us to the following extension to Proposition 6.2.

Proposition 6.3. *Let $P = HR_xH^{-1}$ and $H = KR$. Any calibration matrix consistent with P may be written as KK'' where K'' is of the form given in (4) for some $\alpha > 0$.*

The condition that $\alpha > 0$ is required to ensure that the magnification factor k_u of KK'' remains positive. Now, it is an easy matter to choose α so that the calibration matrix KK'' has desired properties.

Zero skew. We consider the condition that the skew parameter is zero. Suppose $K = (k_{ij})$ and $R = (r_{ij})$. The $(1, 2)$-entry (that is, the skew) in the product KK'' is zero exactly when $k_{11}(\alpha^2 - 1)r_{11}r_{21} + k_{12}B = 0$. Solving for α gives

$$\alpha^2 = \frac{k_{11}r_{11}r_{21} - k_{12}r_{11}^2}{k_{11}r_{11}r_{21} - k_{12}(r_{11}^2 - 1)} \quad ; \quad \alpha > 0 \tag{5}$$

This gives a simple algorithm for the calibration of a camera from two views, assuming that the skew is zero.

1. Compute the transformation matrix P that matches points in the two images, such that $\det P = 1$.
2. Compute the rotation angle θ which is the argument of one of the complex eigenvalues of the matrix P.
3. Find a matrix H such that $P = HR_xH^{-1}$ where R_x is a rotation through angle θ about the x-axis. This is done by solving the equations (2) and (3).
4. Take the QR-decomposition $H = KR$.
5. Find $\alpha > 0$ by solving (5).
6. Compute the QR decomposition $H\text{diag}(\alpha, 1, 1) = K'R'$. The matrix K' is the calibration matrix.

Square pixels. An alternative to setting the skew to zero is to set the two magnifications k_u and k_v in the two axial directions to be equal. Multiplying out KK'' and equating the first two diagonal entries leads to an equation $k_{11}\alpha\sqrt{A} = k_{22}B$. Squaring both sides of this equation leads to a quadratic equation in α^2. In particular, we obtain

$$\alpha^4(k_{11}^2 r_{31}^2) + \alpha^2\left(k_{11}^2(1 - r_{31}^2) - k_{22}^2(1 - r_{11}^2)\right) - k_{22}^2 r_{11}^2 = 0$$

This equation is easily solved for α, but in this case there may be two solutions, since a quadratic equation is involved. We have chosen the strategy of selecting the solution that has the smaller skew. The algorithm for finding the calibration matrix is otherwise the same as the previous one.

7 Iterative Estimation of the Calibration matrix

The non-iterative algorithm given here, although performing quite well does not quite give the optimal solution. For greatest accuracy, least-square techniques may be used for the iterative determination of the calibration matrix K. In particular, we seek a set of points $\mathbf{x}_j$, a matrix K and rotation matrices $R_1 = I$ and R_2 such that

$$\mathbf{u}_i^j = KR_i\mathbf{x}_j + \boldsymbol{\epsilon}_i^j$$

The goal is to minimize the squared error sum, $\sum ||\boldsymbol{\epsilon}_i^j||^2$. The calibration matrix K may be forced to have a special form by the specification of further constraints (such as $s = 0$ or $k_u = k_v$). The solution found by the non-iterative methods is used as an initial seed for the iteration.

The implementation of this method is a straight-forward application of the Levenberg-Marquardt algorithm ([8]). Convergence is rapid and trouble-free, because of the accuracy of the initial values of the parameters. Sparse block techniques described in [9] may be used to separate the determination of the point coordinates $\mathbf{x}_i$ from the determination of the camera parameters, resulting in substantial time savings.

8 Exceptional Cases

It was seen that the calibration algorithm fails if P represents a rotation through 0 or π radians. The first case means that the two images are identical, and the second means that two images are taken with the camera pointing in opposite directions. These special cases are of no interest. There are, however other exceptional cases.

Rotation about the x-axis. If the rotation is about the x axis, then the transformation matrix P is of the form $P = KR_xK^{-1}$ where R_x is a matrix of the form previously given. Any other conjugating element K' satisfying this relationship is of the form $K' = K\text{diag}(\alpha, 1, 1)$ for any α. However, the matrix K' so obtained is the same as K, except that the $(1, 1)$ entry, representing the parameter k_u is multiplied by α. The skew is unchanged. It follows therefore, that constraining the skew to be zero may be an impossible constraint, and in any case puts no restriction on k_u. In other words, we can not determine k_u if the rotation is about the x axis.

Rotation about the y-axis. Similar considerations apply to rotations about the y-axis. In this case, if $P = KR_yK^{-1}$, then any other conjugating element K' must be of the form $K' = K\text{diag}(1, \alpha, 1)$. In this case, the the value of k_v can not be determined.

Rotation about the z-axis. Unless the camera is calibrated, we do not know precisely where the principal axis (that is the z-axis) is. However, if the rotation does happen to be about the z axis, so that K satisfies the condition $P = KR_zK^{-1}$, then any other matrix of the form $K' = K\text{diag}(\alpha, \alpha, 1)$ will do so as well. This means that the two magnification factors, k_u and k_v as well as the skew are multiplied by the factor α. Consequently, it is not possible to determine any of these parameters. Only the position of the principal point and the ratio k_u/k_v may be computed.

9 Experimental Verification of the Algorithm

9.1 Tests with Synthetic Data

First of all, the calibration algorithm was applied to synthetic data to determine its performance in the presence of noise.

Non-iterative algorithm						Levenberg-Marquardt		
Noise	k_u	skew	p_u	p_v	angle	k_u	p_u	p_v
0.0	1000.0	0.0	20.0	30.0	19.29	1000.0	20.0	30.0
0.1	1002.3	0.7	19.8	31.0	19.25	1002.3	19.9	31.0
0.25	1005.7	1.9	19.6	32.6	19.18	1005.7	19.7	32.5
0.5	1011.7	3.8	19.2	35.2	19.07	1011.4	19.4	35.1
1.0	1023.5	9.6	18.2	40.7	18.86	1022.5	18.7	40.4
2.0	1050.7	21.2	16.3	52.4	18.38	1046.6	17.4	51.9
3.0	1082.3	35.5	14.4	65.5	17.85	1072.5	15.9	64.5
4.0	1119.0	53.4	12.4	80.2	17.27	1100.3	14.3	78.5
5.0	1162.2	76.0	10.4	97.0	16.62	1130.4	12.5	94.1

Table 1: Calibration from two images with 50% overlap assuming the condition $k_u = k_v$. For the Levenberg-Marquardt iteration, the condition that skew $s = 0$ was also assumed. The 6-th column shows the computed rotation angle between the two views. The rotation was 19.29 degrees about the x axis.

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. For such a camera, the image measures approximately 35mm by 23mm. When digitized with 20 pixels per mm, the image measures 700×460 pixels. The field of view is approximately $38° \times 26°$. This is approximately the resolution of the images used for the experiments with real images described later. For such images, the magnification factors, k_u and k_v in the two image-plane axial directions are equal to the focal length in pixels. In other words, $k_u = k_v = 1000$. The skew calibration parameter, s was taken to be zero, and the principal point was taken to have coordinates $(p_u, p_v) = (20, 30)$.

The square-pixel constraint: A first set of experiments were conducted with two images overlapping by 50% side-by-side. Thus, the rotation was through an angle of 19.29° (that is, half the image width) about the y axis. A set of 100 matched points were generated, and varying degrees of noise were added. Noise was zero mean Gaussian noise, with the indicated standard deviation. The quoted noise levels are for the deviation applied to *each* of the u and v image coordinates, hence the root-mean-squared pixel displacement is $\sqrt{2}$ times as great. The calibration algorithm was run with the constraint that magnification factors were equal : $k_u = k_v$. First the non-iterative calibration algorithm was run. It was found that for large amounts of noise the skew parameter s became substantially different from zero. Therefore, starting from the calibration already obtained, an iterative Levenberg-Marquardt optimization was run, clamping the skew to zero and maintaining the condition $k_u = k_v$. The results of these experiments are found in table 1. As may be seen, the calibration becomes progressively less exact as noise increases, but for noise levels of the order of 0.5 pixels, which may be obtained in practice, the magnification is accurate to about 1% and the principal point is displaced by about 5 pixels. The results obtained by the Levenberg Marquardt algorithm are not significantly better, except for the zero skew. Note that setting skew to zero does not affect the other parameters very much, which suggests that skew is somewhat hard to estimate exactly.

Non-iterative algorithm						Levenberg-Marquardt		
Noise	k_u	k_v	p_u	p_v	angle	k_u	p_u	p_v
0.0	1000.0	1000.0	20.0	30.0	90.63	1000.0	20.0	30.0
0.1	1002.6	1002.5	19.7	30.5	90.60	1002.3	19.7	30.4
0.25	1006.6	1006.4	19.2	31.2	90.57	1005.9	19.2	30.9
0.5	1013.6	1013.0	18.4	32.3	90.51	1012.3	18.3	31.7
1.0	1028.2	1027.1	16.6	34.7	90.39	1027.0	16.0	33.0
2.0	1088.8	1086.5	0.1	34.2	90.18	1080.4	4.4	28.5
3.0	1160.8	1157.4	-17.4	36.4	89.96	1150.3	-10.6	25.0
4.0	1260.1	1255.8	-43.0	38.4	89.72	1253.3	-34.5	20.2
5.0	1409.2	1404.6	-85.4	40.4	89.48	1457.3	-85.3	15.4

Table 2: Calibration from two images assuming no skew. For the Levenberg-Marquardt iteration, the condition that $k_u = k_v$ was also assumed. The rotation angle is 10° about the x-axis and 90° about the z axis, for a combined rotation of 90.63°.

The zero-skew constraint: A second set of experiments were conducted with the second image panned sideways through 10° and then rotated 90° about the principal axis. In this case, calibration was carried out assuming zero skew. Because of the 90° rotation about the principal axis, the ratio of k_u/k_v was computed very exactly, and a complete Levenberg-Marquardt optimization makes little difference to the final result. These results are shown in table 2.

Using knowledge of the rotation: During the Levenberg-Marquardt parameter fitting it is easy to add a constraint fixing the camera rotation to the known value. This was done for comparison using the same data as in table 1 for noise level of 2.0 pixels. The results of the calibration were then :

$$k_u = k_v = 1000.35 \quad ; \quad p_u = 15.9 \quad ; \quad p_v = 47.7$$

This is (as expected) considerably more accurate that the results for with unknown camera motion. The magnification factors are determined almost exactly, though there is still some error in the estimated position of the principal point (about 20 pixels).

Experiments with real data: Finally, calibration was carried out on a set of images of the capitol taken with a 35mm camera with a zoom lens set at a focal length of approximately 40mm. The five images are shown in Fig 1 and a composite image is shown in Fig 2.

The camera was calibrated from all five images using the algorithm of [**?**]. This calibration result is provided as a good approximation to ground truth, since it is derived from more images and is expected to be accurate. Pairs of images were then taken and calibration carried out. Between 100 and 200 matched points were found between image pairs. The results are given in table 3.

In general, magnification is accurate within 10%, usually much less, and the principal point is accurate within 30 pixels. These results verify the conclusion suggested by the results with synthetic data that best results are obtained using panning rotations and the square-pixel constraint.

Figure 1: Five images of the capitol, numbered 1 – 5 left-to-right and top-to-bottom.

Figure 2: 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.

Image numbers	constraint	k_u	k_v	skew	p_u	p_v	angle
1,2,3,4,5	–	964.4	966.4	-4.9	392.8	282.0	–
2,3	k	1002.9	1002.9	-25.0	330.1	214.8	25.49
2,5	k	963.6	963.6	-11.3	396.5	286.6	31.43
3,5	k	882.2	882.2	38.0	386.1	277.2	23.40
4,5	k	943.7	943.7	-4.7	389.3	250.8	9.57
1,5	k	1197.3	1197.3	-43.7	531.4	416.7	54.15
1,5	s	812.7	819.4	-0.0	381.3	224.3	54.15

Table 3: **Calibration from real images.** The second column shows the type of constraint used (k = square-pixels, s = zero-skew). The first line gives the result of a calibration using all five images, provided as (approximate) ground truth. The next four lines show results of calibration for pairs of images for which the main component of rotation is a panning rotation. For such a rotation, the constraint skew = 0 will not give good results. The sixth and seventh lines show the result for a pair of images that differ by a rotation with its major component about the principal axis. As demonstrated theoretically, rotations about the principal axis do not lead to good calibration results. Accordingly, the results in the last two lines are substantially inferior.

10 Conclusions

The algorithm given in this paper derives the camera calibration from the smallest possible number of views, without using calibration rigs with known geometry. Naturally, the results are inferior to those obtained with a greater number of views, but they suggest that for suitable rotations, particularly panning rotations, the results are quite good. Further work is required to determine the optimal rotation that should be applied to give best calibration.

The mathematical derivations in this paper make clearer the theory behind self-calibration schemes such as those of [1, 3]). As was demonstrated, knowledge about the actual motion of the camera (which was assumed in [1, 3]) may be incorporated into our algorithm to give high quality results.

As a means of calibrating cameras in the field, the methods of this paper and [**?**] seem much more practical than methods based on a moving camera ([7]), both because of the ease of point matching and the simplicity of the calibration algorithms (for instance compare with [6, 5]).

References

[1] Anup Basu. Active calibration: Alternative strategy and analysis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 495–500, 1993.

[2] Lisa Dron. Dynamic camera self-calibration from controlled motion sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 501–506, 1993.

[3] 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.

[4] 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.

[5] 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.

[6] Q.-T Luong. *Matrice Fondamentale et Calibration Visuelle sur l'Environnement.* PhD thesis, Universite de Paris-Sud, Centre D'Orsay, 1992.

[7] 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.

[8] 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.

[9] C. C. Slama, editor. *Manual of Photogrammetry.* American Society of Photogrammetry, Falls Church, VA, fourth edition, 1980.