MOTION STRUCTURE ANALYSIS AND ERROR QUANTIFICATION - AN EPIPOLAR GEOMETRY BASED APPROACH -

Adriana Karlstroem
Department of Mechatronic and Mechanical Systems Engineering
Escola Politécnica da Universidade de São Paulo
Rua Prof. Mello Moraes, 2231 – Cidade Universitária, São Paulo – SP – Brasil – CEP 05508-900
adriana.karlstroem@poli.usp.br

Fabio Kawaoka Takase
Department of Mechatronic and Mechanical Systems Engineering
Escola Politécnica da Universidade de São Paulo
Rua Prof. Mello Moraes, 2231 – Cidade Universitária, São Paulo – SP – Brasil – CEP 05508-900
fabio.takase@poli.usp.br

Abstract. *The problem of determining object geometry and position from image matches is an open issue in Computer Vision. It can be addressed using epipolar geometry, calculating the fundamental matrix from the correspondence of image pairs from a moving camera sequence. The fundamental matrix obtained from a set of point matches between two images in a motion sequence determines an homography of points in the first image to lines in the second one, and provides, besides a scale factor, the projective matrix that transforms one image into the other. The process of calculating the fundamental matrix and the projective transformation is object to errors due to image quantization, point matches selection, numerical determination of the correspondence equation system, among others. These errors and the propagation of their effects must be carefully considered in any study of methods and algorithms for motion structure and object position determination that are based in epipolar geometry of image sequences. This work presents concepts of epipolar geometry, from camera model, fundamental matrix of image matches, to the determination of the canonical projective matrices, the analytical study of measure and calculation uncertainties of the process, and the analysis of results in real cases, with the objective of developing a method for the extraction of Euclidean geometry information of scenes and motion structure of image sequences.*

Keywords*: computer vision, epipolar geometry, image correspondence, fundamental matrix, error quantification.*

1. Introduction

Two images in a sequence or from a stereo pair of a single scene are related by epipolar geometry, represented by a 3×3 singular matrix. It contains all geometric information necessary for establishing correspondences between the two images, and to infer the three-dimensional structure of the scene. In a stereovision system where the camera geometry is calibrated, and its intrinsic parameters are known, it is possible to calculate the fundamental matrix from the camera perspective projection matrices, and the matrix derived from point correspondences under epipolar geometry is called essential matrix. When the intrinsic parameters are known but the extrinsic ones (the rotation and translation between the two images) are not, the problem is known as motion and structure from motion, and has been extensively studied in Computer Vision. This work concentrates in techniques for estimating the fundamental matrix from two uncalibrated images, where both the intrinsic and extrinsic parameters of the images are unknown. From this matrix, we can reconstruct a projective structure of the scene, represented as a 4×4 matrix transformation.

The study of uncalibrated images has many important applications. We cannot obtain any metric information from a projective structure: measurements of lengths and angles are distorted by a scale factor, and the solution to the projective matrices is not unique. However, this structure still contains information about coplanarity, collinearity, and cross ratios (ratio of ratios of distances), which can be used in robotic systems to perform tasks such as navigation, stabilization and object recognition (Armangué, X. and Salvi, J., 2003; Cheong, L. and Peh, C., 2004; Gracias, N. and Santos-Victor, J., 2000; Zwaan, S., Bernardino, A., and Santos-Victor, J., 2002).

In applications such as the reconstruction of the environment from a sequence of video images, the parameters of the video lens are submitted to continuous modification, and camera calibration is not possible. We cannot reconstruct the exact Euclidean geometry of the scene, but a projective structure can be derived from a pinhole camera model. Assuming that the camera parameters do not change between successive views, the projective invariants can be used to calibrate the cameras and calculate its intrinsic parameters (known as self-calibration).

This work gives an introduction to the epipolar geometry, from camera model, fundamental matrix of image matches, to the determination of the canonical projective matrices, presents two methods for determining the fundamental matrix from point matches in two images from a scene, analytical study of the error in these methods, the application of the uncertainty calculation in the comparison of the presented methods and its use in deriving an improved method. Simulated data and real images are used to build a comparison between the methods and to analyze the results.

2. Epipolar Geometry

A pinhole camera is modeled by its optical center *C* and its retinal plane (or image plane) *R*. A 3D point *W* is projected into an image point *m* given by the intersection of *R* with the line containing *C* and *W*.

Let $\boldsymbol{w} = (x; y; z)$ be the coordinates of *W* in the world reference frame (fixed arbitrarily) and $\boldsymbol{m}$ the pixel coordinates of *m*. Their equivalents in homogeneous (or projective) coordinates are $\tilde{\boldsymbol{w}} = [x \quad y \quad z \quad 1]^T$ and $\tilde{\boldsymbol{m}} = [u \quad v \quad 1]^T$. The *perspective projection matrix* $\boldsymbol{P}$ (or simply *camera matrix*) gives the transformation from $\tilde{\boldsymbol{w}}$ to $\tilde{\boldsymbol{m}}$: $s\tilde{\boldsymbol{m}} = \boldsymbol{P}\tilde{\boldsymbol{w}}$, where *s* is an arbitrary scale factor (*projective depth*). If $\boldsymbol{P}$ is suitably normalized, *s* becomes the true orthogonal distance of the point from the focal plane of the camera.

The 3×4 matrix $\boldsymbol{P}$ can be decomposed as $\boldsymbol{P} = \boldsymbol{A}[\boldsymbol{R} \quad \boldsymbol{t}]$ where the 3×3 matrix $\boldsymbol{A}$ depends on the intrinsic parameters only, and has the following form:

$$A = \begin{bmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

Where $\alpha_u = -fk_u$; $\alpha_v = -fk_v$ are the focal lengths in horizontal and vertical pixels, respectively (*f* is the focal length in millimeters, k_u and k_v are the effective number of pixels per millimeter along the u- and v-axes), (u_0; v_0) are the coordinates of the principal point, given by the intersection of the optical axis with the retinal plane (Fig. 1), and γ is the skew factor. The camera position and orientation (extrinsic parameters), are encoded by the 3×3 rotation matrix $\boldsymbol{R}$ and the translation vector $\boldsymbol{t}$, representing the rigid transformation that aligns the camera reference frame (Fig. 1) and the world reference frame.

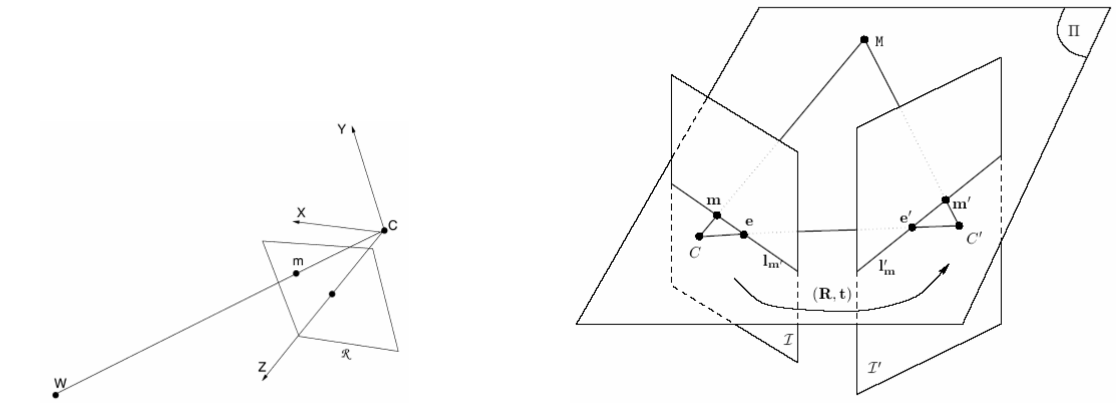

Figure 1. The pinhole camera model (left) and the epipolar geometry model (right).

A line $\boldsymbol{l}$ in the image passing through point $\boldsymbol{m} = [u \quad v]^T$ is described by equation $au + bv + c = 0$. Let $\boldsymbol{l} = [a \quad b \quad c]^T$, then the equation can be rewritten as $\boldsymbol{l}^T\tilde{\boldsymbol{m}} = 0$ or $\tilde{\boldsymbol{m}}^T\boldsymbol{l} = 0$. Multiplying $\boldsymbol{l}$ by any non-zero scalar will define the same 2D line. Thus, a homogeneous 3D vector represents a 2D line. The distance from point $\boldsymbol{m}_0 = [u_0 \quad v_0]^T$ to line $\boldsymbol{l} = [a \quad b \quad c]^T$ is given by $d(\boldsymbol{m}_0, \boldsymbol{l}) = (au_0 + bv_0 + c)/\left(\sqrt{a^2 + b^2}\right)$ (signed distance).

The epipolar geometry exists between any two-camera systems. Consider the case of two cameras as shown in Fig. 1. Let *C* and *C'* be the optical centers of the first and second cameras, respectively. Given a point *m* in the first image, its corresponding point in the second image is constrained to lie on a line called the epipolar line of *m*, denoted by $\boldsymbol{l}'_m$. The line $\boldsymbol{l}'_m$ is the intersection of the plane *I*, defined by *m*, *C* and *C'* (known as the epipolar plane), with the second image plane *I''*. This is because image point *m* may correspond to an arbitrary point on the semi-line *CM* (*M* may be at infinity) and that the projection of *CM* on *I'* is the line $\boldsymbol{l}'_m$. Furthermore, one observes that all epipolar lines of the points in the first image pass through a common point *e'*, which is called the epipole. Epipole *e'* is the intersection of the line *CC'* with the image plane *I'*. This can be easily understood as follows. For each point m_k in the first image *I*, its epipolar line $\boldsymbol{l}'_{m_k}$ in *I'* is the intersection of the plane Π^k, defined by m_k, *C* and *C'*, with image plane *I''*. All epipolar planes Π^k thus form a pencil of planes containing the line *CC'*. They must intersect *I'* at a common point, which is *e'*. Finally, one can easily see the symmetry of the epipolar geometry. The corresponding point in the first image of each point m'_k lying on $\boldsymbol{l}'_{m_k}$ must lie on the epipolar line $\boldsymbol{l}_{m'_k}$, which is the intersection of the same plane Π^k with the first image plane *I*. All epipolar lines form a pencil containing the epipole *e*, which is the intersection of the line *CC'* with the image plane *I*. The symmetry leads to the following observation. If *m* (a point in *I*) and *m'* (a point in *I'*) correspond to a single physical point *M* in space, then *m*, *m'*, *C* and *C'* must lie in a single plane. This is the well-known co-planarity constraint in solving motion and structure from motion problems when the intrinsic parameters of the cameras are known. The computational significance in matching different views is that for a point in the first image, its correspondence in the second image must lie on the epipolar line in the second image, and then the search space for a correspondence is reduced from 2 dimensions to 1 dimension. This is called the epipolar constraint. Algebraically, in order for *m* in the first image and *m'* in the second image to be matched, the following equation must be satisfied:

$$\tilde{\boldsymbol{m}}^T \boldsymbol{F} \tilde{\boldsymbol{m}}' = 0 \text{, with } \boldsymbol{F} = \boldsymbol{A}^{-T} [\boldsymbol{t}]_\times \boldsymbol{R} \boldsymbol{A}'^{-1} \tag{1}$$

Matrix $\boldsymbol{R}$ and vector $\boldsymbol{t}$ are the rigid transformation (rotation and translation) which brings points expressed in the second camera coordinate system to the first one, and $[\boldsymbol{t}]_\times$ is the anti-symmetric matrix defined by $\boldsymbol{t}$ such that $[\boldsymbol{t}]_\times \boldsymbol{x} = \boldsymbol{t} \times \boldsymbol{x}$ for all 3D vector $\boldsymbol{x}$. Without loss of generality, we assume that the world coordinate system coincides with the second camera coordinate system. From the pinhole model, we have $s\tilde{\boldsymbol{m}} = \boldsymbol{A}[\boldsymbol{R} \quad \boldsymbol{t}]\tilde{\boldsymbol{w}}$ and $s'\tilde{\boldsymbol{m}}' = \boldsymbol{A}'[\boldsymbol{I} \quad \boldsymbol{0}]\tilde{\boldsymbol{w}}$.

Eliminating $\tilde{\boldsymbol{w}}$, *s* and *s'* in the above two equations, we obtain equation (1). Geometrically, $\boldsymbol{F}\boldsymbol{m}'$ defines the epipolar line $\boldsymbol{l}_{m'}$ of point $\boldsymbol{m}'$ in the first image. Equation (1) says no more than that the correspondence in the first image of point $\boldsymbol{m}'$ lies on the corresponding epipolar line $\boldsymbol{l}_{m'}$. Transposing (1) yields the symmetric relation from the first image to the second image.

The 3×3 matrix $\boldsymbol{F}$ is called the *fundamental matrix*. Since $det([\boldsymbol{t}]_\times) = 0$, $det(\boldsymbol{F}) = 0$ and $\boldsymbol{F}$ is of rank 2. Besides, it is only defined up to a scalar factor, because if $\boldsymbol{F}$ is multiplied by an arbitrary scalar, equation (1) still holds. Therefore, a fundamental matrix has only seven degrees of freedom. There are only 7 independent parameters among the 9 elements of the fundamental matrix.

The fundamental matrix $\boldsymbol{F}$ can also be expressed in terms of the two camera matrices $\boldsymbol{P}$ and $\boldsymbol{P}'$ (Xu and Zhang, 1996), as $\boldsymbol{F} = [\boldsymbol{e}']_\times \boldsymbol{P}'\boldsymbol{P}^+$, where $\boldsymbol{e}'$ is the epipole in the second image, and $\boldsymbol{P}^+$ is the pseudo-inverse of $\boldsymbol{P}$.

3. Computation of The Fundamental Matrix from Point Correspondences

As showed in the previous section, for any pair $x \leftrightarrow x'$ of matching points in two images, the fundamental matrix $\boldsymbol{F}$ is defined by the equation (1) as $\boldsymbol{x}'^T \cdot \boldsymbol{F} \cdot \boldsymbol{x} = 0$. Given a sufficient number of point matches $\boldsymbol{x}_i \leftrightarrow \boldsymbol{x}'_i$ (at least 7), this equation can be used to compute the unknown matrix $\boldsymbol{F}$. Denoting $\boldsymbol{x} = [x, y, 1]^T$ and $\boldsymbol{x}' = [x', y', 1]^T$, each point match gives rise to one linear equation in the unknown entries of $\boldsymbol{F}$. The coefficients of this equation are easily written in terms of the coordinates of $\boldsymbol{x}$ and $\boldsymbol{x}'$:

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0 \tag{2}$$

We denote by $\boldsymbol{f}$ the 9-vector made up of the entries of $\boldsymbol{F}$ in row-major order, then (2) can be expressed in as the inner product $[x'x, x'y, x', y'x, y'y, y', x, y, 1] \cdot \boldsymbol{f} = 0$.

From a set of *n* point matches we obtain a linear equations system of the form

$$\boldsymbol{A} \cdot \boldsymbol{f} = \begin{bmatrix} x'_1 x_1 & x'_1 y_1 & x'_1 & y'_1 x_1 & y'_1 y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x'_n x_n & x'_n y_n & x'_n & y'_n x_n & y'_n y_n & y'_n & x_n & y_n & 1 \end{bmatrix} \cdot \boldsymbol{f} = 0 \tag{3}$$

This is a homogeneous set of equations and $\boldsymbol{f}$ can be determined only up to scale. For a solution to exist, matrix $\boldsymbol{A}$ must have rank at most 8, and in this case the solution is unique and determined by linear methods (determining the generator of the right null-space of $\boldsymbol{A}$). If the data is not exact, because of noise in the matching points coordinates, then the rank of $\boldsymbol{A}$ may be equal to 9, and we must use a least-squares solution for $\boldsymbol{f}$, that is the singular vector corresponding to the smallest singular value of $\boldsymbol{A}$, or the last column of V in the SVD $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$. This solution minimizes $\|\boldsymbol{A} \cdot \boldsymbol{f}\|$ subject to the condition $\|\boldsymbol{f}\| = 1$.

An important property of the fundamental matrix is that it is singular, with rank 2. Furthermore, the vectors representing the two epipoles in the two images generate the left and right null-spaces of F. The matrix F obtained by solving the set of linear equations in (3) will not in general have rank 2, and this constraint must be enforced a posteriori. The most convenient way to do this is to correct the matrix F found by the SVD solution of A. Matrix F is replaced by the matrix F' that minimizes the Frobenius norm $\|\boldsymbol{F}-\boldsymbol{F}'\|$ subject to the condition $det(\boldsymbol{F}')=0$. This can be done using the SVD $\boldsymbol{F} = \boldsymbol{UDV}^T$, where $\boldsymbol{D}=diag(r,s,t)$ is a diagonal matrix satisfying $r \geq s \geq t$. Then $\boldsymbol{F}'=\boldsymbol{U}\cdot diag(r,s,0)\cdot \boldsymbol{V}^T$ minimizes the Frobenius norm $\|\boldsymbol{F}-\boldsymbol{F}'\|$.

3.1. The Normalized 8-points Algorithm

The method of calculating the vector f described above is the core of the named 8-points algorithm for computation of the fundamental matrix F. It is the simplest method of computing the fundamental matrix, involving no more than the construction and least-squares solution of a set of linear equations. However, it presents high numerical instability when pixel coordinates are directly used. In order to reduce these effects, a proper careful normalization of the input data can be done (Hartley, 2000), consisting of translation and isotropic scaling of the points of both images. This process is represented by the transformation matrices $\boldsymbol{T}$ and $\boldsymbol{T}'$, applied respectively on the points $\boldsymbol{x}_i$ and $\boldsymbol{x}_i'$, such that the normalized points $\hat{\boldsymbol{x}}_i$ and $\hat{\boldsymbol{x}}_i'$ are given by $\hat{\boldsymbol{x}}_i=\boldsymbol{T}\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i'=\boldsymbol{T}'\boldsymbol{x}_i'$, so that the fundamental matrix $\hat{\boldsymbol{F}}$ obtained from the normalized points $\hat{\boldsymbol{x}}_i$ and $\hat{\boldsymbol{x}}_i'$ yields $\boldsymbol{x}_i'^T\cdot\boldsymbol{F}\cdot\boldsymbol{x}_i=0 \rightarrow \hat{\boldsymbol{x}}_i'^T\cdot\boldsymbol{T}'^{-T}\cdot\boldsymbol{F}\cdot\boldsymbol{T}^{-1}\cdot\hat{\boldsymbol{x}}_i=0 \rightarrow \hat{\boldsymbol{F}}=\boldsymbol{T}'^{-T}\cdot\boldsymbol{F}\cdot\boldsymbol{T}^{-1}$.

The normalized 8-points algorithm consists of three steps, as follows:

1. **Normalization**: the coordinates of the point matches in each image are transformed according to $\hat{\boldsymbol{x}}_i=\boldsymbol{T}\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i'=\boldsymbol{T}'\boldsymbol{x}_i'$, where $\boldsymbol{T}$ and $\boldsymbol{T'}$ are normalizing transformations so that the centroid of the selected points is translated to the origin, and the RMS distance of these points to the origin is scaled to $\sqrt{2}$.
2. A solution $\hat{\boldsymbol{F}}$ is obtained from the point matches $\hat{\boldsymbol{x}}_i \leftrightarrow \hat{\boldsymbol{x}}_i'$ by
   a. **Linear solution**: A solution $\hat{\boldsymbol{F}}$ is obtained from the vector $\hat{\boldsymbol{f}}$ corresponding to the smallest singular value of $\hat{\boldsymbol{A}}$, as defined in (3).
   b. **Constraint enforcement**: $\hat{\boldsymbol{F}}$ is replaced by $\hat{\boldsymbol{F}}'$ such that $det(\hat{\boldsymbol{F}}')=0$, determined using SVD as described above.
3. **De-normalization**: The fundamental matrix corresponding to the original data $\boldsymbol{x}_i \leftrightarrow \boldsymbol{x}_i'$ is given by $\boldsymbol{F}=\boldsymbol{T}'^T\cdot\hat{\boldsymbol{F}}\cdot\boldsymbol{T}$.

3.2. The Minimum 7-points Algorithm

In the case that the matrix $\boldsymbol{A}$, as defined in (3), has rank 7, it is still possible to solve for the fundamental matrix by using the singularity constraint. The most important case is when only 7 point-correspondences are known (other cases arise from degeneracies such as points on the same plane, points in a ruled quadric, or no translation of the camera). This leads to a 7×9 matrix $\boldsymbol{A}$, which generally have rank 7.

The solution to the equations $\boldsymbol{A}\cdot\boldsymbol{f}=0$ in this case is a 2-dimensional space of the form $\alpha\boldsymbol{F}_1+(1-\alpha)\boldsymbol{F}_2$, where α is scalar. The matrices $\boldsymbol{F}_1$ and $\boldsymbol{F}_2$ correspond to the generators f1 and f2 of the right null-space of A, given by the singular vector corresponding to the zero singular values of $\boldsymbol{A}$, or the last two columns of $\boldsymbol{V}$ in the SVD $\boldsymbol{A} = \boldsymbol{UDV}^T$. Using $det(\boldsymbol{F})=det(\alpha\boldsymbol{F}_1+(1-\alpha)\boldsymbol{F}_2)=0$, we arrive at a cubic polynomial equation in α. There will be either one or three real solutions to this equation, leading to one or three $\boldsymbol{F}=\alpha\boldsymbol{F}_1+(1-\alpha)\boldsymbol{F}_2$ possible solutions to the fundamental matrix.

3.3. Residual Error

The error is defined as

$$\sum_{i=1}^{N}\left(d(\boldsymbol{x}_i',\boldsymbol{F}\boldsymbol{x}_i)^2+d(\boldsymbol{x}_i,\boldsymbol{F}\boldsymbol{x}_i')^2\right) \tag{4}$$

Where $d(\boldsymbol{x},\boldsymbol{l})$ is the distance (in pixels) between point $\boldsymbol{x}$ and line $\boldsymbol{l}$. The error is the average over all N matches of the squared distance between a point's epipolar line and the matching point in the other image, computed for both points of the match. Note that the error is evaluated considering all *N* matches taken from the images, not only the *n* points used in computing the fundamental matrix $\boldsymbol{F}$.

The two algorithms described in the previous subsections do not directly minimize this error, but the iterative 7-points and recursive normalized 8-points algorithms presented in the next subsections use it to find the best estimative of the fundamental matrix calculated from a given set of point correspondences.

3.4. Iterative 7-points Algorithm

Given a set of *N* point-correspondences $\boldsymbol{x}_i \leftrightarrow \boldsymbol{x}'_i$ between two images, $N > 7$, the iterative 7-points method of calculating the fundamental matrix $\boldsymbol{F}$ consists of dividing it in subsets of 7 point-correspondences each, determining the solutions $\boldsymbol{F} = \alpha \boldsymbol{F}_1 + (1-\alpha)\boldsymbol{F}_2$ as described in section 3.2, determining the residual error as described in section 3.3, and comparing it to the current minimum error value. If the new error value is less than the current minimum, its value becomes current and the new calculated $\boldsymbol{F}$ becomes the estimative of the fundamental matrix. Otherwise, they are both discarded and the current values remain the same.

This method gives the best fitting estimative of the fundamental matrix for a given set of point matches, but becomes extremely computationally costly for large values of N. The number of iterations necessary to exhaust the set of correspondence is the combination of N elements taken 7 at a time, given by $K = N!/(7!(N-7)!)$. For 8 points $K = 8$, for 10 points $K = 120$, for 15 points $K = 6435$, and for 20 points $K = 77520$. As it is, this method will be applied only to small sets of point correspondences.

3.5. Recursive Normalized 8-points Algorithm

This method uses the residual error presented in section 3.3 as minimization criterion for a recursive form of the normalized 8-points algorithm. Let $\boldsymbol{l}'_i = \boldsymbol{F}\tilde{\boldsymbol{x}}_i \equiv [l'_1 \quad l'_2 \quad l'_3]^T$ and $\boldsymbol{l}_i = \boldsymbol{F}^T\tilde{\boldsymbol{x}}'_i \equiv [l_1 \quad l_2 \quad l_3]^T$, then equation (4) can be rewritten as $\sum_{i=1}^{N} w_i^2 \left(\boldsymbol{x}_i'^T \boldsymbol{F}\boldsymbol{x}_i\right)^2$, where $w_i = \sqrt{\left(1/\left(l_1^2 + l_2^2\right)\right) + \left(1/\left(l_1'^2 + l_2'^2\right)\right)}$. The task of finding $\min_F\left(\sum_{i=1}^{N} w_i^2 \left(\boldsymbol{x}_i'^T \boldsymbol{F}\boldsymbol{x}_i\right)^2\right)$ is similar to solving the set of equations shown in (3), and represents a *weighted* solution. If we can compute the weight w_i for each point match, the corresponding equation can be multiplied by w_i, and exactly the same normalized 8-points algorithm can be run to estimate the fundamental matrix, which minimizes (4). As the weights w_i depend themselves on the fundamental matrix, we apply a recursive linear method. We first assume that all $w_i = 1$ and run the normalized 8-points algorithm to obtain an initial estimation of the fundamental matrix. The weights w_i are then computed from this initial solution. Then we re-run the normalized 8-points algorithm on the weighted set of equations. This procedure will be repeated several times, until a minimum error or a maximum iteration threshold is achieved.

4. Characterization of the Fundamental Matrix Uncertainty

As experimental data points are always corrupted by noise, and sometimes there are false matches among the collected data, it is important to model the uncertainty of the estimated fundamental matrix in order to correctly interpret its geometric information. For instance, the covariance of the fundamental matrix can be used to compute the uncertainty of the projective reconstruction or the projective invariants.

In order to quantify the uncertainty of the estimation of the fundamental matrix by the methods described in the previous sections, we model the fundamental matrix as a random vector $\boldsymbol{f} \in \boldsymbol{R}^7$ (vector space of real 7-vectors) whose mean is the exact solution and the uncertainty is given its covariance.

Considering a general random vector $\boldsymbol{y} \in \boldsymbol{R}^p$, where *p* is the dimension of the vector space, the covariance of $\boldsymbol{y}$ is the positive symmetric matrix $\boldsymbol{\Lambda}_y = E\left[(\boldsymbol{y} - E[\boldsymbol{y}])(y - E[\boldsymbol{y}])^T\right]$, where $E[\boldsymbol{y}]$ denotes the mean of the random vector $\boldsymbol{y}$.

4.1. The Statistical Method

The statistical method is based on the supposition that the number *N* of samples $\boldsymbol{y}_i$ of the random vector $\boldsymbol{y}$ is sufficiently large to approximate the mean $E[\boldsymbol{y}]$ by the sample mean:

$$E_N[\boldsymbol{y}] = \frac{1}{N}\sum_{i=1}^{N}\boldsymbol{y}_i \tag{5}$$

And Λ_y is then approximated by $\boldsymbol{\Lambda}_y = \sum_{i=1}^{N}(\boldsymbol{y}_i - E_N[\boldsymbol{y}])(\boldsymbol{y}_i - E_N[\boldsymbol{y}])^T \Big/ (N-1)$.

This method works well for $N > 30$ and it is particularly useful in simulations.

4.2. The Analytical Method

In the present case, **y** is computed from another random vector **x** of $\boldsymbol{R}^m$ using a $\boldsymbol{C}^n$ function φ: $y=\varphi(x)$. Considering a first order approximation of **φ** centered in E[**x**], it yields $E[y]\approx\varphi(E[x])$, where $\varphi(x)-\varphi(E[x])\approx\frac{\partial\varphi(x)}{\partial x}(E[x])\cdot(x-E[x])$, and

$$\Lambda_y=\left(\frac{\partial\varphi(x)}{\partial x}(E[x])\right)\cdot\Lambda_x\cdot\left(\frac{\partial\varphi(x)}{\partial x}(E[x])\right)^T \tag{6}$$

As explained in Section 4, $\boldsymbol{F}$ is computed from the minimization of a sum of squares, and in cases where the parameter is obtained through minimization the $\boldsymbol{\varphi}$ function is implicit, leading to the following result (Zhang, 1998): considering a criterion function $\boldsymbol{C}:\boldsymbol{R}^m\rightarrow\boldsymbol{R}^p$ of $\boldsymbol{C}^\infty$, $\boldsymbol{x_0}\in\boldsymbol{R}^m$ the measurement vector and $\boldsymbol{y_0}\in\boldsymbol{R}^p$ a local minimum of $\boldsymbol{C}(\boldsymbol{x_0};\ \boldsymbol{z})$, if the Hessian $\boldsymbol{H}$ of $\boldsymbol{C}$ with respect to $\boldsymbol{z}$ is invertible at ($\boldsymbol{x}$; $\boldsymbol{z}$) = ($\boldsymbol{x_0}$; $\boldsymbol{y_0}$) then we have $(\partial\varphi(x)/\partial x)(E[x])=-\boldsymbol{H}^{-1}(\partial\boldsymbol{\Phi}/\partial x)$, where $\boldsymbol{\Phi}=(\partial\boldsymbol{C}/\partial\boldsymbol{z})^T$ and $\boldsymbol{H}=\partial\boldsymbol{\Phi}/\partial\boldsymbol{z}$. If $\boldsymbol{x_0}=E[\boldsymbol{x}]$ and $\boldsymbol{y_0}=E[\boldsymbol{y}]$, equation (5) becomes

$$\Lambda_y=\boldsymbol{H}^{-1}\frac{\partial\boldsymbol{\Phi}}{\partial\boldsymbol{x}}\cdot\Lambda_x\cdot\frac{\partial\boldsymbol{\Phi}}{\partial\boldsymbol{x}}^T\boldsymbol{H}^{-T} \tag{7}$$

For the computation of **F** from point correspondences, the criterion is the norm of the implicit function and thus $\boldsymbol{C}$ is of the form $\sum_{i=1}^{N}\boldsymbol{C}_i^2(\boldsymbol{x}_i,\boldsymbol{z})$, where $\boldsymbol{x}=[\boldsymbol{x}_1^T\cdots\boldsymbol{x}_i^T\cdots\boldsymbol{x}_n^T]^T$. Neglecting the second order terms, the results are $\boldsymbol{H}\approx\boldsymbol{2}\sum_i(\partial\boldsymbol{C}_i/\partial\boldsymbol{z})^T(\partial\boldsymbol{C}_i/\partial\boldsymbol{z})$ and $\partial\boldsymbol{\Phi}/\partial\mathbf{x}\approx\boldsymbol{2}\sum_i(\partial\boldsymbol{C}_i/\partial\boldsymbol{z})^T(\partial\boldsymbol{C}_i/\partial\boldsymbol{x})$. Assuming that the noise in $\mathbf{x_i}$ and that in $\mathbf{x_j}$ ($j\neq i$) are independent, then equation (7) becomes

$$\Lambda_y=\boldsymbol{4H}^{-1}\sum_i\frac{\partial\boldsymbol{C}_i}{\partial\boldsymbol{z}}^T\cdot\Lambda_{C_i}\cdot\frac{\partial\boldsymbol{C}_i}{\partial\boldsymbol{z}}\boldsymbol{H}^{-T} \tag{8}$$

Where (up to the first order approximation) $\Lambda_{C_i}=(\partial\boldsymbol{C}_i/\partial\boldsymbol{x}_i)\cdot\Lambda_{x_i}\cdot(\partial\boldsymbol{C}_i/\partial\boldsymbol{x}_i)^T$.

Considering that the mean of the value of $\boldsymbol{C_i}$ at the minimum is zero and that the $\boldsymbol{C_i}$'s are independent and have identical distributed quadratic errors, $\boldsymbol{\Lambda}_{Ci}$ can be approximated by its sample variance $\Lambda_{C_i}=(\boldsymbol{1}/(\boldsymbol{n}-\boldsymbol{p}))\sum_i\boldsymbol{C}_i^2=\boldsymbol{S}/(\boldsymbol{n}-\boldsymbol{p})$, where $\boldsymbol{S}$ is the value of the criterion $\boldsymbol{C}$ at the minimum, and p is the number of parameters (the dimension of **y**). This formula uses p to correct the effect of a small sample set. For $n=p$, almost always can be found an estimate of **y** such that $\boldsymbol{C}_i=0$ for all i, and equation (8) becomes

$$\Lambda_y=\boldsymbol{2}\frac{\boldsymbol{S}}{\boldsymbol{n}-\boldsymbol{p}}\boldsymbol{H}^{-T} \tag{9}$$

Applying this result to the fundamental matrix $\boldsymbol{F}$, with p = 7, and the criterion function $C(\hat{m},f_7)$ (where $\hat{m}=[m_1,m_1',\cdots,m_n,m_n']^T$ and $\boldsymbol{f}_7$ is the vector of the seven chosen parameters of $\boldsymbol{F}$) is given by the minimization of (4). $\boldsymbol{\Lambda}_{f7}$ is thus computed by (9) using the Hessian obtained as a by-product of the minimization of $C(\hat{m},f_7)$. According to (6), $\boldsymbol{\Lambda}_F$ is then computed from $\boldsymbol{\Lambda}_{f7}$, and we have $\Lambda_F=(\partial\boldsymbol{F}(\boldsymbol{f}_7)/\partial\boldsymbol{f}_7)\cdot\Lambda_{f_7}\cdot(\partial\boldsymbol{F}(\boldsymbol{f}_7)/\partial\boldsymbol{f}_7)^T$. Actually, the fundamental matrix $\boldsymbol{F}(\boldsymbol{f}_7)$ is a 9-vector composed of the 9 coefficients which are functions of the 7 parameters of $\boldsymbol{f}_7$.

5. Simulation and Experimental Results

The methods for calculating the fundamental matrix described in section 3 were applied to computer-generated data of two known camera movement cases, the pure translation and the planar motion. The results are used for their comparison and to infer about their behavior apart from uncertainties in the point-correspondences sets, since there is no noise in the input data. The same methods are then applied to experimental point-correspondences sets, obtained from a real image sequence of uncalibrated camera movement around a static scene. Again the results were compared and leaded to conclusions about their behavior when subjected to noisy input data.

5.1. Simulation Results

Pure Translation: The computer-generated data consists of an image of a wireframe cube with a side of 200 pixels, positioned at the origin in 3D space. The camera movement is restricted to translation only, i.e. there is no rotation around the 3D axis. Since all the extrinsic parameters are known, and all intrinsic parameters are also known, the projective matrices relative to each image, and the fundamental matrix, are perfectly determined. Input data is randomly selected from the analytically calculated coordinates of points on the cube's sides to form sets of 7, 8, 10 and 15 point matches between the two images. The sets are used with each algorithm where they apply to calculate an estimative of the fundamental matrix. Table 1 shows the residual error for these estimative for each set-algorithm combination. To test the efficiency of the algorithms we generated a set of 20 points, and the resulting residual error for each estimative of the fundamental matrix is shown in Tab. 2.

Planar Motion: The computer-generated data consists of an image of a wireframe cube with a side of 100 pixels, positioned at the origin in 3D space. The camera now moves on a single plane (only x and y coordinates vary, and the z coordinate is fixed), but is free to rotate around any of the 3D axis. As all parameters are known, the projective matrices relative to each image, and the fundamental matrix, can be exactly determined. Input data is randomly selected from the analytically calculated coordinates of points on the cube's sides to form sets of 7, 8, 10 and 15 point matches between the two images. As in the previous case, the sets are used with each algorithm where they apply to calculate an estimative of the fundamental matrix. Table 1 shows the residual error for these estimative for each set-algorithm combination. Again, the efficiency of the algorithms was tested against another set of 20 points, and the resulting residual error for each estimative of the fundamental matrix is shown in Tab. 2.

Table 1. Residual error of the fundamental matrix estimative (quadratic distance in square pixel fractions).

Algorithm	Pure Translation	Planar Motion	1st Real Pair	2nd Real Pair
Minimal 7-points 1st. solution	0.0000	0.0000	0.0000	0.0000×10^{-4}
Minimal 7-points 2nd solution	0.0000	0.0000	0.0000	0.0000×10^{-4}
Minimal 7-points 3rd solution	0.0000	0.0000	0.0000	0.0000×10^{-4}
Normalized 8-points on 8 point-matches	0.6362	0.0636	0.0006	0.5906×10^{-4}
Recursive normalized 8-points on 8 point-matches	0.6362	0.0636	0.0006	0.5906×10^{-4}
Iterative 7-points on 8 point-matches	0.0018	0.0000	0.0000	0.0135×10^{-4}
Normalized 8-points on 10 point-matches	0.0000	0.0002	0.0021	0.3784×10^{-4}
Recursive normalized 8-points on 10 point-matches	0.0000	0.0002	0.0019	0.4025×10^{-4}
Iterative 7-points on 10 point-matches	0.0000	0.0001	0.0001	0.1500×10^{-4}
Normalized 8-points on 15 point-matches	0.0177	0.0006	0.0006	0.6115×10^{-4}
Recursive normalized 8-points on 15 point-matches	0.0177	0.0006	0.0004	0.9948×10^{-4}
Iterative 7-points on 15 point-matches	0.0061	0.0006	0.0001	0.2220×10^{-4}

Table 2. Residual error of a set of 20 randomly selected point-matches (quadratic distance in square pixel fractions).

Algorithm	Pure Translation	Planar Motion	1st Real Pair	2nd Real Pair
Minimal 7-points 1st. solution	0.4401	0.0809	0.0144	0.0004
Minimal 7-points 2nd solution	32.9524	0.9377	0.4278	0.0011
Minimal 7-points 3rd solution	515.4311	3.8259	1.8355	0.0035
Normalized 8-points on 8 point-matches	14.5389	5.8422	0.0038	0.0002
Recursive normalized 8-points on 8 point-matches	14.5389	5.8422	0.0038	0.0002
Iterative 7-points on 8 point-matches	0.0946	0.0231	0.0005	0.0002
Normalized 8-points on 10 point-matches	0.1545	0.0012	0.0060	0.0001
Recursive normalized 8-points on 10 point-matches	0.1545	0.0012	0.0056	0.0001
Iterative 7-points on 10 point-matches	0.1502	0.0015	0.0005	0.0001
Normalized 8-points on 15 point-matches	0.1984	0.0010	0.0020	0.0001
Recursive normalized 8-points on 15 point-matches	0.1984	0.0010	0.0011	0.0001
Iterative 7-points on 15 point-matches	0.0450	0.0032	0.0002	0.0000

5.2. Experimental Results

The experimental data was obtained from a digital still camera adapted to a computer controlled robot arm, so that the exact displacement between any two images is known. Each image has resolution of 640 × 480 pixels and was stored in JPEG format with maximum quality. The complete image sequence has 12 images, from which were taken two pairs to serve as base for the calculations in this work. Point-correspondences where manually determined with aid of a graphical user interface developed to the purpose of pre-processing the data to be used with the algorithms of computation of the fundamental matrix.

The first pair of images considered here comprises a general movement of the camera, including translation in x, y and z coordinates and rotation around the three axes. The overall displacement on the three coordinates is large enough to be distinctly noted in the images. The input data was carefully selected from the two images to form a set of 20 point-correspondences, which was reduced to subsets of 15, 10, 8 and 7 point-matches for the performance comparisons. The residual error was calculated against the original sets (Tab. 1) and the complete 20 point-matches set (Tab. 2).

The second pair of images comprises a simple translation movement of the camera, displaced along the x, y and z coordinates. The overall camera movement is very small and is only slightly noted in the images. Again the input data is selected from the two images to form a set of 20 point-correspondences, which is reduced to subsets of 15, 10, 8 and 7 point-matches determine the fundamental matrix estimative. As before, the residual error was calculated against the original sets (Tab. 1) and the complete 20 point-matches set (Tab. 2).

6. Conclusion

The results show that the recursive normalized 8-points algorithm does not improve the results from the normalized 8-points algorithm for any of the perfectly matched point-correspondences of the pure translation and planar motion simulations. In the case of the experimental image pairs, the first one shows slight improvement on the results from normalized 8-points to recursive normalized 8-points algorithm, for the 10 and 15 point-matches sets. On the other hand, the second experimental image pair, the results of the original normalized 8-points algorithm are better than the recursive version. From the results shown in Tab. 1 and Tab.2 we notice that the residual error will decrease as the number of point-matches in the input set increases, independent of the algorithm used (excluding the minimal 7-points). Although the minimal 7-points algorithm gives negligible residual error when the original 7 point-matches set is used, the results for the 20 point-matches set are the worst for any of the test cases, simulated or experimental. The iterative 7-points algorithm achieves the best results in all studied cases. The analysis of the simulated results gives the performance of each algorithm when there is no error on the coordinates of the point-matches. Then the residual error observed is due to numerical error introduced by the underlying process. As all those methods rely on the solution of under or over-determined equation systems, a particular choice of point-correspondences can lead to an ill conditioned $\boldsymbol{A}$ matrix in equation (3), they can easily become unstable. In either simulated and experimental cases, the iterative 7-points algorithm will perform better because it searches the complete data set for the combination of point-matches that gives the smallest residual error, and that will automatically exclude the ill conditioned combinations. It could happen that there is no good combination in the complete data set, but that will be less noticed in larger point-matches sets. On the other hand, computational costs will grow as the number of combinations increase. The algorithms that minimize the residual error will also be subject to the underlying uncertainty of the fundamental matrix demonstrated in section 4. The experimental results show that recursive methods that minimize the residual error could give better performance than their direct form, and encourage further studies on the subject of adaptive algorithms derived from the system determination techniques, such as Least Mean Squares (LMS) and Recursive Least Squares (LMS), as well as mixed techniques involving adaptive computation and the methods presented in this work.

7. References

Armangué, X. and Salvi, J., 2003, “Depth Overall view regarding fundamental matrix estimation”, Computer Image and Vision Computing xx (2003), pp. 1–16.

Cheong, L. and Peh, C., 2004, “Depth distortion under calibration uncertainty”, Computer Vision and Image Understanding 93 (2004), pp. 221–244.

Gracias, N. and Santos-Victor, J., 2000, “Underwater Video Mosaics as Visual Navigation Maps”, Computer Vision and Image Understanding 79 (2000), pp. 66–91.

Hartley, R. and Zisserman, A., 2000, “Multiple View Geometry in Computer Vision”, Cambridge University Press.

Horn, R. A. and Johnson, C. R., 1990, “Matrix Analysis”, Cambridge University Press.

Xu, G. and Zhang, Z., 1996, “Epipolar Geometry in Stereo, Motion and Object Recognition, A Unified Approach”, Kluwer Academic Publishers.

Zhang, Z., 1998, “Determining The Epipolar Geometry and its Uncertainty: A Review”, International Journal of Computer Vision Vol. 27(2), pp. 161-195.

Zwaan, S., Bernardino, A., and Santos-Victor, J., 2002, “Visual station keeping for floating robots in unstructured environments” Robotics and Autonomous Systems 39 (2002), pp. 145–155.

8. Responsibility notice

The authors are the only responsible for the printed material included in this paper.