Digital Image and Vision Computing – Techniques and Application, DICTA/IVCNZ 97, December1997, Auckland, New Zealand.

An Efficient Method for the Estimation of the Fundamental Matrix

Miroslav Trajković and Mark Hedley
Department of Electrical Engineering
University of Sydney, NSW 2006
Australia
E-mail: {miroslav, hedley}@ee.usyd.edu.au

Abstract

In this paper we present a new technique for computing the fundamental matrix. The algorithm finds the epipole to determine the fundamental matrix, and this approach leads to a simpler minimisation problem than previous algorithms. From this we also developed a linear algorithm superior to existing ones.

Keywords – *Fundamental matrix, singularity constraint, epipoles, subspace methods*

1. Introduction

The fundamental matrix is a basic tool in the analysis of scenes recorded using an uncalibrated camera. It is a 3×3 rank 2 matrix that describes the epipolar geometry for a pair of images. The importance of the fundamental matrix has been stressed in the last few years and it is now believed that it will have a crucial role in future applications of Computer Vision. Applications using the fundamental matrix include 3D-structure recovery [1,2], motion segmentation [7] and camera self-calibration [9]. Because of its importance, many algorithms have been developed for estimating the fundamental matrix and they can be classified into two categories:

- Linear, eight-point algorithms ([3, 5, 8]);
- Non-linear algorithms ([2, 10]).

The advantage of the eight-point algorithms is that they are linear, hence fast, and easily implemented. These algorithms do not use the known constraint that ($\det(F) = 0$), and consequently are sensitive to noise. Employing this constraint, which is necessary when less than eight matched feature points are available, results in a non-linear problem. Existing algorithms for this require minimisation of a non-linear function in a high dimensional space (at least seven), which is rather costly for algorithms intended for real-time applications.

Recently, Hartley [3] has shown that by preceding the linear algorithm with a normalisation (translation and scaling) of the coordinates of the matched points, the performance of the linear algorithm improves to the point that the results are comparable with those produced by non-linear techniques.

In this paper, we propose a new linear algorithm for computing the fundamental matrix and show that its performance is superior to Hartley's algorithm, especially when only a small number of matched points are available between the two images. Further, we modify this algorithm into a (non-linear) subspace method that computes the fundamental matrix through an optimisation of the cost function in a compact two-dimensional space (the surface of a unit sphere) and show that this results in superior performance for the algorithm.

2. Background

2.1 Notation

In this paper we assume a pinhole camera model. The coordinates of an arbitrary 3D world point are denoted $\mathbf{P} = (X, Y, Z)^T$ and its projection on the first and the second images are $\mathbf{p} = (x, y, 1)^T$ and $\mathbf{p}' = (x', y', 1)^T$ respectively, where $\mathbf{p}$ and $\mathbf{p}'$ denote homogeneous coordinates.

2.2 Linear Algorithm

A fundamental matrix is defined by the equation:

$$\mathbf{p}'^{\mathbf{T}} F\mathbf{p} = 0 \qquad (1)$$

This matrix is defined up to scale only and has rank two. The eigenvectors of matrix F and F^{T} correspond to the epipoles in the first and second image respectively.

Given n point correspondences, equation (1) can be rewritten as a set of linear equations in the elements of F and we obtain

$$A\mathbf{F} = \mathbf{0}, \qquad (2)$$

where

$$A = \begin{bmatrix} x_1 & x_1' & x_1 y_1' & x_1 & y_1 x_1' & y_1 y_1' & y_1 & x_1' & y_1' & 1 \\ & & & & \vdots & & & & & \\ x_n & x_n' & x_n y_n' & x_n & y_n x_n' & y_n y_n' & y_n & x_n' & y_n' & 1 \end{bmatrix}$$

and

$\mathbf{F} = [\, f_{11} \;\; f_{21} \;\; f_{31} \;\; f_{12} \;\; f_{22} \;\; f_{32} \;\; f_{13} \;\; f_{23} \;\; f_{33}\,]^T$, for $F = \lfloor f_{ij} \rfloor_{3\times 3}$.

If eight or more ideal matches are available the rank of A is exactly eight (except for some degenerate cases) and $\mathbf{F} = \mathbf{h}$, where $\mathbf{h}$ is the unit eigenvector associated with the zero eigenvalue of $A^T A$. In the presence of noise $\mathbf{h}$ is found as a unit vector that minimises $||A\mathbf{h}||$, and this is the unit eigenvector of $A^T A$ associated with the smallest eigenvalue. The matrix F is then determined as

$$F = \begin{bmatrix} h_1 & h_4 & h_7 \\ h_2 & h_5 & h_8 \\ h_3 & h_6 & h_9 \end{bmatrix}. \qquad (3)$$

As det(F) has to be 0 (singularity constraint), the matrix F is replaced by the matrix F' that minimises Frobenius norm $\|F - F'\|$ subject to the condition det(F')=0 and this is done in the following way. Let $F = USV^T$ be the singular value decomposition of F, where S is a diagonal matrix $S = \mathrm{diag}(r, s, t)$ satisfying $r \geq s \geq t$. Then $F' = U\,\mathrm{diag}(r, s, 0)V^T$. The proof that the matrix F' minimises Frobenius norm $\|F - F'\|$ may be found elsewhere (see [10]). This method has been proposed by Tsai and Huang and is widely accepted.

2.3 Normalised linear algorithm

Hartley [3] studied the numerical instability of the linear algorithm when pixel coordinates are directly used and has found that typically the linear algorithm has quite a large condition number which causes its high instability. To overcome this drawback he proposed a simple normalisation of the input data, which significantly reduces the condition number, prior to running the 8-point algorithm. He has found (and it was confirmed elsewhere [10]) that the normalisation step leads to significantly better results, comparable to those obtained by non-linear algorithms His algorithm is summarised below (for more details the interested reader is referred to the original paper [3]):

1. Transform image coordinates according to $\hat{\mathbf{p}}_i = T\mathbf{p}_i$ and $\hat{\mathbf{p}}_i' = T'\mathbf{p}_i'$. The transfor-mations T and T' are chosen in such a way that the centroids of $\hat{\mathbf{p}}_i$ and $\hat{\mathbf{p}}_i'$ are at the origin and the average distance of points to the origin is $\sqrt{2}$.

2. Find the fundamental matrix $\hat{F}$ corres-ponding to the matches $\hat{\mathbf{p}}_i \leftrightarrow \hat{\mathbf{p}}_i'$.

3. Compute the original fundamental matrix as $F = T'^T \hat{F} T$.

3. The computation of the Fundamental Matrix

Let us consider system (2) again. For convenience we will rewrite it here.

$$A\mathbf{h}\mathbf{=0.} \qquad (4)$$

To solve the system we have to perform an eigendecomposition of matrix $A^T A$. Let the eigenvectors of this matrix be $\mathbf{h_1}, \cdots, \mathbf{h_9}$ corresponding to the eigenvalues $\lambda_1, \cdots, \lambda_\mathbf{9}$ in non-decreasing order and let $H = [\mathbf{h_1} \cdots \mathbf{h_9}]$ be a matrix containing these vectors. The best solution of in terms of the least square error (4) is vector $\mathbf{h}_1$ and the error is $\|A\mathbf{h}_1\| = \lambda_1$. However, the fundamental matrix associated with this vector will not, in general, satisfy the constraint $\det(F) = 0$. To satisfy this constraint we are attempting to directly solve the system

$$\min\|A\mathbf{h}\|, \quad \|\mathbf{h}\| = 1, \quad \det(F) = 0 \tag{5}$$

We write the solution of this system as

$$\mathbf{h} = a_1\mathbf{h_1} + \cdots + a_9\mathbf{h_9} \tag{6}$$

where $\mathbf{a} = [a_1 \quad \cdots \quad a_9]^T$ are coefficients to be found. The fundamental matrix F can be written as

$$F = a_1F_1 + \cdots + a_9F_9 \tag{7}$$

where F_i are fundamental matrices correspon-ding to eigenvectors $\mathbf{h}_i$. And now, if the matrix F is to have rank 2 then one of the eigenvalues of F^T must be zero. Let $\mathbf{T}$ denote the unit eigen-vector corresponding to unique zero eigenvalue. Then it must hold:

$$\begin{aligned} &F^T\mathbf{T} = a_1F_1^T\mathbf{T} + \cdots + a_9F_9^T\mathbf{T} = \\ &[F_1^T\mathbf{T} \quad \cdots \quad F_9^T\mathbf{T}]\mathbf{a} = B\mathbf{a} = \mathbf{0} \end{aligned} . \tag{8}$$

We note here that matrix B is of rank 3 and that consequently matrix B^TB must have rank 3. Let $\mathbf{g_1}, \cdots, \mathbf{g_6}$ denote the null space of B^TB and let $G = [\mathbf{g_1} \cdots \mathbf{g_6}]$. Then the solution of (8) may be written as

$$\mathbf{a} = b_1\mathbf{g_1} + \cdots + b_6\mathbf{g_6} = G\mathbf{b} \tag{9}$$

where $\mathbf{b} = [b_1 \quad \cdots \quad b_6]^T$ are the coefficients to be found. Now, equation (4) can be written as

$$A\mathbf{h} = AH\mathbf{a} = AHG\mathbf{b} \tag{10}$$

and the cost function

$$\begin{aligned} &\|A\mathbf{h}\|^2 = (AHG\mathbf{b})^T(AHG\mathbf{b}) = \\ &\mathbf{b^T}G^TH^T(A^TAH)G\mathbf{b} = \\ &\mathbf{b^T}G^TH^T(H\,\mathrm{diag}(\lambda_1 \quad \cdots \quad \lambda_9))G\mathbf{b} = \\ &\mathbf{b^T}G^T\,\mathrm{diag}(\lambda_1 \quad \cdots \quad \lambda_9)G\mathbf{b} = \mathbf{b^T}M(\mathbf{T})\mathbf{b} \end{aligned} \tag{11}$$

Vector $\mathbf{b}$ that minimises (11) is given as the eigenvector corresponding to the smallest eigenvalue of matrix $M(\mathbf{T})$ and the minimum value of the cost function $||A\mathbf{h}||$ is a square root of this eigenvalue [see Appendix]. Once $\mathbf{b}$ is computed, $\mathbf{a}$ is found using (9) and F is found using (7).

3.1 Linear algorithm

As shown in Section 2.2 the linear algorithm is typically performed in two steps: 1) Find the fundamental matrix F that is the best solution of (2) in the least squares sense; and 2) Replace F with matrix F' that minimises Frobenius norm $\|F - F'\|$.

Our linear algorithm has the same first step, but the second step is different. Namely, we first compute the epipole in the second image **e'** as the eigenvector corresponding to the minimum eigenvalue of matrix F^T and then find matrix F'' which has the same epipole and minimises (2), using the procedure described in the previous section:

- Compute $B = B(\mathbf{e'})$
- $G = \mathrm{nullspace}(B^TB)$
- Compute M from (11)
- Compute $\mathbf{b}$ as the eigenvector corresponding to the smallest eigenvalue of matrix M
- Compute a from (8)
- Compute F'' from (7)

Note that F' will have the same epipole as F''. However F'' is the best solution of (5) over the set of matrices characterised by the eigenvector **e'** and hence it has a lower error than F'.

3.2 Subspace algorithm (Non-linear)

From (11), it may be seen that the task of finding the fundamental matrix that minimises (5) can be reduced to the finding the unit vector $\mathbf{T_O}$ that minimises the smallest eigenvalue of $M(\mathbf{T})$. As $\mathbf{T}$ is a unit three-dimensional vector the search space is the surface of the unit sphere, which is a compact two-dimensional space, and the problem is relatively simple. As an initial value we can take the unit eigenvector $\mathbf{T_i}$ corresponding to the smallest eigenvalue of matrix F^T and then use any minimisation method to find the optimal value for $\mathbf{T}$. We noted that the eigenvector corresponding to the zero eigenvalue of the matrix F' is the epipole of the second image **e'**. This means that our algorithm is trying to find the value of the epipole **e'** that minimises the cost function and then recovers the other parameters. This is in line with the results obtained by Luong [4], who found that the stability of fundamental matrix is characterised by the epipole stability. Finally, to perform the optimisation any iterative procedure can be used – we used either the simplex method implemented in MATLAB or Powell's method ([6]) as they do not require the computation of the derivatives.

4. Experimental Results

The algorithms were tested using a number of images with differing placements of the epipole and having a range of the number of matched feature points. The results shown here were obtained for the hotel image sequence and for the image calibration pair (Figures 1 and 2). For the hotel image pair there are 197 matches and for the image calibration pair there are 128 matches. For both images we performed five different algorithms to find the fundamental matrix: Hartley's linear algorithm, our linear algorithm, our subspace algorithm, Luong's algorithm which minimises distances from points to their epipolar lines in both images and the optimal algorithm which minimises distances from points to their reprojections in both images and which is sometimes referred to as the optimal algorithm.

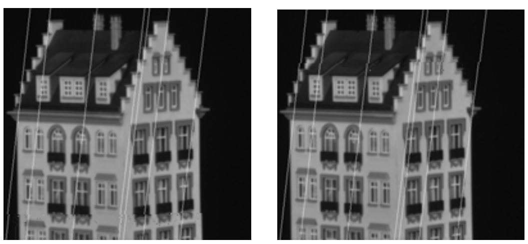

Figure 1. Two images of the hotel image sequence and corresponding epipolar lines.

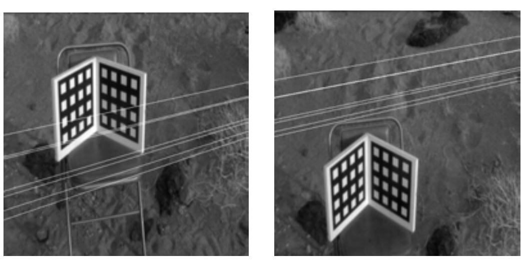

Figure 2. Image calibration pair.

For all algorithms, we first performed norma-lisation described in section 2.3 to improve the condition number of the system. For the first three algorithms we ran our own programs, while for the last two we used programs written by Zhang (accompanying [10]), which are publicly available at INRIA.

The graphs below show the results over thousands of runs of the algorithms with N matched feature points being randomly selected from those available. The algorithms were run fifty times for each selection of set of N points, where N took values between 8 and 50. The error was defined as a distance from a true point location to its optimal reprojection. Projective reconstruction was done using the algorithm introduced by Hartley ([3, 10]). As the true point location was unknown, it was estimated by using all available points, while its optimal reprojection was computed using only N point matches. The graph shows the average error over 50 runs for each value of N.

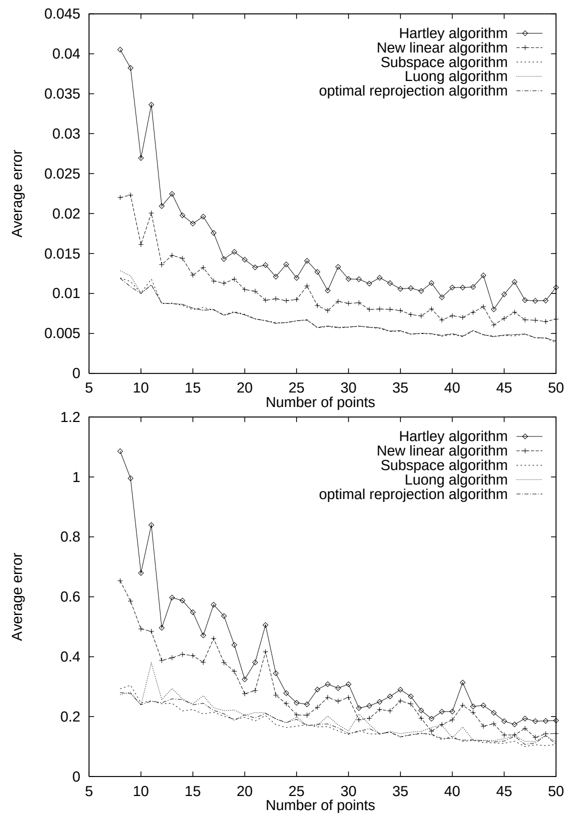

Figure 3. Comparison of Hartley's normalised linear algorithm, our normalised linear algorithm, our subspace algorithm, Luong's algorithm and optimal reprojection algorithm: a) the hotel images pair; b) the calibration pair.

The Figure 3 shows the performance of the five algorithms. The graph on the left is for the data from the hotel sequence. It may be seen that our linear algorithm performs significantly better than Hartley's algorithm, especially for a small number of points, when the average error is about half of that obtained by Hartley's algorithm. The subspace algorithm performs even better having an average error about four times smaller than Hartley's algorithm. It may be noted that our linear algorithm performs consistently better than Hartley's for all values of N. For the first image pair (hotel image) all non-linear algorithms have almost the same performance, while for the second pair our algorithm and the optimal reprojection algorithm have a similar performance, which is better than Luong's algorithm. The significant difference is that optimal reprojection algorithm is much slower than our subspace algorithm.

Figure 4 shows the average reprojection error of the above algorithms versus number of used point matches. The reprojection error is compu-ted as a distance of measured point location $\mathbf{p}_i$ and its reprojection $\hat{\mathbf{p}}_i$. In other words, for a set of N point matches $\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N$ we compute a fundamental matrix F and then compute reprojection set $\hat{\mathbf{p}}_1, \hat{\mathbf{p}}_2, \ldots, \hat{\mathbf{p}}_N$ and the average Euclidean distance $d(\mathbf{p}_i, \hat{\mathbf{p}}_i)$. The average reprojection error is computed by running each algorithm fifty times for each value of N and then computing the mean value of the error.

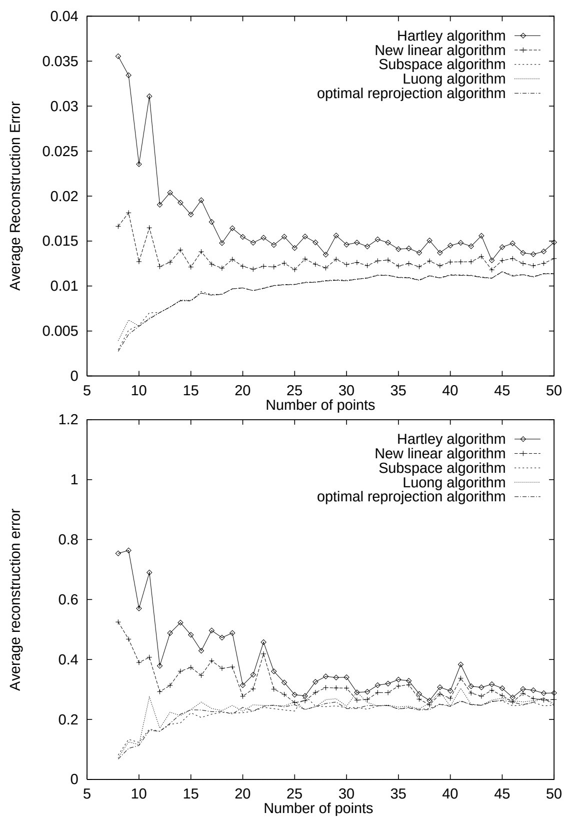

Figure 4. Comparison of average reprojection error obtained by five algorithms. The first graph is for the hotel image pair, while the second one is for the calibration pair.

It may be noted that for a small number of matched feature points our non-linear algorithm performs considerably better than Hartley's algorithm, while our subspace algorithm performs even better. As the number of matches increases, the performance of the algorithms tends to converge. As before, for the second image pair the subspace and optimal reprojection algorithm have similar performance, again better than Luong's algorithm.

5. Conclusion

In this paper we have presented two new algorithms for the estimation of the fundamental matrix. We have shown that the common procedure of imposing the rank-2 constraint on the fundamental matrix by minimising Frobenius norm is suboptimal, and that a significantly better, yet still linear procedure can be employed. In this linear procedure we fix one of the epipoles and then find the fundamental matrix with the same epipole that minimises total error. Further, using the fact that if one of the epipoles is known the fundamental matrix can be uniquely computed, we reduce the task of finding the fundamental matrix (7-D minimisation problem) to the task of finding one of the epipoles (2-D minimisation problem). A possible drawback of our algorithms are that they do not minimise "physically meaningful values" such as the distance from points to their epipolar lines or reprojections, as is typically done by other non-linear algorithms and therefore the small error of our algorithms may be found somewhat surprising. However, Luong [4] has found that the parameterisation of the fundamental matrix has a much larger influence on the stability of the computed fundamental matrix than the minimising criteria. Both algorithms have been tested using a wide range of images. The linear algorithm was found to perform better than existing linear algorithms, and in fact had a performance comparable to non-linear algorithm with a much lower computational cost. The non-linear algorithm was found to perform as well as the best non-linear algorithms, with the advantage that our algorithm is easier to compute due to the smaller optimisation space.

Appendix

Here we prove that the value for **b** which minimises (11) is the eigenvector of $M(\mathbf{T})$ corresponding to the smallest eigenvalue and that the value of the cost function is this eigenvalue.

First, note that the matrix $M(\mathbf{T})$ is symmetric, *i.e*.

$$M^T(\mathbf{T}) = (G^T \Lambda G)^T = G^T \Lambda G = M(\mathbf{T})$$

Consequently, $M(\mathbf{T})$ can be written as

$M(\mathrm{T}) = U^{\mathrm{T}}\Gamma U$

where $U = [\mathbf{u_1} \quad \cdots \quad \mathbf{u_6}]$, $\Gamma = \mathrm{diag}(\gamma_1, \cdots, \gamma_6)$ and $\gamma_1, \cdots, \gamma_6$ are eigenvalues of $M(\mathbf{T})$ in non-decreasing order and $\mathbf{u_1}, \ldots, \mathbf{u_6}$ are correspon-ding eigenvectors. It is easy to show that $M(\mathbf{T})$ is positive definite, *i.e*. all it's eigenvalues are positive. Now, we have

$$\mathbf{b^T} M(\mathbf{T})\mathbf{b} = \mathbf{b^T} U^T \Gamma U \mathbf{b} =$$
$$\mathbf{q^T} \Gamma \mathbf{q} = \gamma_1 q_1^2 + \cdots + \gamma_6 q_6^2$$

and since $\gamma_1 < \cdots \le \gamma_6$ it must be $\mathbf{q} = [\pm 1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$. Without loss of generality we can take plus sign and then

$$\mathbf{b^T} M(\mathbf{T})\mathbf{b} = \gamma_1$$

and

$$\mathbf{b} = U^T \mathbf{q} = \mathbf{u_1} \,,$$

which is the desired result.

References

[1] O. Faugeras, "What can be seen in three-dimensions with an uncalibrated stereo rig?" *Proc. 2nd European Conf. Comp. Vision,* pp. 563-578, 1992.

[2] R. Hartley, "Euclidean reconstruction from uncalibrated views", In *Proc. 2nd Europe-US Workshop on Invariance, editors Mundy J., Zisserman A. and Forsyth D.,* pp. 237-257, 1993.

[3] R. Hartley, "In Defence of the 8-point Algorithm", *in Proc. 5th Internl. Conf. Comp. Vision,* pp. 1064-1070, 1995.

[4] Q-T Luong, *et all*, "On determining the fundamental matrix: analysis of different methods and experimental results", *Report RR-1894, INRIA*, 1993.

[5] H. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections", *Nature,* vol. 293, pp.133-135, September 1991.

[6] W. Press, *et all*, "*Numerical recipes in C: The Art of Scientific Computing"*, Cambridge University Press, 1994.

[7] P. Torr and D. Murrey, "Outlier Detection and Motion Segmentation", *Sensor Fusion VI, editor Schenker, P. S., SPIE*, vol 2059, pp. 432-443, 1993.

[8] R. Tsai and T. Huang, "Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces", *IEEE Trans. PAMI*, vol 6, pp.13-27, 1984.

[9] C. Zeller, and O. Faugeras, "Camera Self-Calibration from Video Sequences: the Kruppa Equations Revisited", *Report RR-1894, INRIA*, 1996.

[10] Z. Zhang, "Determining the Epipolar Geometry and its Uncertainty: A Review", *to appear in Intl. Journal Computer Vision.*