SOLVING FUNDAMENTAL MATRIX FOR UNCALIBRATED SCENE RECONSTRUCTION

Uğur Topay, Engin Tola and *A. Aydın Alatan*
Department of Electrical and Electronics Engineering
M.E.T.U., Balgat, 06531, Ankara, TURKEY
e-mail: alatan@eee.metu.edu.tr

ABSTRACT

3D scene reconstruction from uncalibrated image sequences is a challenging problem. One of its critical subproblems is to solve for fundamental matrix in which the algebraic relations between consecutive images are stored. 8-point, normalized 8-point, algebraic minimization and geometric distance minimization methods are tested for their performances against noise by synthetic and real image simulations. The performances of these methods are also tested for determining camera intrinsic parameters by solving Kruppa equations. Considering their computational complexities and noise robustness, the normalized 8-point algorithm gives a comparable performance against more complex algorithms in terms of errors, especially when the number of corresponding points is high.

1. INTRODUCTION

In order to reconstruct a scene from images taken from different locations, camera calibration (intrinsic camera parameters and relative motion of the images with respect to each other) must be known. 3-D reconstruction of scenes from uncalibrated images is one of the most challenging problems in computer vision. The process of 3D scene reconstruction from uncalibrated images is composed of the following sub-problems: finding corresponding 2D points between images; determination of algebraic relations between the images; camera self-calibration; determination of relative motions between images and calculation of 3D scene points.

The performance of camera self-calibration depends on the accuracy of corresponding points and calculated fundamental matrices (F-matrix) (i.e. algebraic relations between the images). In order to determine the fundamental matrix, the corresponding points on two images are required. In the literature, different methods have been developed for solving F-matrix [1-3]. Four of these methods are introduced and compared in the next sections.

On the other hand, among different methods developed for camera self-calibration, the most well-known is developed by Maybank and Faugeras [6]. In this method, some nonlinear quadratic equations, called as *Kruppa Equations*, are constructed via fundamental matrices and unknown relative camera matrices and tried to be solved in different ways [6-8].

By using fundamental matrices and estimated camera intrinsic parameters, the relative motion (**R, t**) between the cameras or images can be determined by using well-known method developed by Longuet-Higgins [10]. At the end, by using estimated camera intrinsic parameters and relative motions between images, 3D coordinates of corresponding points up to scale can be determined by linear triangulation method [1].

2. SOLVING FUNDAMENTAL MATRIX

The *fundamental matrix* $\boldsymbol{F}$ is the algebraic representation of the epipolar constraint for the uncalibrated cameras. The *epipolar constraint* is described as follows: For each point $\boldsymbol{m}$ in the 1st image plane, its corresponding point $\boldsymbol{m'}$ lies on its epipolar line $\boldsymbol{l'_m}$ and similarly for any point $\boldsymbol{m'}$ in the 2nd image plane, its corresponding point $\boldsymbol{m}$ lies on its epipolar line $\boldsymbol{l_m}$. This relation can be given as:

$$\boldsymbol{l'_m} = \boldsymbol{Fm} \quad \text{and} \quad \boldsymbol{l_m} = \boldsymbol{F}^T\boldsymbol{m'} \qquad (1)$$

Since $\boldsymbol{m}$ lies on $\boldsymbol{l'_m}$ and $\boldsymbol{m'}$ lies on $\boldsymbol{l_m}$, following relations are obtained :

$$\boldsymbol{m'}^T\boldsymbol{Fm} = 0 \quad \text{and} \quad \boldsymbol{m}^T\boldsymbol{F}^T\boldsymbol{m'} = 0 \qquad (2)$$

where $\boldsymbol{m_i} = [u_i,\ v_i,\ 1]^T$ and $\boldsymbol{m_i'} = [u_i',\ v_i',\ 1]^T$. In the next section different methods for solving F-matrix are examined.

2.1. 8-Point Algorithm [10]:

If n corresponding points (at least 8) are given, a set of linear equations is obtained as:

$$\mathbf{Af} = \begin{bmatrix} u_1' u_1 & u_1' v_1 & u_1' & v_1' u_1 & v_1' v_1 & v_1' & u_1 & v_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n' u_n & u_n' v_n & u_n' & v_n' u_n & v_n' v_n & v_n' & u_n & v_n & 1 \end{bmatrix} \begin{bmatrix} f_{11} \\ f_{12} \\ f_{13} \\ f_{21} \\ f_{22} \\ f_{23} \\ f_{31} \\ f_{32} \\ f_{33} \end{bmatrix} = 0 \qquad (3)$$

A robuts solution of this equation is the eigenvector corresponding to the smallest singular value of $\boldsymbol{A}$, that is, the last column of $\boldsymbol{V}$ in the Singular Value Decomposition (SVD) of $\boldsymbol{A} = \boldsymbol{UDV}^T$ [2]. In order to obtain a unique solution, the rank of $\boldsymbol{A}$ matrix must be equal to 8. Therefore, the closest singular $\boldsymbol{F'}$ matrix to F matrix can be obtained as:

$$\boldsymbol{F'} = \boldsymbol{U}\, diag(r,\ s,\ 0)\, \boldsymbol{V}^T \qquad (4)$$

where $\boldsymbol{D} = diag(r, s, t)$ where $r \geq s \geq t$.

2.2. Normalized 8-Point Algorithm [4]:

Hartley [4] proposed a simple normalization on the corresponding points of each image prior to applying 8-point algorithm to improve the performance. This normalization performed by translating center of corresponding points to origin of image reference frame and then scaling the corresponding points so that the average distance from the origin becomes equal to $\sqrt{2}$. Finally, after the calculation of

$\hat{F}$ matrix by using the 8-point algorithm, it is converted to F matrix of corresponding points before normalization as:

$$F = T_2^T \hat{F} T_1 \qquad (5)$$

where T_1 and T_2 are transformation (normalization) matrices for first and second images, respectively.

2.3. Algebraic Minimization Algorithm [1]:

In the 8-Point algorithm, the singular matrix $\boldsymbol{F'}$ is computed by using SVD which minimizes the difference $\|\boldsymbol{F'}\text{-}\boldsymbol{F}\|$ [1]. Since, all the entries of $\boldsymbol{F}$ do not have equal importance, some entries are more affected by the corresponding points. Hence, $\boldsymbol{F}$ can be represented as a product $\boldsymbol{F} = \boldsymbol{M}[\boldsymbol{e}]_x$ where $\boldsymbol{M}$ is a non-singular matrix and $[\boldsymbol{e}]_x$ is the skew-symmetric matrix of the epipole $\boldsymbol{e}$ on the first image. This equation can also be written as:

$$\boldsymbol{f} = \boldsymbol{E}\boldsymbol{\eta} \qquad (6)$$

where $\boldsymbol{E} = \begin{bmatrix} [e]_x & & 0 \\ & [e]_x & \\ 0 & & [e]_x \end{bmatrix}$ and $\boldsymbol{\eta}$ contains the entries of $\boldsymbol{M}$.

Hence, the minimization problem becomes minimizing $\varepsilon = \|\boldsymbol{A}\boldsymbol{E}\boldsymbol{\eta}\|$ subject to the condition $\|\boldsymbol{E}\boldsymbol{\eta}\| = 1$. By using *Levenberg-Marquardt (LM)* algorithm, the epipole $\boldsymbol{e}$ can be varied to minimize $\|\varepsilon\|$. The initial estimate of epipole $\boldsymbol{e}$ can be found from the 8-Point or the Normalized 8-Point Algorithm.

2.4. Geometric Distance Minimization Algorithm [5]:

In this method, the distances between epipolar lines and the corresponding points are minimized [1-3,5]. Therefore, the cost function representing the total square of distances between corresponding points and epipolar lines is given as:

$$\text{Cost}_{GD} = \sum_{i=1}^{n}\left(\frac{1}{(\mathbf{Fm}_i)_1^2 + (\mathbf{Fm}_i)_2^2} + \frac{1}{(\mathbf{F}^T\mathbf{m}_i')_1^2 + (\mathbf{F}^T\mathbf{m}_i')_2^2}\right)(\mathbf{m}_i'\mathbf{Fm}_i)^2$$

In order to minimize this non-linear cost function, the $\boldsymbol{F}$ matrix is parametrized by using epipoles on both images ($\boldsymbol{e}$ and $\boldsymbol{e}'$). The details of parametrization can be found in [5]. Since the parametrization set of the F-matrix can be totally divided into 36 maps, a best map selection algorithm is also proposed [5]. After selecting one of the maps, F-matrix is parametrized and cost function is minimized by using LM algorithm.

3. SOLVING KRUPPA EQUATIONS

CCD camera model utilized during calibrations is given as:

$$\mathbf{K} = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \qquad (7)$$

where $[u_0, v_0, 1]^T$ is the principal point in terms of pixel coordinates, α is the skew angle, $\alpha_u = f/p_x$ and $\alpha_v = f/p_y$ are focal length of the camera in terms of pixel dimensions on the x and y directions, respectively. The Kruppa Equations between two images can be found by using the relation [6-9] :

$$\frac{r^2\mathbf{v}_1^T\mathbf{A}\mathbf{v}_1}{\mathbf{u}_2^T\mathbf{A}\mathbf{u}_2} = \frac{rs\mathbf{v}_1^T\mathbf{A}\mathbf{v}_2}{-\mathbf{u}_2^T\mathbf{A}\mathbf{v}_1} = \frac{s^2\mathbf{v}_2^T\mathbf{A}\mathbf{v}_2}{\mathbf{u}_1^T\mathbf{A}\mathbf{u}_1} \qquad (8)$$

where $\mathbf{v_i}$ and $\mathbf{u_i}$ are the columns of $\boldsymbol{U}$ and $\boldsymbol{V}$ matrices, respectively (SVD($\boldsymbol{F}$) = $\boldsymbol{UDV^T}$) and $\boldsymbol{A}=\boldsymbol{KK^T}$. In order to estimate 5 unknown parameters, one can use at least 6 Kruppa equations obtained from F matrices of 3 image sequences. This nonlinear least squares problem can be solved using LM minimization algorithm for finding the parameters of $\boldsymbol{A}$. Finally, camera calibration matrix $\boldsymbol{K}$ can be calculated from $\boldsymbol{A}$ by Cholesky factorization.

4. SIMULATIONS

4.1. Fundamental Matrix Solutions

The simulations are conducted in two phases using synthetic and real data. During synthetic tests, performance of the algorithms is first tested against correspondence errors by the help of additive Gaussian noise. On the second part, the performance for different number of correspondences is tested. The synthetic scene is composed of two orthogonal planes which are divided into 10×10 grids. The image plane is the size of 35×35mm, the focal length is equal to 50mm. and the image plane is divided into 500×500 pixels. The origin of the pixel frame is placed on the centre of the image plane. The camera calibration matrix is arbitrarily chosen as:

$$K_{synthetic=}\begin{bmatrix} 714.286 & 0 & 0 \\ 0 & 714.286 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Finally, by applying different rotations and translations as the camera motion, approximately 11000 different views of the scene are generated.

As a first step, 100 image pairs are randomly selected. Then, Gaussian noise with zero mean and between 0 to 2 pixels standard deviation is added on both u and v coordinates of the corresponding points. In order to measure the performance, the mean distance between the corresponding points and the epipolar lines is used as an error criterion :

$$\text{Error} = \frac{1}{8}\sum_{i=1}^{8}\left(\frac{1}{(\mathbf{Fm}_i)_1^2 + (\mathbf{Fm}_i)_2^2} + \frac{1}{(\mathbf{F}^T\mathbf{m}_i')_1^2 + (\mathbf{F}^T\mathbf{m}_i')_2^2}\right)(\mathbf{m}_i'\mathbf{Fm}_i)^2 \qquad (9)$$

The mean and standard deviation of errors for 8-Point algorithm are given in Fig. 1. The mean and standard deviation of percentage improvements for other algorithms with respect to 8-Point algorithm are also plotted in Figure 2.

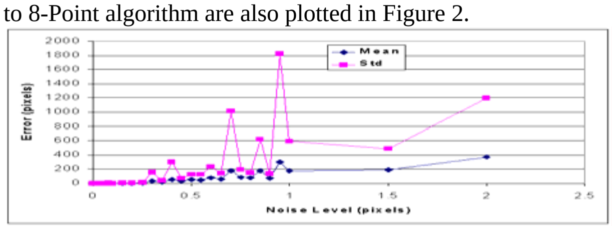

Figure 1 : Mean and standard deviation of errors for 8-Point algorithm for different noise levels

In the second part of synthetic image simulations, the performance of the fundamental matrix methods are compared while increasing the number of corresponding points from 8-200. 100 image pairs, which have corresponding points with 0 mean and 0.5 pixel standard deviation Gaussian noise, are used. The results of simulations are plotted in Figs. 3 and 4.

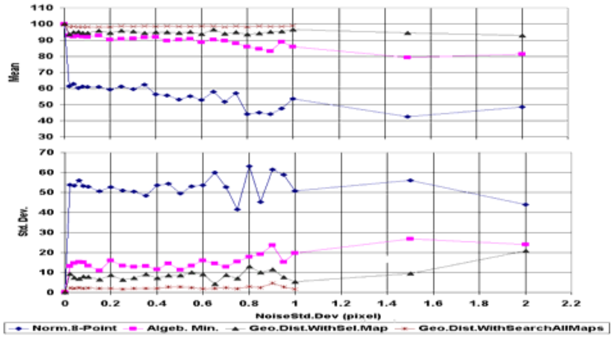

Figure 2 : Mean & standard deviation of *Percentage Improvements* wrt 8-Point Algorithm for different noise levels

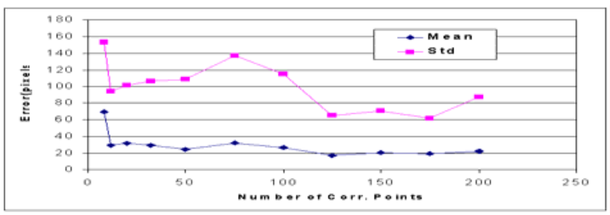

Figure 3 : Means and standard deviations of errors by 8-Point algorithm for different number of corresponding points

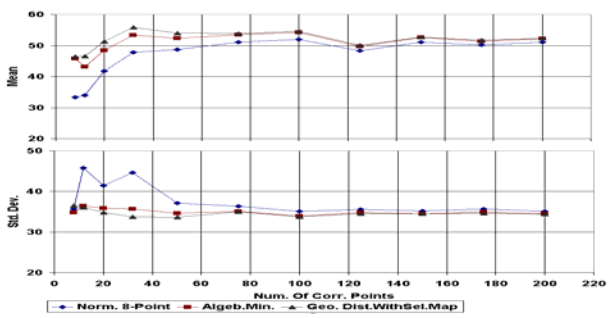

Figure 4 : Means and standard deviations of percentage Improvements for different number of corresponding points

In order to compare the performance of the fundamental matrix methods for real image pairs, the *BatInria* and *ColorIm* image pairs are used with 433 and 266 correspondence points, respectively. In Table 1, the mean distance between the corresponding points and the epipolar lines for image are given.

Algorithm	*BatInria* (pix.)	*ColorIm*(pix.)
8-Point	1.020089	551.9505
Normalized 8-Point	0.234465	9.1086
Algebraic Minimization	0.226639	8.973397
Geometric Distance Min. with Map Selecting	0.227583	8.630695

Table 1 : The mean distances between the corresponding points and epipolar lines for real image pairs

In order to compare the complexity of algorithms, the execution times of algorithms in MATLAB with 200 corresponding points are given in Table 2 (AMD Athlon 1800 MHz. processor and 512 MBytes RAM).

Algorithm	**Exec. time (secs)**
8-Point	< 0.1
Normalized 8-Point	< 0.1
Algebraic Minimization	~ 21.3
Geo. Distance Min. with Map Selecting	~ 936

Table 2 : Execution times for different algorithms

4.2. Results for Camera Self-Calibration :

In order to show the effects of fundamental matrix algorithms to the solution of Kruppa equations, synthetic and real image simulations are performed. In synthetic image simulations, 100 different 3-image sequences are randomly selected from synthetic image collection and then Gausian noise (0 to 0.5 pixel standard deviation) added on 200 corresponding points. Since the LM algorithm needs an initial points near to solution for optimization parameters, the initial camera parameters are taken as: α_u = 785.7146, α_v = 785.7146, s = 10, u_0 = 10 and v_0 = 10. The mean and standard deviation of estimated α_u, s and u_0 parameters with respect to different noise levels are plotted in Fig.5.(a),(b) and (c), respectively.

For the real image simulation, the camera calibration paramaters of *Church* 3-image sequence are estimated by solving Kruppa equations, which are constructed using fundamental matrices obtained with different methods (see Table 3). 128 corresponding points found and the initial values for camera intrinsic parameters are given close to the values in [8]: α_u = 700, α_v = 700, s = 0, u_0 = 300 and v_0 = 400. Finally, the 3D coordinates of corresponding points are estimated by using the camera parameters calculated ny Geometric distance minimization method (Fig. 6).

5. CONCLUSIONS

According to the simulation results, it should be easily stated that 8-Point algorithm should not be preferred in any application, since this algorithm is highly susceptible to noise over the corresponding point coordinates.

It is observed that if the number of corresponding points is limited, the geometric distance minimization algorithm gives the best improvement over 8-point algorithm. However, the normalized 8-point algorithm and algebraic minimization algorithm give similar results for larger number of correspondence points. Since the complexity of the geometric distance minimization algorithm is higher, the normalized 8-point and algebraic minimization algorithm might be preferred for larger number of corresponding points.

Observing different level of errors for the estimated fundamental matrices, one can conclude that the quality of this matrix depends on the relative motion between images. In other words, the noise on some of the correspondences is more effective while estimating the fundamental matrix.

It should also be noted that the best map selection algorithm, which is utilized in geometric distance minimization method is suboptimal. If one performs minimization for all 36 maps, and selects the one with the minimum error value, then the performance of this algorithm is improved while sacrificing from computational complexity.

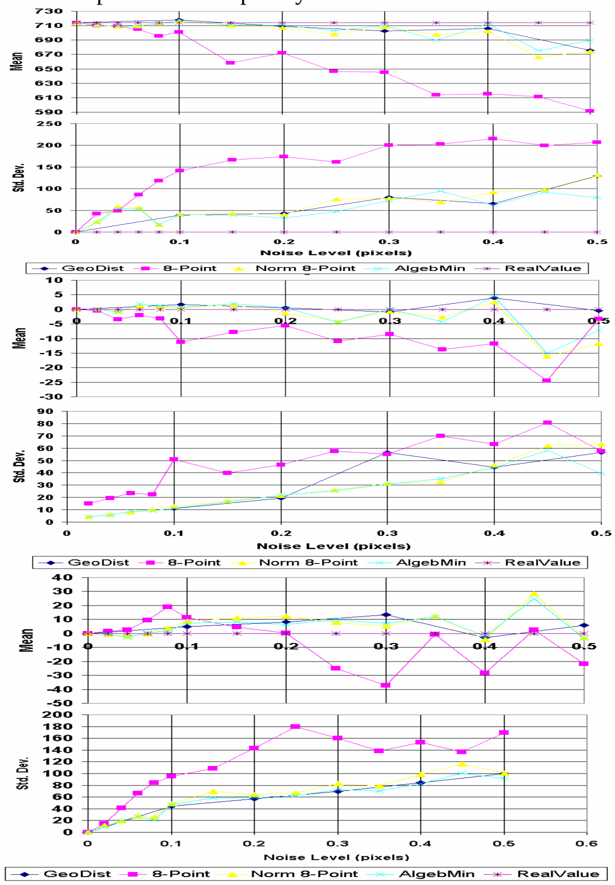

Figure 5 : Mean and standard deviations of estimated (a) α_u (b) *s* and (c) u_0 values after solving Kruppa Equations, formed by the fundamental matrices estimated by different methods

Algorithm	α_u	α_v	**s**	u_0	v_0
Norm.8-Pt	647.78	488.65	-155.89	164.48	441.02
AlgebMin	652.24	453.31	-143.07	122.05	495.76
GeoDisMin	638.87	477.49	-173.10	161.03	486.72

Table 3 : Estimated Camera Intrinsic Parameters for *Church*

The error over the estimated camera calibration parameters increases rapidly after a noise level on the corresponding points. Since the normalized 8-point and algebraic minimization algorithm results with similar errors (even similar for the geometric distance minimization algorithm for high number of corresponding points), all these methods give similar errors on the estimated camera calibration parameters.

Finally, 3D depths of the correspondences of the *Church* image sequence are quite acceptable for geometric distance minimization method, showing the applicability of the algorithms in practice.

6. REFERENCES

[1]Hartley R, Zisserman A, "Multiple View Geometry in Computer Vision", Cambridge University, 2000

[2] Luong Q, Deriche R, Faugeras O, Papadopoulo T, "On Determining The Fundamental Matrix: Analysis of Different Methods and Experimental Results", INRIA, April 1993

[3] Zhang Z, "Determining the Epipolar Geometry and its Uncertainty", International Journal of Computer Vision, 27(2), 161-198, 1998

[4] Hartley R, "In Defence of the 8-Point Algorithm", IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(2), 133-137, 1997

[5] Csurka G, Zeller C, Zhang Z, Faugeras O, "Characterizing the Uncertainty of the Fundamantal Matrix", INRIA, 1995

[6] Faugeras O, Luong Q, Maybank S.J., "Camera Self-Calibration: Theory and Experiments", Proc. of 2nd European Conference on Computer Vision, 321-334, 1992

[7] Luong Q, Faugeras O, "Self-Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices", Int. Jour. of Computer Vision, 22(3), 261-289, 199

[8] Zeller C, Faugeras O, "Camera Self-Calibration from Video Sequences: the Kruppa Equations Revisited", INRIA, 1996

[9] Hartley R, "Kruppa's Equations Derived from the Fundamental Matrix", IEEE PAMI, 19, 2, 1997

[10] Longuet-Higgins H.C., "A Computer Algorithm for Reconstructing a Scene from Two Projections", Mac Millian Journals, 1981

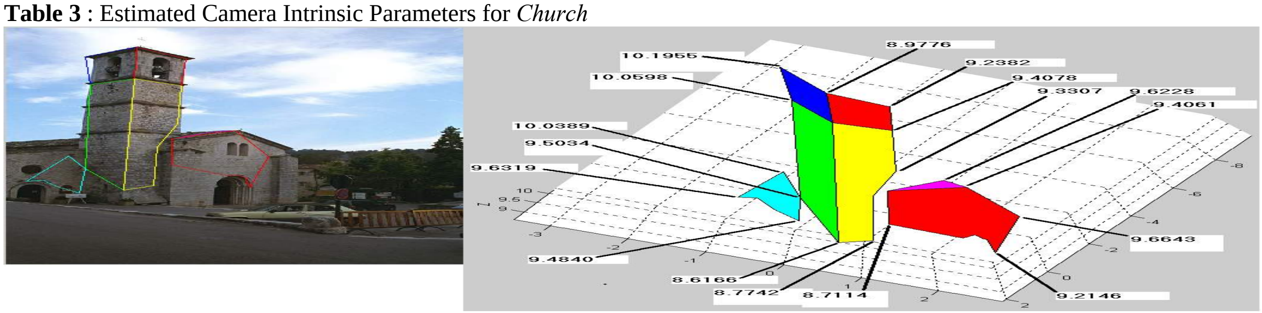

Figure 6: Estimated 3D positions of corresponding points between images1-2 of Church sequence (Geo. Dist. Min.)