Updated version of ***INRIA Research Report No.2910***, June 1996

Published in ***Journal of the Optical Society of America A***,
Vol.14, no.11, pages 2938–2950, 1997.

Motion and Structure From Two Perspective Views: From Essential Parameters to Euclidean Motion Via Fundamental Matrix

Zhengyou Zhang

INRIA, 2004 route des Lucioles, BP 93, F-06902 Sophia-Antipolis Cedex, France
E-mail: zzhang@sophia.inria.fr, Phone: +33-4-9365-7833, Fax: +33-4-9365-7845

Abstract

The standard approach consists of two stages: (i) using the 8-point algorithm to estimate the 9 essential parameters defined up to a scale factor; (ii) refining the motion estimation based on some statistically optimal criteria, which is a nonlinear estimation problem on a five-dimensional space. Unfortunately, the results obtained are often not satisfactory. The problem is that the second stage is very sensitive to the initial guess, and that it is very difficult to obtain a precise initial estimate from the first stage. This is because we perform a projection of a set of quantities which are estimated in a space of 8 dimensions (by neglecting the constraints on the essential parameters), much higher than that of the real space which is five-dimensional. We propose in this paper a novel approach by introducing an intermediate stage which consists in estimating a 3×3 matrix defined up to a scale factor by imposing the *rank-2 constraint* (the matrix has seven independent parameters, and is known as the fundamental matrix). The idea is to *gradually* project parameters estimated in a high dimensional space onto a *slightly lower* space, namely from 8 dimensions to 7 and finally to 5. The proposed approach has been tested with synthetic and real data, and a considerable improvement has been observed. Our conjecture from this work is that the imposition of the constraints arising from projective geometry should be used as an intermediate step in order to obtain reliable 3D Euclidean motion and structure estimation from multiple calibrated images. The software is available from the Internet.

Keywords: Machine Vision, Motion Analysis, Structure from Motion, Gradual Constraint Enforcing, Multistage Algorithm, 3D Reconstruction

1 Introduction

Motion and structure from motion has been of the central interest in Computer Vision since its infancy, and is still an active domain of research. There are a large number of pieces of work reported in the literature in this domain. The reader is referred to[1–3] for a review. The problem is usually divided into two steps: (i) extract features and match them between images; (ii) determine motion and structure from corresponding features. Points or straight lines are usually used. Line segments have been recently studied by Zhang.[4] The earlier work was mainly on the development of linear algorithms and the existence and uniqueness of solutions.[5–8] More recently, a number of researchers developed algorithms which are noise-resistant by using a sufficient number of correspondences.[9–12] Least-squares or statistical techniques are used to smooth out noise. In these works, the authors assume that matches are given and are correct. In real applications, however, among the feature correspondences established at the first step, several may be incorrect. These false matches (called *outliers* in terms of robust statistics), sometimes even only one, will completely perturb the motion and structure estimation so that the result will be useless. The reader is referred to[13,14] and Appendix A for a technique which uses the least-median-squares method to detect false matches. Instead of perspective views, the structure and motion problem for orthographic or, more generally, affine projections has also been extensively studied since Ullman's pioneer work.[15–19] We also mention recent work on recovering motion and structure from long image sequences.[20–26]

The standard approach to motion and structure estimation problem from two given sets of matched image points consists of two stages: (i) using the 8-point algorithm to estimate the 9 essential parameters defined up to a scale factor, which is a linear estimation problem; (ii) refining the motion estimation based on some statistically optimal criteria, which is a nonlinear estimation problem on a five-dimensional space. Unfortunately, the results obtained using this approach are often not satisfactory, especially when the motion is small or when the observed points are close to a degenerate surface (e.g. plane). The problem is that the second stage is very sensitive to the initial guess and that it is very difficult to obtain a precise initial estimate from the first stage. This is because we perform a projection of a set of quantities which are estimated in a space of 8 dimensions, much higher than that of the real space which is five-dimensional.[27] We propose in this paper a novel approach by introducing an intermediate stage which consists in estimating a 3×3 matrix defined up to a scale factor by imposing the *zero-determinant constraint* (the matrix has seven independent parameters, and is known as the fundamental matrix). The idea is to *gradually* project parameters estimated in a high dimensional space onto a *slightly lower* space, namely from 8 dimensions to 7 and finally to 5. The proposed approach has been tested with synthetic and real data, and considerable improvement has been observed for the delicate situations mentioned above. Note that the constraints we use in the intermediate stage are actually *all* constraints existing between two sets of image points if the images are *uncalibrated* (the intrinsic parameters are not know). That is, we are determining projective motion and structure in this stage. Our conjecture from this work is that the imposition of the constraints arising from projective geometry should be used as an intermediate step in order to obtain reliable 3D Euclidean motion and structure estimation from multiple calibrated images. The local minimum problem is less severe in the projective framework than in the Euclidean one: The optimization for the projective structure often succeeds in locating the true global minimum starting from the unreliable initial guess when the Euclidean optimization does not. This has been shown by Oliensis and Govindu in their experimental study of the accuracy of projective versus Euclidean reconstruction from multiple images.[28]

For readers who are not interested in the implementation details, they can go directly to Sect. 4 to examine how our new multistage algorithm produces much more reliable results. In the following sections, we present our formulation of the motion and structure from motion problem and describe our technique for determining 3D motion and structure. Besides the introduction of the above-mentioned new stage, our technique differs from the classical techniques presented in the literature in the work space used. We directly use *pixel* image coordinates, instead of *normalized* image coordinates. We can reasonably assume that the noise levels in both point coordinates are the same if *pixel* coordinates are used, but they are not the same anymore after having been transformed into *normalized* image coordinates because the scales in the two axes are usually not equal (the ratio is approximately 0.7 in our CCD cameras). A criterion based on pixel image coordinates is thus physically more meaningful. (If the ratio is equal to 1, one can, of course, use either pixel or normalized image coordinates.)

In the Appendix, a robust technique based on the least-median-squares is developed to detect false matches. The complete software, called `SFM`, can be checked out from my home page:

`http://www.inria.fr/robotvis/personnel/zzhang/zzhang-eng.html`

The software also determines the uncertainty, in terms of the covariance matrix, of the estimated motion and structure (see[29] for details).

The intermediate stage introduced in this paper increases the robustness in recovering the motion and structure. The final quality of the motion and structure estimate depends, however, on the criterion used in the final stage. The one we use is based on the maximum likelihood estimation, which can be justified in the limit of small noise. How to design statistically optimal estimators when the data are noisy is an important research field. If we apply *naively* the statistical techniques described in a textbook of statistics to computer vision problems, we may sometimes get surprising results. A deeper understanding of geometric structures in the presence of noise should be promoted. Serious work in this direction includes that of Kanatani,[12] where he also addresses such important problems as the lower bound on the attainable accuracy and the model selection. For example, the algorithm presented here assumes a general motion and structure model. If the camera motion between two views is a pure rotation or if the observed scene is a planar surface, the result given by this algorithm will be useless. Kanatani[30] proposes to test different situations by Akaike information criterion (AIC).

2 Notation and Problem Statement

In this section, we formulate the problem we want to solve and describe the epipolar equation which is fundamental in solving motion and structure from motion problems.

2.1 Notation

A camera is described by the widely used pinhole model. The coordinates of a 3-D point $\mathtt{M} = [x, y, z]^T$ in a world coordinate system and its retinal image coordinates $\mathbf{m} = [u, v]^T$ are related by

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbb{P} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad \text{or} \qquad s\widetilde{\mathbf{m}} = \mathbb{P}\widetilde{\mathtt{M}}$$

where s is an arbitrary scale, and $\mathbb{P}$ is a 3×4 matrix, called the perspective projection matrix. Here, we have used $\widetilde{\mathbf{x}}$ to denote the augmented vector (adding 1 as its last element) of vector $\mathbf{x}$, i.e., if $\mathbf{x} = [x, y, \cdots]^T$, then $\widetilde{\mathbf{x}} = [x, y, \cdots, 1]^T$.

The matrix $\mathbb{P}$ can be decomposed as

$$\mathbb{P} = \mathbf{A}\,[\mathbf{R}\;\mathbf{t}] \quad \text{with} \quad \mathbf{A} = \begin{bmatrix} \alpha_u & c & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} ,$$

where $\mathbf{A}$ is a 3×3 matrix, mapping the normalized image coordinates to the retinal/pixel image coordinates, and $(\mathbf{R}, \mathbf{t})$ is the 3D displacement (rotation and translation) from the world coordinate system to the camera coordinate system. The five parameters in $\mathbf{A}$ are called *intrinsic parameters*, and can be obtained through calibration.[9]

The first and second images are respectively denoted by I_1 and I_2. A point $\mathbf{m}$ in the image plane I_i is noted as $\mathbf{m}_i$. The second subscript, if any, will indicate the index of the point in consideration.

2.2 Problem Statement

We consider two perspective images of a single scene, and we want to determine the relation between the two images and the structure of the scene. This can arise from several situations:

- The two images are taken by a moving camera at two different time instants in a static environment. Then the displacement of the camera and the structure of the scene will be estimated.
- The two images are taken by a fixed camera at two different time instants in a dynamic scene. We assume the two images are projections of a single moving rigid object, otherwise a pre-segmentation of images into different regions is necessary. The displacement and structure of the object will be estimated.
- The two images are taken by two cameras either at the same time or at two different instants. In the latter case, we assume the scene is static. The relative location and orientation of the two cameras and the structure of the scene will be estimated.

In either of the above situations, we assume the cameras are calibrated, i.e., their intrinsic parameters, or the $\mathbf{A}$ matrices, are known. Furthermore, since all these problems are mathematically equivalent, we only consider the third situation.

2.3 Epipolar Equation

Consider now the case of two cameras as shown in Fig. 1, where C_1 and C_2 are the optical centers of the cameras. Let the displacement from the first camera to the second be $(\mathbf{R}, \mathbf{t})$. Let $\mathbf{m}_1$ and $\mathbf{m}_2$ be the images of a 3-D point $\mathtt{M}$ on the cameras. Without loss of generality,

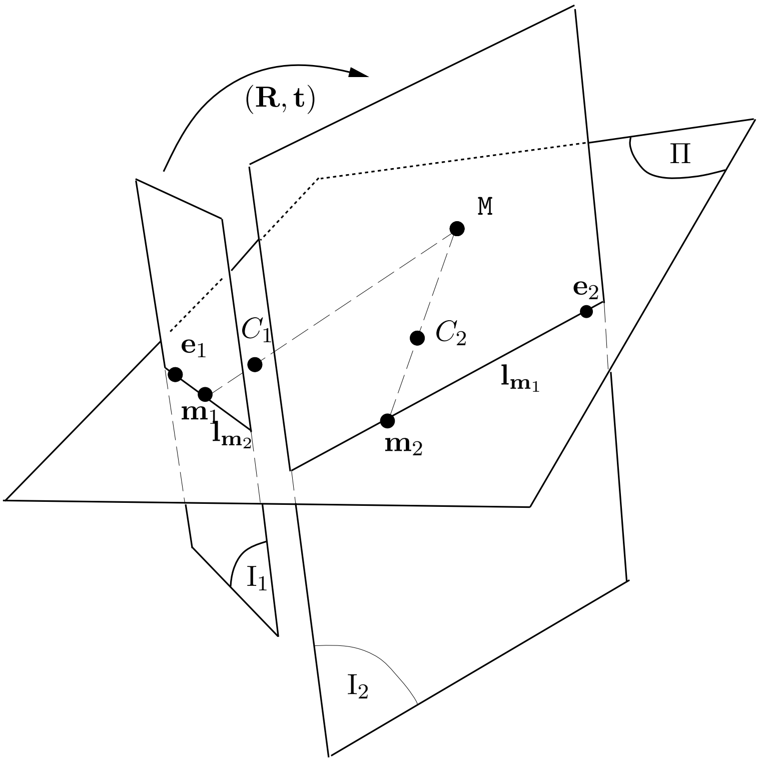


Figure 1: Geometry of motion and structure from motion

we assume that M is expressed in the coordinate frame of the first camera. Under the pinhole model, we have the following two equations:

$$s_1\widetilde{\mathbf{m}}_1 = \mathbf{A}_1\,[\mathbf{I}\ \mathbf{0}]\,\widetilde{\mathtt{M}}\ , \tag{1}$$

$$s_2\widetilde{\mathbf{m}}_2 = \mathbf{A}_2\,[\mathbf{R}\ \mathbf{t}]\,\widetilde{\mathtt{M}}\ , \tag{2}$$

where $\mathbf{A}_1$ and $\mathbf{A}_2$ are the intrinsic matrices of the first and second cameras, respectively. Eliminating M, s_1 and s_2 from the above equations, we obtain the following fundamental equation

$$\widetilde{\mathbf{m}}_2^T\mathbf{A}_2^{-T}[\mathbf{t}]_\times\mathbf{R}\mathbf{A}_1^{-1}\widetilde{\mathbf{m}}_1 = 0\ , \tag{3}$$

where $[\mathbf{t}]_\times$ is an antisymmetric matrix defined by $\mathbf{t}$ such that $[\mathbf{t}]_\times\mathbf{x} = \mathbf{t}\times\mathbf{x}$ for all 3-D vector $\mathbf{x}$ (× denotes the cross product).

Equation (3) is a fundamental constraint underlying any two images if they are perspective projections of one and the same scene. There are two geometric interpretations:

- Equation (3) expresses the fact that four points (C_1, $\mathbf{m}_1$, C_2 and $\mathbf{m}_2$) are coplanar.

- Equation (3) can also be interpreted as the point $\mathbf{m}_2$ lying on the epipolar line of $\mathbf{m}_1$. Let

$$\mathbf{F} = \mathbf{A}_2^{-T}[\mathbf{t}]_\times\mathbf{R}\mathbf{A}_1^{-1}\ ,$$

which is known as the *fundamental matrix*.[31,32] The epipolar line of $\mathbf{m}_1$, denoted by $\mathbf{l}_{\mathbf{m}_1}$ in Fig. 1, is the projection of the semi-line $\mathbf{m}_1C_1$ on the second image, which is defined, up to a scale factor, by the vector $\mathbf{l}_{\mathbf{m}_1} = \mathbf{F}\widetilde{\mathbf{m}}_1$ (i.e., for all point $\mathbf{m}$ on line $\mathbf{l}_{\mathbf{m}_1}$, we have $\widetilde{\mathbf{m}}^T\mathbf{F}\widetilde{\mathbf{m}}_1 = 0$). The fact that $\mathbf{m}_1$ and $\mathbf{m}_2$ correspond to a single point in space implies that $\mathbf{m}_2$ is on $\mathbf{l}_{\mathbf{m}_1}$, which gives equation (3).

For convenience, we use $\mathbf{p}$ to denote a point in the *normalized* image coordinate system, i.e., $\widetilde{\mathbf{p}}_1 = \mathbf{A}_1^{-1}\widetilde{\mathbf{m}}_1$, and $\widetilde{\mathbf{p}}_2 = \mathbf{A}_2^{-1}\widetilde{\mathbf{m}}_2$. We can then rewrite equation (3) as

$$\widetilde{\mathbf{p}}_2^T\mathbf{E}\widetilde{\mathbf{p}}_1 = 0 \qquad \text{with} \quad \mathbf{E} = [\mathbf{t}]_\times\mathbf{R} \ , \tag{4}$$

where $\mathbf{E}$ is known as the *Essential matrix.* It was introduced by Longuet-Higgins,[5] and its property has been studied in the literature.[7,8,33] Because the magnitude of $\mathbf{t}$ can never be recovered from two perspective images, we set $\|\mathbf{t}\| = 1$. The relationship between $\mathbf{E}$ and $\mathbf{F}$ is readily described by $\mathbf{F} = \mathbf{A}_2^{-T}\mathbf{E}\mathbf{A}_1^{-1}$ and $\mathbf{E} = \mathbf{A}_2^T\mathbf{F}\mathbf{A}_1$.

Since $[\mathbf{t}]_\times$ is a skew-symmetric matrix, the determinant of matrix $\mathbf{E} = [\mathbf{t}]_\times\mathbf{R}$ must be zero, i.e., matrix $\mathbf{E}$ is of rank two. In turn, matrix $\mathbf{F}$ is also of rank two. Furthermore, the elements of $\mathbf{E}$ satisfy the following polynomial equation of degree 4:[27,33]

$$\|\boldsymbol{\varepsilon}_1 \times \boldsymbol{\varepsilon}_2\|^2 + \|\boldsymbol{\varepsilon}_2 \times \boldsymbol{\varepsilon}_3\|^2 + \|\boldsymbol{\varepsilon}_3 \times \boldsymbol{\varepsilon}_1\|^2 = \frac{1}{4}(\|\boldsymbol{\varepsilon}_1\|^2 + \|\boldsymbol{\varepsilon}_2\|^2 + \|\boldsymbol{\varepsilon}_3\|^2)^2 \ , \tag{5}$$

where $\boldsymbol{\varepsilon}_i$ is the i^{th} row vector of matrix $\mathbf{E}$.

3 The New Multistage Motion Algorithm

In this section, we first recall the 8-point algorithm, which ignores the constraints on the essential parameters, and the nonlinear algorithm, which performs an optimization directly over the five-dimensional motion space. Finally, we show how we can impose the zero-determinant constraint as an intermediate step in order to provide a better initial estimate for the previously mentioned nonlinear algorithm.

3.1 The Linear Criterion

Equation (4) can be rewritten as a linear and homogeneous equation in the 9 unknown coefficients of matrix $\mathbf{E}$:

$$\mathbf{u}^T\boldsymbol{\epsilon} = 0 \ , \tag{6}$$

where

$$\begin{aligned}
\mathbf{u} &= [x_1x_2, y_1x_2, x_2, x_1y_2, y_1y_2, y_2, x_1, y_1, 1]^T \\
\boldsymbol{\epsilon} &= [E_{11}, E_{12}, E_{13}, E_{21}, E_{22}, E_{23}, E_{31}, E_{32}, E_{33}]^T \ .
\end{aligned}$$

Here $\mathbf{p}_1 = [x_1, y_1]^T$, $\mathbf{p}_2 = [x_2, y_2]^T$, and E_{ij} is the element of $\mathbf{E}$ at row i and column j. If we are given n point matches, by stacking (4), we have the following linear system to solve:

$$\mathbf{U}\boldsymbol{\epsilon} = \mathbf{0} \ ,$$

where $\mathbf{U} = [\mathbf{u}_1, \cdots, \mathbf{u}_n]^T$. This set of linear homogeneous equations, together with the constraints described at the end of Sect. 2.3, allow us to solve for the motion $(\mathbf{R}, \mathbf{t})$.

The minimum number of point matches is five ($n = 5$) because the rotation has three degrees of freedom and the translation is only determined up to a scale factor. Faugeras and

Maybank[7] show that at most ten real solutions are possible in this case, but the algorithm is quite complex. When $n > 5$, we usually have a unique solution, but in some special cases we may have at most three solutions.[8,34,35] The algorithm for $n = 6$ is complex, and is not addressed here.

For $n = 7$, $\text{rank}(\mathbf{U}) = 7$. Through singular value decomposition, we obtain vectors $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_2$ which span the null space of $\mathbf{U}$. The null space is a linear combination of $\boldsymbol{\epsilon}_1$ and $\boldsymbol{\epsilon}_2$, which correspond to matrices $\mathbf{E}_1$ and $\mathbf{E}_2$, respectively. Because of its homogeneity, the essential matrix is a one-parameter family of matrices $\alpha\mathbf{E}_1 + (1-\alpha)\mathbf{E}_2$. Since the determinant of $\mathbf{E}$ must be null, i.e.,

$$\det[\alpha\mathbf{E}_1 + (1-\alpha)\mathbf{E}_2] = 0 \, ,$$

we obtain a cubic polynomial in α. The maximum number of real solutions is 3. For each real solution we substitute it into (5) to check if it is satisfied. When data are noisy, none of the solutions satisfies these constraints. If one solution gives a *much smaller* absolute value of the polynomials than the other solutions, it can be considered as the motion between the two images; otherwise, the three solutions are equally feasible, and must all be considered.

If we are given 8 or more matches and *ignore* the constraints on the essential parameters, we will be able, in general, to determine a unique solution for $\mathbf{E}$, defined up to a scale factor. This can be done by solving the following least-squares problem:

$$\min_{\boldsymbol{\epsilon}} \|\mathbf{U}\boldsymbol{\epsilon}\|^2 \, .$$

Several methods are possible to solve this problem. The first uses a closed-form solution via the linear equations by setting one of the coefficients of $\mathbf{E}$ to 1. The second solves the classical problem:

$$\min_{\boldsymbol{\epsilon}} \|\mathbf{U}\boldsymbol{\epsilon}\|^2 \qquad \text{subject to } \|\boldsymbol{\epsilon}\| = \sqrt{2} \, . \tag{7}$$

The constraint on the norm of $\boldsymbol{\epsilon}$ is derived from the fact that $\mathbf{R}$ is an orthonormal matrix and $\|\mathbf{t}\| = 1$.[9] The solution is the eigenvector of $\mathbf{U}^T\mathbf{U}$ associated with the smallest eigenvalue. This approach, known as the eight-point algorithm, was proposed by Longuet-Higgins[5] and has been extensively studied in the literature.[6,11,36] It has been proven to be very sensitive to noise.

Once we have estimated the essential matrix $\mathbf{E}$, we can recover the motion $(\mathbf{R}, \mathbf{t})$. See[9,29] for the details.

The advantage of the linear criterion is that it leads to an analytic solution. However, we have found that it is quite sensitive to noise, even with a large set of data points. There are two reasons for this:

- We have omitted the constraints on the essential matrix. The elements of $\mathbf{E}$ are not independent from each other.

- The quantity we try to minimize (7) does not have much physical meaning.

3.2 Minimizing the Distances to Epipolar Lines

As was described in Sect. 2.3, $\mathbf{F}\widetilde{\mathbf{m}}_1$ represents actually the epipolar line of $\mathbf{m}_1$ in the second image. If $\mathbf{m}_2$ corresponds exactly to $\mathbf{m}_1$, we would expect the distance from $\mathbf{m}_2$ to the epipolar line $\mathbf{F}\widetilde{\mathbf{m}}_1$ to be zero. Thus, a natural idea is to use a nonlinear criterion by minimizing:

$\sum_i d^2(\widetilde{\mathbf{m}}_{2i}, \mathbf{F}\widetilde{\mathbf{m}}_{1i})$, where $d(\widetilde{\mathbf{m}}_2, \mathbf{F}\widetilde{\mathbf{m}}_1)$ is the Euclidean distance of point $\mathbf{m}_2$ to its epipolar line $\mathbf{F}\widetilde{\mathbf{m}}_1$ in the second image. It is given by

$$d(\widetilde{\mathbf{m}}_2, \mathbf{F}\widetilde{\mathbf{m}}_1) = \frac{\widetilde{\mathbf{m}}_2^T \mathbf{F}\widetilde{\mathbf{m}}_1}{\sqrt{(\mathbf{F}\widetilde{\mathbf{m}}_1)_1^2 + (\mathbf{F}\widetilde{\mathbf{m}}_1)_2^2}} \ ,$$

where $(\mathbf{F}\widetilde{\mathbf{m}}_1)_i$ is the i^{th} component of vector $\mathbf{F}\widetilde{\mathbf{m}}_1$, and the distance is *signed*. However, unlike the case of the linear criterion, the two images do not play a symmetric role. To obtain a consistent epipolar geometry, we also consider distances in the first image. This yields the following criterion:

$$\sum_i \left(d^2(\widetilde{\mathbf{m}}_{2i}, \mathbf{F}\widetilde{\mathbf{m}}_{1i}) + d^2(\widetilde{\mathbf{m}}_{1i}, \mathbf{F}^T\widetilde{\mathbf{m}}_{2i})\right) \ ,$$

which operates simultaneously in the two images. Using the fact that $\widetilde{\mathbf{m}}_2^T \mathbf{F}\widetilde{\mathbf{m}}_1 = \widetilde{\mathbf{m}}_1^T \mathbf{F}^T\widetilde{\mathbf{m}}_2$, it can be rewritten as:

$$\sum_i \left(\frac{1}{(\mathbf{F}\widetilde{\mathbf{m}}_{1i})_1^2 + (\mathbf{F}\widetilde{\mathbf{m}}_{1i})_2^2} + \frac{1}{(\mathbf{F}^T\widetilde{\mathbf{m}}_{2i})_1^2 + (\mathbf{F}^T\widetilde{\mathbf{m}}_{2i})_2^2} \right) (\widetilde{\mathbf{m}}_{2i}^T \mathbf{F}\widetilde{\mathbf{m}}_{1i})^2 \ . \tag{8}$$

Unlike the case of the linear criterion which uses the elements of the essential matrix, we minimize the above functional over the motion parameters. Recall that we deal with calibrated cameras, i.e., $\mathbf{F}$ depends only on $\mathbf{R}$ and $\mathbf{t}$. The rotation is represented by a 3D vector, whose direction is parallel to the rotation axis and whose magnitude is equal to the rotation angle. The translation is represented by its spherical coordinates. Thus, the minimization is carried out over these five unknowns. As the minimization is nonlinear, we use the result of the analytical method as its initial guess.

In the above formulation, we use the *pixel* image coordinates $\mathbf{m}_{ij}$. We can also use the *normalized* image coordinates $\mathbf{p}_{ij}$ with a similar formulation (i.e., replace $\mathbf{F}$ and $\mathbf{m}$ in (8) by $\mathbf{E}$ and $\mathbf{p}$). We have implemented both criteria. Experiments[29] have shown that better results were obtained using *pixel* image coordinates, i.e., (8), than using *normalized* image coordinates. This is because points are usually extracted in pixel images, but not in normalized images. We can reasonably assume that the noise levels in both point coordinates are the same if *pixel* coordinates are used, but they are not the same anymore after having been transformed into *normalized* image coordinates because the scales in the two axes are usually not equal (the ratio is approximately 0.7 in our CCD cameras). Hence, the criterion (8) is physically more meaningful than using normalized image coordinates.

The criterion (8) is only empirical. Sect. 3.4 will give a criterion based on the maximum likelihood principle. It involves both motion and structure parameters, and its optimization is computational much more expensive. The criterion (8) is used as an approximation in order to achieve a faster convergence with the maximum likelihood criterion. Kanatani[12] has derived, through variational analysis, a better criterion which only involves the motion parameters.

3.3 3D Reconstruction

Once we know the motion $(\mathbf{R}, \mathbf{t})$, given a match $(\mathbf{m}_1, \mathbf{m}_2)$, we can estimate the 3D coordinates $\mathtt{M}$ by minimizing the distance between the back-projection of the 3D reconstruction and the observed image point, that is

$$\hat{\mathtt{M}} = \arg\min_{\mathtt{M}} \left(\|\mathbf{m}_1 - \mathbf{h}_1(\mathbf{a}, \mathtt{M})\|^2 + \|\mathbf{m}_2 - \mathbf{h}_2(\mathbf{a}, \mathtt{M})\|^2 \right) ,$$

where $\mathbf{h}_1(\mathbf{a}, \mathtt{M})$ and $\mathbf{h}_2(\mathbf{a}, \mathtt{M})$ are the camera projection functions corresponding to (1) and (2), respectively.

3.4 Maximum Likelihood Estimation

Let $\mathbf{a} = [\mathbf{r}^T, \boldsymbol{\phi}^T]^T$ be the 5-D vector composed of three parameters representing the rotation between the two images and two parameters representing the translation (see Sect. 3.2). Let $\mathtt{M}_j$ be the 3-D vector corresponding to the j^{th} point expressed in the coordinate system associated with the first camera. The motion and structure parameters are then represented by a vector of $(5 + 3n)$ dimensions, denoted by

$$\boldsymbol{\theta} = [\mathbf{a}^T, \mathtt{M}_1^T, \ldots, \mathtt{M}_j^T, \ldots, \mathtt{M}_n^T]^T .$$

Assume that each point $\mathbf{m}_{ij}$ is corrupted by additive independent Gaussian noise $N(0, \boldsymbol{\Lambda}_{ij})$. The maximum likelihood (ML) estimate, $\hat{\boldsymbol{\theta}}$, of the parameter vector $\boldsymbol{\theta}$ is the solution to the following weighted nonlinear least-squares formulation:

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{2} \sum_{j=1}^{n} \boldsymbol{\delta}\mathbf{m}_{ij}^T \boldsymbol{\Lambda}_{ij}^{-1} \boldsymbol{\delta}\mathbf{m}_{ij} \quad \text{with } \boldsymbol{\delta}\mathbf{m}_{ij} = \mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j) , \tag{9}$$

which is actually the sum of squared Mahalanobis distances. Here, $\mathbf{h}_1(\mathbf{a}, \mathtt{M}_j)$ and $\mathbf{h}_2(\mathbf{a}, \mathtt{M}_j)$ are the camera projection functions corresponding to (1) and (2), respectively. Due to the nonlinear nature of perspective projection, the solution to the above problem demands the use of numerical nonlinear minimization technique such as the Levenberg-Marquardt algorithm implemented in the `MINPACK` library.[37] An initial guess on the motion and structure is required, which can be obtained by using the techniques described previously.

The exact value of the covariance matrix $\boldsymbol{\Lambda}_{ij}$ is very difficult to obtain in practice. It depends on the point detector used, and on the image intensity variation in the neighborhood of the feature point. However, qualitatively, we expect, and it is confirmed by our experience, that the noise is reasonably isotropic in the image plane and identically distributed. Thus, we can assume $\boldsymbol{\Lambda}_{ij} = \sigma^2 \ \mathrm{diag}\,(1, 1)$, where σ is called the *noise level* which depends on the quality of the point detector used. We do not need to know the noise level, because the minimization is not affected by a multiplication of a constant value. From this assumption, the problem (9) can be simplified as

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i=1}^{2} \sum_{j=1}^{n} \|\mathbf{m}_{ij} - \mathbf{h}_i(\mathbf{a}, \mathtt{M}_j)\|^2 . \tag{10}$$

The solution to either (9) or (10) requires a numerical minimization to be performed in a $(5 + 3n)$-D space, which is huge for a large n. By examining the above formulation, we find that the structure parameter $\mathtt{M}_j$ is only involved in two terms. We can thus "separate" the motion estimation from the structure estimation,[11] namely

$$\hat{\boldsymbol{\theta}} = \arg\min_{\mathbf{a}} \sum_{j=1}^{n} \left[\min_{\mathtt{M}_j} \left(\|\mathbf{m}_{1j} - \mathbf{h}_1(\mathbf{a}, \mathtt{M}_j)\|^2 + \|\mathbf{m}_{2j} - \mathbf{h}_2(\mathbf{a}, \mathtt{M}_j)\|^2 \right) \right] . \tag{11}$$

That is, we conduct an outer minimization over $\mathbf{a}$ which is 5-dimensional, and n *independent* inner minimizations over $\mathtt{M}_j$ which is 3-dimensional.

3.5 Imposing the Zero-Determinant Constraint to Refine the Initial Motion Estimate

The two problems described in Sect. 3.2 and Sect. 3.4 are both highly nonlinear, and their solutions are very sensitive to the initial guess. The initial guess is obtained by projecting the essential parameters onto the five-dimensional motion space. Unfortunately, the estimation of the essential parameters as described in Sect. 3.1 is very sensitive to data noise, especially when motion is small and the space points are close to a degenerate surface such as a plane. One major reason is that we have ignored the constraints which exist on the essential parameters and have used 8 parameters instead of 5 (i.e., three redundant parameters). Here, we propose to impose the zero-determinant (rank-2) constraint, and to estimate 7 parameters before projecting onto the motion space.

Indeed, there are only 7 independent parameters in a rank-2 matrix defined up to a scale factor (the scale factor and the rank-2 constraint remove two free parameters), and the fundamental matrix in the context of two uncalibrated images[31] has exactly the same properties. There are several possible parameterizations for such a matrix, e.g., one can express one row (or column) of the fundamental matrix as the linear combination of the other two rows (or columns). The following parameterization

$$\mathbf{F} = \begin{bmatrix} a & b & -ax_1 - by_1 \\ c & d & -cx_1 - dy_1 \\ -ax_2 - cy_2 & -bx_2 - dy_2 & (ax_1 + by_1)x_2 + (cx_1 + dy_1)y_2 \end{bmatrix} \tag{12}$$

expresses of course a matrix of rank 2, because both the third row and column are the combinations of the other two rows and columns. Furthermore, there is a nice geometric interpretation. The parameters (x_1, y_1) and (x_2, y_2) are the coordinates of the two epipoles $\mathbf{e}_1$ (projection of the optical center of the second camera in the first camera) and $\mathbf{e}_2$ (projection of the optical center of the first camera in the second camera) (see Fig. 1). The remaining four parameters (a, b, c, d) define the relationship between the orientations of the two pencils of epipolar lines. To take into account the fact that the matrix is defined only up to a scale factor, the matrix is normalized by dividing the four elements (a, b, c, d) by the largest in absolute value.

The seven parameters, $(x_1, y_1, x_2, y_2$, and three among $a, b, c,\ d)$, are estimated by minimizing the sum of distances between points and their epipolar lines. That is, we minimize the same objective function as the one (8) described in Sect. 3.2, except that the minimization is conducted over the above *7-dimensional parameter space*, instead of the five-dimensional motion space. The minimization is nonlinear, and we use the matrix estimated in (7) as the initial guess. The determinant of that matrix, denoted by $\mathbf{M}$ for clarity, is in general not equal to zero. We use the following technique to compute the initial guess. Let

$$\mathbf{M} = \mathbf{U}\mathbf{S}\mathbf{V}^T$$

be the singular value decomposition of matrix $\mathbf{M}$, where $\mathbf{S} = \operatorname{diag}(s_1, s_2, s_3)$ is a diagonal matrix satisfying $s_1 \geq s_2 \geq s_3 \geq 0$ (s_i is the i^{th} singular value), and $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. Then, it can be shown that

$$\mathbf{F} = \mathbf{U}\hat{\mathbf{S}}\mathbf{V}^T \tag{13}$$

with $\hat{\mathbf{S}} = \operatorname{diag}(s_1, s_2, 0)$ is the matrix of rank-2 which minimizes the Frobenius norm of $\mathbf{M} - \mathbf{F}$.[38] It is easy to verify that

$$\mathbf{F}\widetilde{\mathbf{e}}_1 = \mathbf{0} \quad \text{and} \quad \mathbf{F}^T\widetilde{\mathbf{e}}_2 = \mathbf{0}\ . \tag{14}$$

Therefore, $\widetilde{\mathbf{e}}_1 = [e_{11}, e_{12}, e_{13}]^T$ and $\widetilde{\mathbf{e}}_2 = [e_{21}, e_{22}, e_{23}]^T$ are equal to the last column of $\mathbf{V}$ and $\mathbf{U}$, respectively. From them, we have

$$x_i = e_{i1}/e_{i3} \quad \text{and} \quad y_i = e_{i2}/e_{i3} \qquad \text{for } i = 1, 2.$$

In turn, the four remaining elements (a, b, c, d) can be computed from $\mathbf{F}$.

Note that this intermediate stage can be applied to normalized image coordinates as well as pixel image coordinates, because both the determinant of $\mathbf{E}$ and that of $\mathbf{F}$ should be equal to 0.

3.6 Summary of the New Multistage Algorithm

We now summarize the main steps of our multistage algorithm:

Step 1: Estimate the essential parameters with 8-point algorithm (7). The obtained matrix is denoted by $\mathbf{E}_1$.

Step 2: Estimate a rank-2 matrix, denoted by $\mathbf{E}_2$, from $\mathbf{E}_1$ using (13), and compute the seven parameters from $\mathbf{E}_2$.

Step 3: Refine the seven parameters by minimizing the sum of squared distances between points and their epipolar lines, i.e., the objective function (8). The obtained matrix is denoted by $\mathbf{E}_3$. The zero-determinant constraint is satisfied.

Step 4: Estimate the motion parameters $\mathbf{t}$ and $\mathbf{R}$ from $\mathbf{E}_3$ (see e.g., [9] for the details).

Step 5: Refine the motion parameters by minimizing the sum of squared distances between points and their epipolar lines, i.e., the objective function (8).

Step 6: Reconstruct the corresponding 3D points as described in Sect. 3.3.

Step 7: Refine the motion and structure estimate by using the maximum likelihood criterion (11).

If we bypass steps 2 and 3, we have a standard 2-stage algorithm. The nonlinear minimization in steps 3, 5, 6, and 7 is done with the Levenberg-Marquardt algorithm implemented in the `Minpack` library.[37]

The uncertainty in motion and structure parameters can also be estimated, which is described in.[29]

4 Experimental Results

In this section, we first describe our Monte-Carlo simulations to show that our new multistage algorithm yields much more reliable results than the standard one when the level of noise in data points is high or when data points are located close to a degenerate configuration. We then present a set of real data with which the standard algorithm does not work while ours does. In,[29] we provide another set of Monte-Carlo simulations and a set of real data which show that better results can be obtained if we work directly with pixel coordinates rather than normalized image coordinates, because points are usually extracted from pixel images.

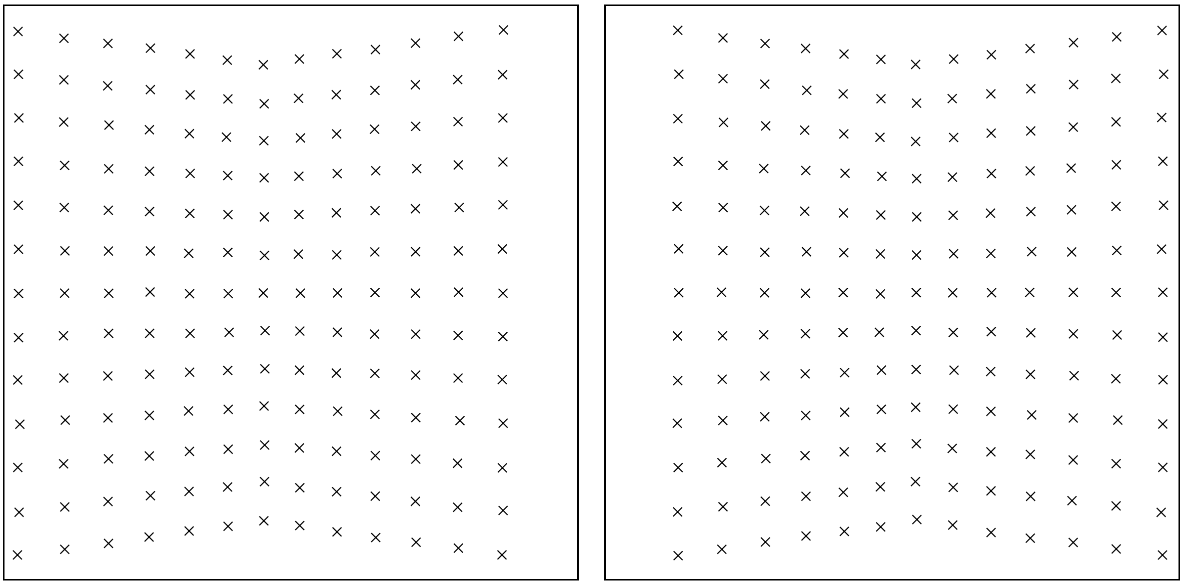


Figure 2: Images of two planar grids hinged together with $\theta = 45°$. Gaussian noise of $\sigma = 0.5$ has been added to each grid point

4.1 Computer Simulated Data

We use the same configuration as that described in.[30] The object is composed of two planar grids which are hinged together with angle $\pi - \theta$. When $\theta = 0$, the object is planar, which is a degenerate configuration for the algorithms considered in this paper. Each grid is of size 180×360 units^2. The object is placed in the scene with a distance of 530 units from the camera. The two images have the same intrinsic parameters: $\alpha_u = \alpha_v = 600$, $u_0 = v_0 = 255$, and $c = 0$. They differ by a pure lateral translation: $\mathbf{t} = [-40, 0, 0]^T$, and $\mathbf{R} = \mathbf{I}$. Small lateral motion is difficult for motion estimation because rotation and translation can be confused. The grid points are used as feature points. The x- and y-coordinates of each grid point are perturbed by independent random Gaussian noise of mean 0 and standard deviation σ pixels.

A pair of images with $\theta = 45°$ and $\sigma = 0.5$ pixels is shown in Fig. 2. The motion estimate given by the 2-stage algorithm is: $\mathbf{r} = [-7.981238e-05, -7.961793e-02, 3.779707e-04]^T$ (in

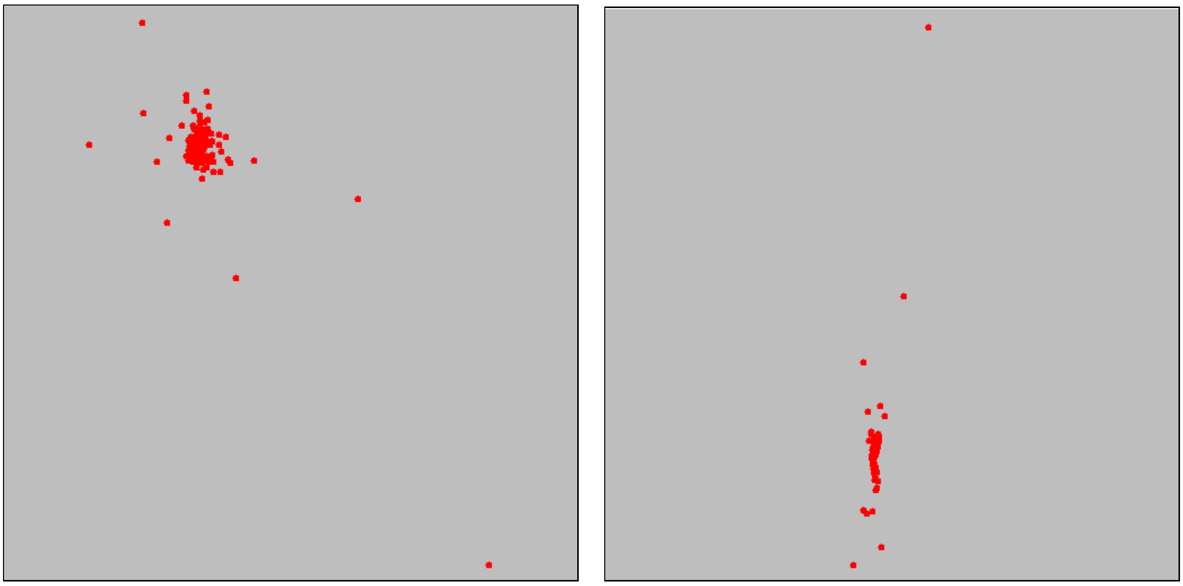


Figure 3: 3D reconstruction of the images shown in Fig. 2 with the 2-stage algorithm: Front and top views

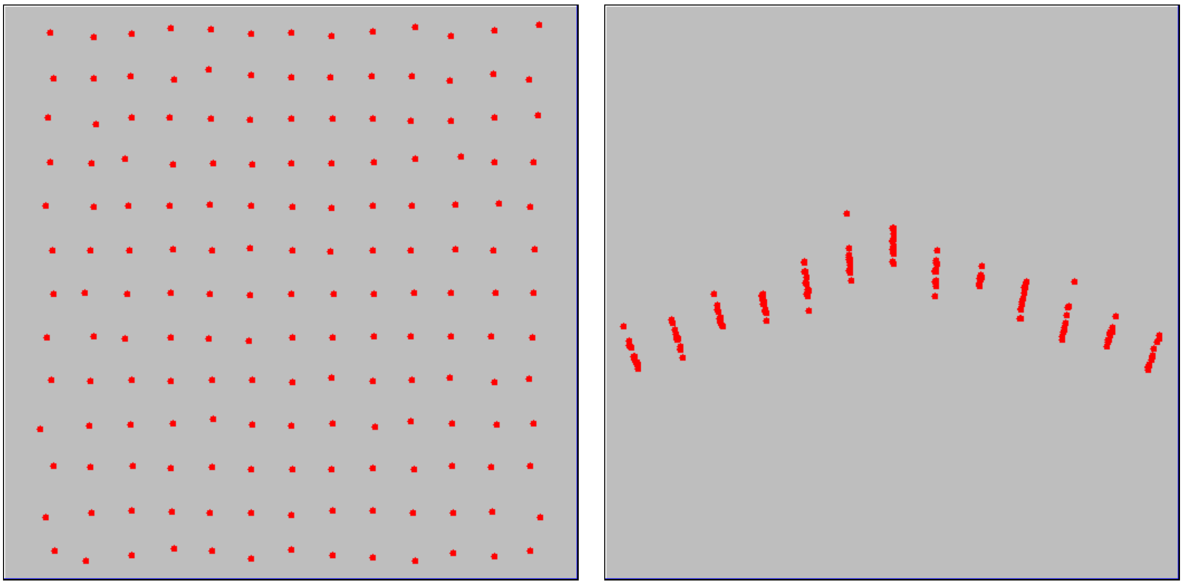


Figure 4: 3D reconstruction of the images shown in Fig. 2 with the 3-stage algorithm: Front and top views

radians) for rotation (which should be $[0, 0, 0]^T$), and $\mathbf{t} = [2.091024e - 01 - 3.089894e - 02 - 9.774055e - 01]^T$ for translation (which should be $[-1, 0, 0]^T$). The corresponding 3D reconstruction is shown in Fig. 3. Clearly, the 2-stage algorithm fails. When we apply the 3-stage algorithm to the same data, we obtain $\mathbf{r} = [-6.969567e{-}04, -1.785441e{-}03, -2.330740e{-}04]^T$ for rotation, and $\mathbf{t} = [-9.999643e - 01, -8.318301e - 03, 1.468760e - 03]^T$ for translation. The corresponding 3D reconstruction is shown in Fig. 4. As can be observed, quite reasonable result has been obtained with our new multistage algorithm, taking into account the fact that

Table 1: Comparison of the number of success trials out of 100 between the standard 2-stage (2S) algorithm and our new multistage (3S) algorithm

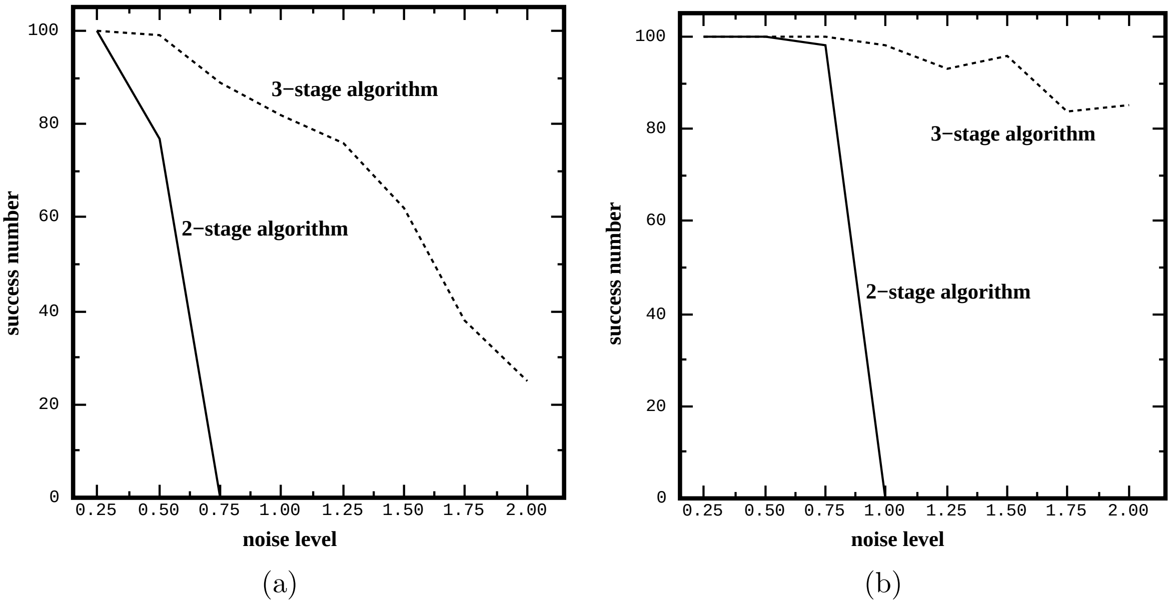
$\sigma =$	0.25		0.5		0.75		1.0		1.25		1.5		1.75		2.0	
θ	2S	3S	2S	3S	2S	3S	2S	3S	2S	3S	2S	3S	2S	3S	2S	3S
10	0	23	0	1	0	0	0	0	0	0	0	1	0	0	0	0
20	0	77	0	29	0	6	0	3	0	0	0	0	0	0	0	0
30	12	91	0	66	0	36	0	7	0	6	0	2	0	0	0	1
40	100	100	0	87	0	60	0	33	0	18	0	10	0	5	0	3
50	100	100	2	94	0	83	0	72	0	50	0	32	0	15	0	12
60	100	100	77	99	0	89	0	82	0	76	0	62	0	38	0	25
70	100	100	100	100	2	95	0	91	0	90	0	78	0	57	0	41
80	100	100	100	100	47	99	0	94	0	91	0	86	0	82	0	69
90	100	100	100	100	98	100	0	98	0	93	0	96	0	84	0	85

the object is close to a plane surface.

Now we provide more systematic and statistic results. We vary the angle θ from 10° to 90° with an interval of 10°. We also vary the level of the Gaussian noise added to the grid points. The standard deviation σ varies from 0.25 pixels to 2.0 pixels with an interval of 0.25 pixels. For each θ and each σ, we add 100 times independent noise to the grid points. For each set of noisy data, we apply the 2-stage algorithm and our multistage algorithm. If the estimated translation vector and the true one (i.e., $[-1, 0, 0]^T$) form an angle larger than 45°, then the algorithm is considered to have failed for this set of data. Among 100 trials for each θ and each σ, we count the number of times that the algorithm succeeds. The result is shown in Table 1. For a more direct perception of the difference of the two algorithms in performance, we show in Fig. 5 the curves of the number of successes with respect to various noise levels when θ is fixed at 60° and 90°, respectively. In Fig. 6, we show the curves with respect to various angles θ when the noise level σ is fixed at 0.5 pixels. A general rule is that the number of success decreases when the angle θ approaches to 0° and when the noise level σ increases. In all cases, our new multistage algorithm outperforms the 2-stage algorithm. The 3-stage algorithm gives much more reliable motion and structure estimate when the points are close to a planar surface (a degenerate configuration for the motion algorithms considered here) and when data points are heavily corrupted by noise. A final point is that the two algorithms give the same result when they both converge to the true solution. This is not surprising because the same optimization criterion is used in the last stage.

4.2 Real Data

In Fig. 7, we show a real image pair taken in a rock scene. This is a difficult set of data because the scene is quite flat except at the upper right corner where there is a big rock. We have overlayed on the images the point matches automatically established by the techniques described in[13] and Appendix A. When the 2-stage algorithm applies to this set of matches, the motion estimate is $\mathbf{r} = [-2.566072e-02, 1.801078e-03, 2.640767e-02]^T$ for rotation, and $\mathbf{t} = [1.986539e-02, 3.913866e-02, -9.990363e-01]^T$ for translation. We do not have the ground truth for this image pair, but since the reconstructed scene is not meaningful (and is thus not shown), we can say that the 2-stage algorithm fails.



(a)

(b)

Figure 5: Comparison of the standard and new algorithms for (a) $\theta = 60°$and (b) $\theta = 90°$with respect to noise level

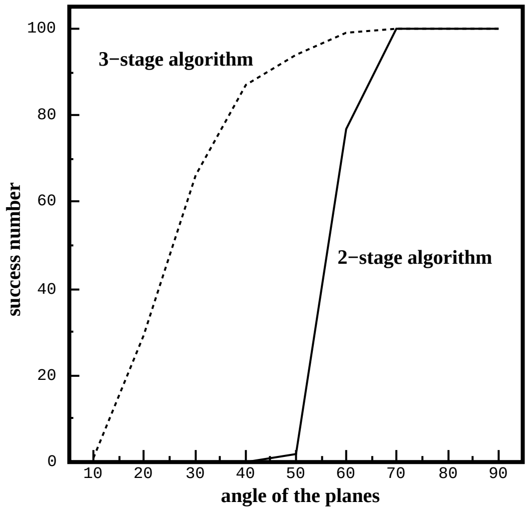


Figure 6: Comparison of the standard and new algorithms for $\sigma = 0.5$ pixels with respect to the angle θ of the object

When our multistage algorithm applies to the same set of matches, the motion estimate is $\mathbf{r} = [-2.101698e-02, 4.207611e-02, 1.042726e-02]^T$ for rotation, and $\mathbf{t} = [-9.667039e-01, 1.859049e-01, -1.758490e-01]^T$ for translation. The corresponding reconstructed 3D points are shown in Fig. 8. For a reader who knows how to do cross-eye fusion, we have generated a pair of stereograms shown in Fig. 9, where points have been linked based on Delaunay triangulation. Qualitatively, the result is quite good.

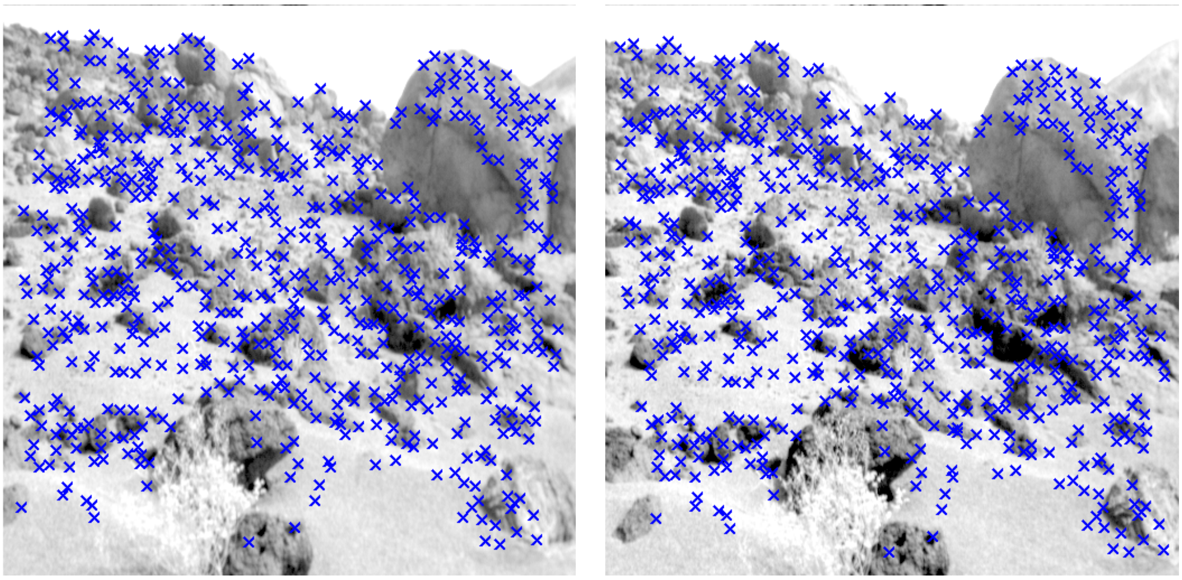


Figure 7: Two images of a rock scene

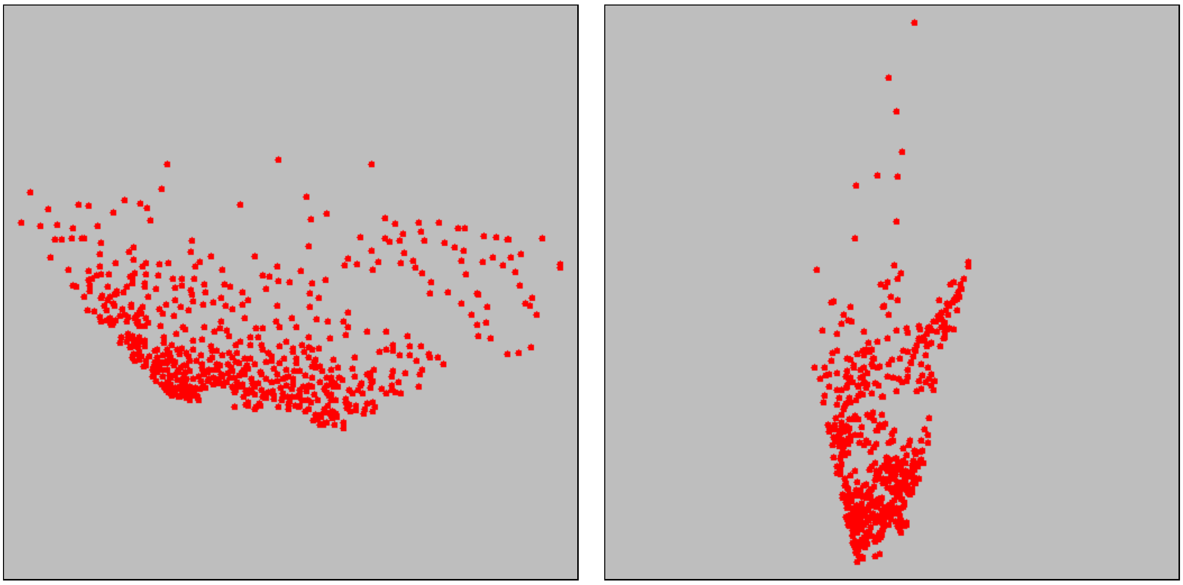


Figure 8: Reconstructed 3D points of the rock scene: Front and top views

5 Conclusions

In this paper, we have actually proposed a general scheme of 3D motion estimation from multiple calibrated images. Instead of projecting directly the essential parameters (defined in 8-dimensional space) onto the motion parameter space (which is 5-dimensional), we consider the estimation of a rank-2 matrix defined up to a scale factor (i.e., fundamental matrix, which is defined in 7-dimensional space) as an intermediate step to determine 3D Euclidean motion and structure. The proposed approach has been tested with synthetic and real data, and considerable improvement has been observed for the delicate situations such as heavily

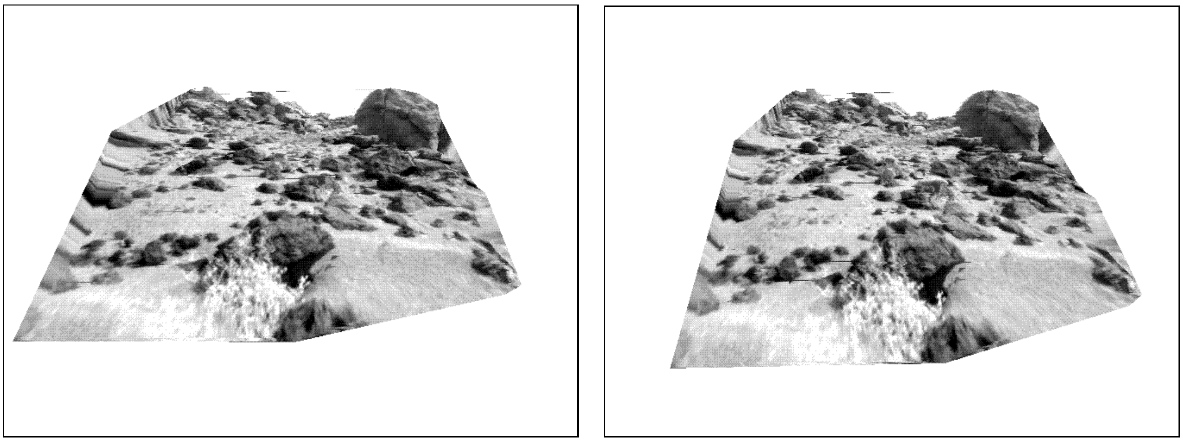


Figure 9: Stereogram of the reconstructed 3D points of the rock scene

noisy data and near-degenerate data. Note that the constraints we used in the intermediate stage are actually *all* constraints existing between two sets of image points if the images are *uncalibrated* (the intrinsic parameters are not know). That is, we are determining *projective* motion and structure in this stage. Our conjecture from this work is that the imposition of the constraints arising from projective geometry should be used as an intermediate step in order to obtain reliable 3D Euclidean motion and structure estimation from multiple calibrated images.

We have also shown in[29] through Monte-Carlo simulations and a set of real data that better results, especially in translation, have been obtained when pixel image coordinates are used. This is not surprising because independent and identically distributed noise has been added to pixel coordinates in simulation; and in practice, points are usually extracted from pixel images.

We have performed a direct projection of a 7-dimensional space to a 5-dimensional one. It would be interesting to know whether there exists a 6-dimensional space which bridges the fundamental matrix and the essential matrix.

A Detection of false matches

Given two images, we need to establish some pixel correspondences between them before computing motion and structure. The only geometric constraint we know between two images of a single scene is the *epipolar constraint*. However, when the motion between the two images is not known, the epipolar geometry is also unknown. The methods reported in the literature all exploit some heuristics in one form or another, for example, intensity similarity, which are not applicable to most cases. In,[39] we have developed an automatic and robust technique for matching two *uncalibrated* images. It consists of the following steps:

- extract high curvature points from each image using a corner detector;
- find match candidates for each high curvature point based on the normalized cross correlation;
- disambiguate matches through fuzzy relaxation by using neighboring information;

- detect false matches with the least-median-squares technique based on a criterion defined by the distances between points and their epipolar lines.

In that work, the intrinsic parameters of the images are not known, and we have to use the fundamental matrix. This is not the case for the problem described in this paper, and indeed we have more information available. In the following, we present the adaptation of the previously developed robust technique to solve the problem at hand.[13]

In all matches established by some heuristic technique, we may find two types of *outliers* due to

bad locations. In the estimation of the motion, the location error of a point of interest is assumed to exhibit Gaussian behavior. This assumption is reasonable since the error in localization for most points of interest is small (within one or two pixels), but a few points are possibly incorrectly localized (more than three pixels). The latter points will severely degrade the accuracy of the estimation.

false matches. In the establishment of correspondences, only heuristics have been used. Because the only geometric constraint, i.e., the epipolar constraint, is not yet available, many matches are possibly false. These will completely spoil the estimation process, and the final estimate of the motion will be useless.

The outliers will severely affect the precision of the motion if we directly apply the methods described above. In the following, we give a brief description of the two most popular robust methods: the *M-estimators* and the *least-median-of-squares* (LMedS) method.

Let r_i be the *residual* of the i^{th} datum, i.e., the difference between the i^{th} observation and its fitted value. The standard least-squares method tries to minimize $\sum_i r_i^2$, which is unstable if there are outliers present in the data. The M-estimators replace the squared residuals r_i^2 by another functions of the residuals, yielding

$$\min \sum_i \rho(r_i) \ ,$$

where ρ is a symmetric, positive-definite function with a unique minimum at zero. For example, Huber[40] employed the squared error for small residuals and the absolute error for large residuals. The M-estimators can be implemented as a weighted least-squares problem. That is, we use $\rho(r_i) = w_i r_i^2$ with small w_i for large r_i.

The LMedS method estimates the parameters by solving the nonlinear minimization problem:

$$\min \operatorname*{median}_i r_i^2 \ .$$

That is, the estimator must yield the smallest value for the median of squared residuals computed for the entire data set. It turns out that this method is very robust to false matches as well as outliers due to bad localization. Unlike the M-estimators, however, the LMedS problem cannot be reduced to a weighted least-squares problem. It is probably impossible to write down a straightforward formula for the LMedS estimator. It must be solved by a search in the space of possible estimates generated from the data. Since this space is too large, only a randomly chosen subset of data can be analyzed. The algorithm which we have implemented for robustly estimating the motion follows that structured in [41, Chap. 5], as outlined below.

Given n point correspondences: $\{(\mathbf{m}_{1i}, \mathbf{m}_{2i})\}$. A Monte Carlo type technique is used to draw m random subsamples of p different point correspondences. For the problem at hand, we select *seven* (i.e., $p = 7$) point matches for the reason to be clearer later. For each subsample, indexed by J, we use the techniques described in Sect. 3.1 to compute the motion $(\mathbf{R}, \mathbf{t})$, which, in turn, determines the fundamental matrix $\mathbf{F}_J$. For each $\mathbf{F}_J$, we can determine the median of the squared residuals, denoted by M_J, with respect to the whole set of point correspondences, i.e.,

$$M_J = \operatorname*{median}_{i=1,\ldots,n}[d^2(\widetilde{\mathbf{m}}_{2i}, \mathbf{F}_J\widetilde{\mathbf{m}}_{1i}) + d^2(\widetilde{\mathbf{m}}_{1i}, \mathbf{F}_J^T\widetilde{\mathbf{m}}_{2i})] \ .$$

We retain the estimate $\mathbf{F}_J$ for which M_J is minimal among all m M_J's. The question now is: *How do we determine m?* A subsample is "good" if it consists of p good correspondences. Assuming that the whole set of correspondences may contain up to a fraction ε of outliers, the probability that at least one of the m subsamples is good is given by

$$P = 1 - [1 - (1 - \varepsilon)^p]^m \ . \tag{15}$$

By requiring that P must be near 1, one can determine m for given values of p and ε. In our implementation, we assume $\varepsilon = 40\%$ and require $P = 0.99$, thus $m = 163$. Note that the algorithm can be speeded up considerably by means of parallel computing, because the processing for each subsample can be done independently.

As described in Sect. 3.1, five correspondences are theoretically sufficient, but we may have at most ten solutions and the algorithm is quite complex.[7] It will take time to compute the solutions and we need to check each solution against the whole set of data. If $p \geq 8$, a simple linear algorithm is available, however, for the same ε and P, the number of tries (i.e., m) is much higher than with a smaller p, following (15). For example, if $p = 8$, then $m = 272$ for $\varepsilon = 40\%$ and $P = 0.99$, almost doubled compared with that with $p = 7$. This is because we decrease the probability to have a good subsample when increasing the number of matches in each subsample. We think that $p = 7$ is a good trade-off.

As noted in,[41] the LMedS *efficiency* is poor in the presence of Gaussian noise. The efficiency of a method is defined as the ratio between the lowest achievable variance for the estimated parameters and the actual variance provided by the given method. To compensate for this deficiency, we further carry out a weighted least-squares procedure. The *robust standard deviation* estimate is given by

$$\hat{\sigma} = 1.4826[1 + 5/(n - p)]\sqrt{M_J} \ ,$$

where M_J is the minimal median. The constant 1.4826 is a coefficient to achieve the same efficiency as a least-squares in the presence of only Gaussian noise; $5/(n-p)$ is to compensate the effect of a small set of data. The reader is referred to [41, page 202] for the details of these magic numbers. Based on $\hat{\sigma}$, we can assign a weight for each correspondence:

$$w_i = \begin{cases} 1 & \text{if } r_i^2 \leq (2.5\hat{\sigma})^2 \\ 0 & \text{otherwise} \ , \end{cases}$$

where $r_i^2 = d^2(\widetilde{\mathbf{m}}_{2i}, \mathbf{F}\widetilde{\mathbf{m}}_{1i}) + d^2(\widetilde{\mathbf{m}}_{1i}, \mathbf{F}^T\widetilde{\mathbf{m}}_{2i})$. The correspondences having $w_i = 0$ are outliers and should not be further taken into account. The motion $(\mathbf{R}, \mathbf{t})$ is finally estimated by solving the weighted least-squares problem:

$$\min_{\mathbf{R},\mathbf{t}} \sum_i w_i r_i^2$$

using the technique described in Sect. 3.2. We have thus robustly estimated the motion because outliers have been detected and discarded by the LMedS method.

As said previously, computational efficiency of the LMedS method can be achieved by applying a Monte-Carlo type technique. However, the seven points of a subsample thus generated may be very close to each other. Such a situation should be avoided because the estimation of the motion from such points is highly instable and the result is useless. It is a waste of time to evaluate such a subsample. In order to achieve higher stability and efficiency, we have developed a *regularly random selection method* based on bucketing techniques. See[29] for the details.

Acknowledgment

The idea in this work was emerging during the discussions with Mi-Suen Lee of USC when she visited INRIA in 1995. Thanks go to Rachid Deriche and Andrew Zisserman for comments, and Kenichi Kanatani for discussions on statistical estimation. The author thanks the reviewers for their constructive comments.

References

[1] H. Nagel, “Image sequences - ten (octal) years- from phenomenology towards a theoretical foundation,” in *Proceedings 8th ICPR*, J.-C. Simon and J.-P. Haton, ed. (IEEE, Paris, France, 1986), pages 1174–1185.

[2] J.K. Aggarwal and N. Nandhakumar, “On the computation of motion from sequences of images — a review,” *Proc. IEEE*, 76(8):917–935, August 1988.

[3] T.S. Huang and A.N. Netravali, “Motion and structure from feature correspondences: A review,” *Proc. IEEE*, 82(2):252–268, February 1994.

[4] Z. Zhang, “Estimating motion and structure from correspondences of line segments between two perspective images,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12):1129-1139, 1995.

[5] H.C. Longuet-Higgins, “A computer algorithm for reconstructing a scene from two projections,” *Nature*, 293:133–135, 1981.

[6] R.Y. Tsai and T.S. Huang, “Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surface,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(1):13–26, January 1984.

[7] Olivier Faugeras and Steve Maybank, “Motion from point matches: multiplicity of solutions,” *The International Journal of Computer Vision*, 4(3):225–246, 1990.

[8] S.J. Maybank, *Theory of reconstruction From Image Motion*, (Springer-Verlag, 1992)

[9] Olivier Faugeras, *Three-Dimensional Computer Vision: a Geometric Viewpoint*, (MIT Press, 1993)

[10] M.E. Spetsakis and Y. Aloimonos, “Optimal visual motion estimation: A note,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(9):959–964, September 1992.

[11] J. Weng, N. Ahuja, and T.S. Huang, “Optimal motion and structure estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):864–884, September 1993.

[12] K. Kanatani, *Statistical Optimization for Geometric Computation: Theory and Practice*, (Elsevier, Amsterdam, 1996)

[13] Z. Zhang, “An automatic and robust algorithm for determining motion and structure from two perspective images,” in *Proceedings of 6th International Conference on Computer Analysis of Images and Patterns (CAIP’95)*, V. Hlavac and R. Sara, ed. pages 174–181, September 1995.

[14] Philip Torr, *Motion Segmentation and Outlier Detection.* PhD thesis, Department of Engineering Science, University of Oxford, 1995.

[15] Shimon Ullman, *The Interpretation of Visual Motion.* (MIT Press, 1979)

[16] T.S. Huang and C.H. Lee, “Motion and structure from orthographic projections,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:536–540, 1989.

[17] Jan J. Koenderink and Andrea J. van Doorn, “Affine structure from motion,” *Journal of the Optical Society of America*, A8:377–385, 1991.

[18] L.S. Shapiro, A. Zisserman, and M. Brady, “3D motion recovery via affine epipolar geometry,” *The International Journal of Computer Vision*, 16:147–182, 1995.

[19] M.D. Pritt, “Structure and motion from two orthographic views,” *Journal of the Optical Society of America A*, 13(5):916–921, May 1996.

[20] T.J. Broida, S. Chandrashekhar, and R. Chellappa, “Recursive 3-D motion estimation from a monocular image sequence,” *IEEE Trans. AES*, 26(4):639–656, July 1990.

[21] Z. Zhang and O. D. Faugeras, “Motion and structure from motion from a long monocular sequence,” in *Progress in Image Analysis and Processing II*, V. Cantoni, M. Ferretti, S. Levialdi, R. Negrini, and R. Stefanelli, ed. (World Scientific, Singapore, 1991) pages 264–271.

[22] C. Tomasi and T. Kanade, “Shape and motion from image streams under orthography: a factorization method,” *The International Journal of Computer Vision*, 9(2):137–154, 1992.

[23] J. Oliensis and J.I. Thomas, “Incorporating motion error in multi-frame structure from motion,” in *Proceedings of the IEEE Workshop on Visual Motion*, T.S. Huang, P.J. Burt, and E.H. Adelson, ed. (IEEE Computer Society Press, 1991), pages 8–13.

[24] R. Szeliski and S.B. Kang, “Recovering 3D shape and motion from image streams using nonlinear least squares,” *Journal Vis. Commun. and Image Repr.*, 5(1):10–28, 1994.

[25] S. Soatto, R. Frezza, and P. Perona, “Motion estimation on the essential manifold,” in *Proceedings of the 3rd European Conference on Computer Vision*, volume II of *Lecture Notes in Computer Science*, J-O. Eklundh, ed. (Springer-Verlag, 1994) pages 61–72.

[26] M. Lee, G. Medioni, and R. Deriche, “Structure and motion from a sparse set of views,” in *IEEE International Symposium on Computer Vision*, Biltmore Hotel, Coral Gables , Florida , USA, November 1995.

[27] C. Braccini, G. Gambardella, A. Grattarola, and S. Zappatore, “Motion estimation of rigid bodies: Effects of the rigidity constraints,” in *Proc. EUSIPCO, Signal Processing III: Theories and Applications*, L. Torres, E. Masgrau, and M.A. Lagunas, ed. pages 645–648, September 1986.

[28] J. Oliensis and V. Govindu, *Experimental evaluation of projective reconstruction in structure from motion*, Technical report, NEC Research Institute, Princeton, NJ 08540, USA, October 1995.

[29] Z. Zhang, *A new multistage approach to motion and structure estimation: From essential parameters to euclidean motion via fundamental matrix*, Research Report 2910, INRIA Sophia-Antipolis, France, June 1996.

[30] K. Kanatani, “Automatic singularity test for motion analysis by an information criterion,” in *Proceedings of the 4th European Conference on Computer Vision*, B. Buxton, ed. pages 697–708, Cambridge, UK, April 1996.

[31] Q.-T. Luong and O.D. Faugeras, “The fundamental matrix: Theory, algorithms and stability analysis,” *The International Journal of Computer Vision*, 1(17):43–76, January 1996.

[32] Olivier Faugeras, “Stratification of 3-D vision: projective, affine, and metric representations,” *Journal of the Optical Society of America A*, 12(3):465–484, March 1995.

[33] T.S. Huang and O.D. Faugeras, “Some properties of the E matrix in two-view motion estimation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(12):1310–1312, December 1989.

[34] B.K.P. Horn, “Motion fields are hardly ever ambiguous,” *The International Journal of Computer Vision*, 1(3):263–278, 1987.

[35] H.C. Longuet-Higgins, “Multiple interpretations of a pair of images of a surface,” *Proc. Roy. Soc. Lond. A.*, 418:1–15, 1988.

[36] C.-H. Lee, “Time-varying images: The effect of finite resolution on uniqueness,” *CVGIP: Image Understanding*, 54(3):325–332, 1991.

[37] J.J. More, “The Levenberg-Marquardt algorithm, implementation and theory,” in *Numerical Analysis*, G. A. Watson, ed., Lecture Notes in Mathematics 630. Springer-Verlag, 1977.

[38] G.H. Golub and C.F. Van Loan, *Matrix computations*, 2nd ed., (The John Hopkins University Press, Baltimore, Maryland, 1989)

[39] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong, “A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry,” *Artificial Intelligence Journal*, 78:87–119, October 1995.

[40] P.J. Huber, *Robust Statistics*, (John Wiley & Sons, New York, 1981)

[41] P.J. Rousseeuw and A.M. Leroy, *Robust Regression and Outlier Detection*, (John Wiley & Sons, New York, 1987)