Motion of points and lines in the uncalibred case

Thierry Viéville, Quang-Tuan Luong

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Motion of Points and Lines in the Uncalibrated Case

Thierry VIÉVILLE
Quang-Tuan LUONG

N° 2054
Octobre 1993

PROGRAMME 4

Robotique, image
et
vision

Motion of points and lines in the uncalibrated case

Thierry Viéville and Quang-Tuan Luong[1]
INRIA, Sophia, BP93, 06902 Valbonne, France
vthierry@sophia.inria.fr

Abstract

In the present paper we address the problem of computing structure and motion, given a set point and/or line correspondences, in a monocular image sequence, when the camera is **not** calibrated. We first describe an algebraic approach of the problem, and then relate the different equations to a specific geometric construction which allows to solve the motion problem. An implementation is proposed and experimental results are provided.

Mouvement de points et de droites en l'absence de calibration

Abstract

Dans ce papier on s'attaque au problème de calculer le mouvement et la structure d'une scène, étant donnés un ensemble de correspondances entre points et/ou droites, au sein d'une séquence d'images, quand le système n'est **pas** calibré. Nous présentons d'abord une approche algébrique de ce problème et relions les différentes équations à une construction géométrique spécifique qui permet de résoudre le problème du mouvement. Une implémentation est proposée et des résultats expérimentaux sont fournis.

[1] University of California, ECCS, Cory Hall, 211-215, Berkeley, CA 94720

Contents

1 Introduction

In the present paper we address the problem of computing structure and motion, given a set point and/or line correspondences, in a monocular image sequence, when the camera is **not** calibrated.

Most of the earlier or recent studies in the field assume that the calibration of the system is known (a far from been exhaustive bibliography being [16, 15, 20, 9]), except for [6, 26, 25, 28]. However, these authors have studied only the case of points correspondences, and have restricted their approach to the case where the intrinsic parameters of the camera are constant, while only 2 or 3 views have been taken into account. The generalization to the case where points and lines are given has not been made (except [11] for some results), and the problem of considering non-constant intrinsic parameters has also not yet been addressed.

This is however an important challenge. In particular, in the case of active vision, the extrinsic parameters and the intrinsic parameters of the visual sensor are modified dynamically. For instance, when tuning the zoom and focus of a lens, these parameters are modified and must be recomputed at any time. It is thus relevant to attempt to determine dynamic calibration parameters by a simple observation of an unknown stationary scene, when performing a rigid motion.

The facility in a motion paradigm is that we can assume the disparity between two frames to be small, leading to easy solutions for the correspondence problem. These correspondences can efficiently been established (Token-Tracking) [2, 24]. The problem is thus, given correspondences between points or lines, to recover, the motion, structure and calibration of the system.

This paper is divided into three parts :

In the first section, we attack the problem from an algebraic point of view, deriving the equations related to the motion of lines and points, for a system of cameras without calibration, and obtain two different algorithms to estimate these parameters.

In the second section, we attack the problem from a geometric point of view, and generalize the epipolar geometry used for points to a new construction, called trifocal geometry, which allows to solve the motion problem in three views, considering lines or points.

In the final section, we propose an implementation and report some experimental results.

2 The motion of lines and points : algebraic approach

2.1 Setting the equations

Camera model and frame of reference. We use *the standard pinhole model* for a camera, in a position indexed by i, assuming the camera performs a perfect perspective transform with center C_i (the camera optic center) at a distance f (the focal length) of the retinal plane.

It must be noted, that the pinhole model can still be used for a zoom lens if the object-to-image distance is not considered as fixed [14]. This is the case here, since we will adapt the camera metric for different object locations.

All coordinates are related to an affine frame of reference $\mathcal{R}_i = (C_i, \mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$ *attached to the retina*, $\mathbf{z}_i$ being aligned with the optical axis, $\mathbf{x}_i$ and $\mathbf{y}_i$ being aligned with horizontal and vertical axis in the image. The retinal plane is thus perpendicular to the optical axis $C_i z_i$. All quantities attached to frame i are indexed by i also.

We represent a 3D-point $M_i = C_i\vec{M} = (X_i, Y_i, Z_i)^T$ using Euclidean coordinates. Points onto the retina, with horizontal and vertical coordinates (u_i, v_i) in pixel will be represented as homogeneous 3D vectors : $s\ m_i = s\ C_i\vec{m} = (s\ u_i, s\ v_i, s)^T$, corresponding to lines of a given direction passing through the optical center (2D projective space). Algebraic developments will thus involve 3D vectors in both cases [2]

[2]We note vectors using bold letters and matrix using capital letters. $\mathbf{x} \wedge \mathbf{y} = \tilde{\mathbf{x}} \cdot \mathbf{y}$ denotes the cross-product,

In this study, *we must not assume the system is calibrated.* On the contrary, we consider the following well established model for a camera, using a A-matrix :

$$Z_i\, m_i = \begin{vmatrix} Z_i\, u_i \\ Z_i\, v_i \\ Z_i \end{vmatrix} = \left(\underbrace{\begin{pmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{A_i} \quad \begin{matrix} 0 \\ 0 \\ 0 \end{matrix} \right) \cdot \begin{vmatrix} X_i \\ Y_i \\ Z_i \\ U \in \{0,1\} \end{vmatrix} = A_i \cdot M_i \qquad (1)$$

Please note that we assume that the intrinsic parameters are different for each camera position, although we have not indexed the A-matrix components for simplicity.

This corresponds, if $\gamma = 0$, to the usual equations $u = \alpha_u\, X/Z + u_0$, $v = \alpha_v\, Y/Z + v_0$. See [27] for a recent review. In particular $det(A_i) = \alpha_u\, \alpha_v = \frac{k}{f^2}$ is proportional to the inverse of the square of the focal length.

For points not at infinity, we have $U = 1$ and Z_i corresponds to the Euclidean depth with this model. Equation (1) is still valid for "points at infinity", i.e. vanishing points, with $U = 0$, but Z_i does not correspond anymore to the depth of the point.

Considering points and line as tokens. .

Point are represented by their 3D coordinates. The 2D information, i.e. what is measured in the image, corresponding to frame i, is m_i and the 3D information, i.e what must be estimated to recover the 3D Euclidean structure -knowing the calibration- is Z_i. Such quantities will also be called the "uncalibrated structure of the point", since we will see that it can be estimated without recovering the calibration parameters.

Lines are represented by their Plucker coordinates as follows. Let D be a 3D line. We consider the unary vector $\mathbf{L}_i$, with $||\mathbf{L}_i|| = 1$, which is aligned with the 3D line direction, and the vector :

$$\mathbf{N}_i = \vec{C_i M} \wedge \mathbf{L}_i \qquad (2)$$

for any point $M \in D$, as illustrated in figure 1. It is straightforward to verify that $\mathbf{N}_i$ is orthogonal to $\mathbf{L}_i$ and does not depend on M (see for instance [4]). Moreover, the magnitude of $\mathbf{N}_i$, $||\mathbf{N}_i||$, is equal to the distance from D to the origin C_i. Then, 3D lines are represented by two 3D-vectors $(\mathbf{L}_i, \mathbf{N}_i)$ with the two quadratic constraints $\mathbf{L}_i^T \cdot \mathbf{L}_i = 1$ and $\mathbf{L}_i^T \cdot \mathbf{N}_i = 0$, yielding a parameter space of dimension 4, as expected.

Given two points $M^a \in D$ and $M^b \in D$ with $M^a \neq M^b$ we have, up to an $\epsilon = \pm 1$:

$$\begin{cases} L_i &= \epsilon \frac{M^b - M^a}{||M^b - M^a||} \\ N_i &= \epsilon \frac{M^a \wedge M^b}{||M^b - M^a||} \end{cases} \qquad (3)$$

as the reader can easily verify.

A 3D line D projects onto a line d_i in the camera indexed by i. Let P_i be the projection plane containing D, the origin C_i and the projection d_i. Obviously $\mathbf{N}_i$ is normal to this plane.

We write the equation of a 2D line d_i, projection of a 3D line D, in the camera indexed by i :

$$m \in d_i \quad \Leftrightarrow \quad m^T \mathbf{n}_i = 0 \qquad (4)$$

the vector $\mathbf{n}_i$ is an homogeneous vector.

Moreover, to the 3D line direction corresponds a point at infinity, i.e a direction in the Euclidean space; let us write $\mathbf{l}_i$ the this point at infinity. Using equation (1), and the previous definitions, we immediately obtain :

$$\alpha_i\, A_i^T \cdot \mathbf{n}_i = \mathbf{N}_i \quad ; \quad \beta_i\, \mathbf{l}_i = A_i \cdot \mathbf{L}_i \qquad (5)$$

---

the dot-product being written as $\mathbf{x}^T \cdot \mathbf{y}$. The identity matrix is written I.

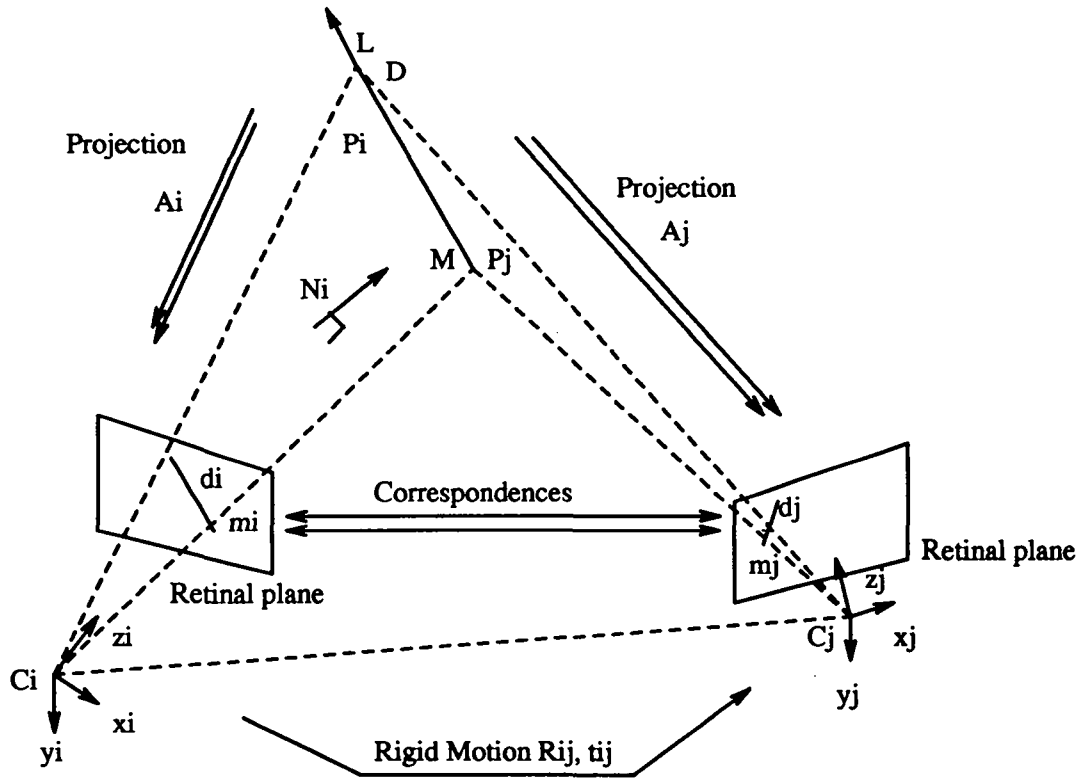

Figure 1: Elements used in the definition of lines in motion

We choose $\alpha_i = ||A_i^{-1\,T} \cdot \mathbf{N}_i||$ and $\beta_i = ||A_i \cdot \mathbf{L}_i||$ to have $\mathbf{l}_i$ and $\mathbf{n}_i$ unary. In fact, α_i and β_i are related to the "uncalibrated structure" of the 3D line. This means that these two parameters corresponds to quantities which are not measured in the image but must be estimated to recover the 3D structure of the line.

If m_i is a point on the 2D-line, the parametric representation of the line is very simply : $m_i + \lambda\, \mathbf{l}_i$ for $\lambda \in \mathcal{R}$.

Given two points m^a and m^b on the 2D line d_i, projection of two points $M^a \in D$ and $M^b \in D$ we have, from eq. (3) and (1), up to an $\epsilon = \pm 1$:

$$\begin{cases} l_i &= \underbrace{\dfrac{\epsilon}{||M^b - M^a||\,\beta_i}}_{\kappa_i} \left[Z^b\, m^b - Z^a\, m^a\right] \\ n_i &= \underbrace{\dfrac{\epsilon\, Z^a\, Z^b}{||M^b - M^a||\,\alpha_i\, det(A_i)}}_{\rho_i} \left[m^a \wedge m^b\right] \end{cases} \tag{6}$$

as the reader can easily verify again[3].

Representation of rigid displacements. We consider only the motion of a unique rigid object or the ego-motion of the camera observing a stationary scene, *in the discrete case.* We thus represent motion through rigid displacements.

[3]We make use of the following relation: $[M \cdot \mathbf{x}] \wedge [M \cdot \mathbf{y}] = \underbrace{det(M)\, M^{-1\,T}}_{M^*} \cdot [\mathbf{x} \wedge \mathbf{y}]$

It means that the tokens in the scene are undergoing a rigid displacement parameterized by a rotation matrix R_{ij} and a translation vector $\mathbf{t}_{ij}$:

$$M_i = R_{ij} \cdot M_j + \mathbf{t}_{ij} \tag{7}$$

This equation defines the 3D correspondence, for a point, between two frames. The reverse relation is $M_j = R_{ij}^T \cdot M_i + \mathbf{t}_{ji}$ and we obviously have $R_{ji} = R_{ij}^T$ and $\mathbf{t}_{ji} = -R_{ij} \cdot \mathbf{t}_{ij}$. Moreover rotations and translations form the well-known group of rigid displacements for which we have $R_{ij} = R_{ik} \cdot R_{kj}$ and $\mathbf{t}_{ij} = \mathbf{t}_{ik} + R_{ik} \cdot t_{kj}$, while $R_{ii} = I$ and $\mathbf{t}_{ii} = 0$.

Applying this equation to the Plucker representation of a 3D line allows to compute the 3D correspondence, for a line, between two frames. Combining equations (2) and (7) yields after a few algebra (see for instance [7]) :

$$\mathbf{L}_i = R_{ij} \cdot \mathbf{L}_j \quad ; \quad \mathbf{N}_i = R_{ij} \cdot \mathbf{N}_j + t_{ij} \wedge \mathbf{L}_i \tag{8}$$

This equation defines the 3D correspondence, for a line, between two frames.

2.2 Using the Qs-representation to analyze retinal correspondences.

Generality of the approach. Having a precise definition for calibration, tokens and motion, we have to relate these parameters to what is measured in an image sequence : the 2D correspondences, for points and lines. This is going to be done using a special representation, the Qs-representation.

But before doing this, let us emphasis why our model is very general : in most of the artificial vision problems or "paradigms", four quantities are to be manipulated (see figure 1) : the projection from the 3D world onto the camera, the correspondences between tokens in two cameras (either at different locations in space such as for stereo, or at different times such as for motion analysis), the rigid displacement (either in space or time) and the 3D-structure. We summarize, for different paradigms, what is given to the algorithm (input)[4] and what is estimated by the algorithm (output) :

Paradigm	Projection	Correspondences	Rigid Displacement	3D Structure
Structure from Motion	input	input	input	output
Stereo Matching	input	output	input	not used
Stereo Reconstruction	input	input	input	output
Pose determination	input	input	output	input
Motion Computation	input	input	output	not used
Token Tracking	not used	output	*expected small*	not used
Off-line Calibration	output	*input*[4]	input	input

In every algorithms, only one quantity is output, whereas all others are input or not used. However, we do not think this is the write way. On the contrary, these quantities are in deep interaction. We thus would like to treat the problem in the following manner :

Paradigm	Projection	Correspondences	Rigid Displacement	3D Structure
Motion Without Calibration	output	input	output	output

This corresponds to what is usually called "auto-calibration".

Definition and properties of the Qs-representation. Considering the 2D correspondences between two points m_i and m_j in two different frames, we obtain, combining equation (1) and

[4] In the case of "off-line calibration," a correspondence means a correspondence between the 2D point in one image and the 3D model used for calibration.

equation (7) :

$$Z_i\, m_i = Z_j \underbrace{A_i \cdot R_{ij} \cdot A_j^{-1}}_{Q_{ij}} \cdot m_j + \underbrace{A_i \cdot \mathbf{t}_{ij}}_{\mathbf{s}_{ij}} \tag{9}$$

- The quantity Q_{ij} corresponds to the "uncalibrated rotational component of the rigid displacement". In fact, this matrix corresponds to *the collineation of the plane at infinity from frame j to frame i*, as easily established considering any vanishing point, or remembering that for points at infinity (in practice at the horizon) the translational component of the motion is not to be taken into account (in practice is negligible). If the calibration parameters are constant, i.e. $A_i = A_j = A$ then $det(Q_{ij}) = 1$ but in any case $det(Q_{ij}) = \frac{det(A_i)}{det(A_j)} \simeq \left(\frac{f_j}{f_i}\right)^2$ is directly related to the variation of the focal length of the system, i.e. the "zoom".
  This matrix is in fact related to the affine geometry of the scene, because the displacements of 3D line directions, i.e. vanishing points, are entirely defined by Q_{ij}. For instance the reader can easily verify that two 3D lines are parallel if and only if their intersection in the frame j is mapped onto their intersection in frame i by Q_{ij}.
- The quantity $\mathbf{s}_{ij}$ corresponds to the "uncalibrated translational component of the rigid displacement", also called "focus of expansion" by some authors. In fact, this vector corresponds to the epipole in frame i, i.e. *the projection of the optical center of frame j onto the retinal plane R_i of frame i*, since it is the image of the quantity $m_j = (0,0,0)^T$. However, despite the fact it is in relation with an homogeneous quantity, the magnitude and direction of this vector is entirely specified.

Considering the previous definitions, and the relation between rotations and translations, we have the following obvious relations between collineations and epipoles :

$$\begin{array}{rcl} Q_{ii} = I & ; & \mathbf{s}_{ii} = 0 \\ Q_{ij} = Q_{ji}^{-1} & ; & \mathbf{s}_{ij} = -Q_{ij} \cdot \mathbf{s}_{ji} \\ Q_{ij} = Q_{ik} \cdot Q_{kj} & ; & \mathbf{s}_{ij} = \mathbf{s}_{ik} + Q_{ik} \cdot \mathbf{s}_{kj} \end{array} \tag{10}$$

Then, in an image sequence, as soon as the Qs-representation between two consecutive views is known, any other Qs-representation can be estimated.

This representation is defined up to two scale-factors as it is easily seen, when making explicit equation (7)[5] :

$$\left\{ \begin{array}{l} \left\{ \begin{array}{rcl} u_i & = & \dfrac{Q_{ij}^{00}\, u_j + Q_{ij}^{01}\, v_j + Q_{ij}^{02} + \frac{1}{Z_j}\, s_{ij}^0}{Q_{ij}^{20}\, u_j + Q_{ij}^{21}\, v_j + Q_{ij}^{22} + \frac{1}{Z_j}\, s_{ij}^2} \\ v_i & = & \dfrac{Q_{ij}^{10}\, u_j + Q_{ij}^{11}\, v_j + Q_{ij}^{12} + \frac{1}{Z_j}\, s_{ij}^1}{Q_{ij}^{20}\, u_j + Q_{ij}^{21}\, v_j + Q_{ij}^{22} + \frac{1}{Z_j}\, s_{ij}^2} \end{array} \right. \\ \dfrac{1}{Z_i} = \dfrac{1}{Z_j}\, \dfrac{1}{Q_{ij}^{20}\, u_j + Q_{ij}^{21}\, v_j + Q_{ij}^{22} + \frac{1}{Z_j}\, s_{ij}^2} \end{array} \right. \tag{11}$$

If we multiply every depth Z_i and both Q_{ij} and $\mathbf{s}_{ij}$ by a constant factor λ in these equations, the quantities are left unchanged. This "expansion" factor control the relative scale between frame i and frame j. This indetermination disappears if the determinant of Q_{ij} is known, that is if the variation of the focal length if known, because this expansion corresponds to the "relative zoom" between two frames. It is - a priori - not known if the calibration is not known.

In addition to that, if we multiply every depth Z_i, Z_j and $\mathbf{s}_{ij}$ by another constant factor μ in these equations, the quantities are also left unchanged. This "scale" factor corresponds to the usual scale factor indetermination observed in a monocular image sequence.

There are thus two kind of scale factors :

[5] We represent the components of a matrix or a vector using upper subscripts from 0 to 2.

- **Local** scale factors control relations between two consecutive frames and vary from one pair of frames to another. The "expansion" factor identified previously is one example.
- **Global** scale factors are unique for the whole sequence and correspond to a global attribute. The "scale" factor identified previously is one example.

Reciprocally, because these equations are linear with respect to Q and $\mathbf{s}$, these are the only transformations which leave these equations unchanged, considering a enumerable set of generic points. So, $9_{Q-components} + 3_{s-components} - 2_{scale-factors} = 10$ parameters if considering only two views, but when considering an image sequence, relating the depths between two consecutive views, only one scale factor remains, the other being global for the whole sequence.

These remarks can be summarized in the following proposition :

Proposition 1 *In a sequence of $N+1$ views, the 2D-correspondences between points are defined by 10 N parameters if the representation is local, considering correspondences between pairs of consecutive views only. When considering correspondences in the whole sequence, the 2D-correspondences are defined by 11 $N-1$ parameters.*

In order to avoid this indetermination we can choose different constraints, depending on the nature of the problem. For instance, if the intrinsic calibration parameters are constant, i.e. if $A_i = A_j$, we have $det(Q_{ij}) = 1$. Moreover, if we choose $||\mathbf{s}_{ij}|| = 1$ both indeterminations are avoided.

Let us compare with the Euclidean parameterization of the rigid motion. When the system is not calibrated, considering $N+1$ views, we have N set of parameters related to the motion between two consecutive views and $N+1$ set of parameters related to the calibration, but the motion is known up to a scale factor. So, the retinal displacement is a function of : $(3\,N)_{3D-rotation} + (3\,N)_{3D-translation} + (5\,(N+1))_{intrinsic\,parameters} - (1)_{scale\,factor} = 11\,N+4$ parameters. We then have more unknowns than for the Qs-representation, and it is generally not possible to recover the rigid motion of a set of points, when the system is not calibrated. This negative result can be stated as follows :

Proposition 2 *When studying the motion of points with an uncalibrated system having variable calibration parameters, we cannot recover the calibration and Euclidean motion 11 $N+4$ parameters from the observation of the retinal disparities unless some additional information is known.*

Relationship between the Qs-representation and the F-matrix. If we eliminate Z_i and Z_j in equation (9) we obtain :

$$m_i^T \underbrace{[\tilde{\mathbf{s}}_{ij} \cdot Q_{ij}]}_{F_{ij}} \cdot m_j = 0 \tag{12}$$

The matrix $F_{ij} = \tilde{\mathbf{s}}_{ij} \cdot Q_{ij}$ is also called the "essential matrix in the uncalibrated case" considering the original Longuet-Higgins equation [16], now generalized . There is a deep relationship between the Qs-representation and the fundamental matrix defined by [18, 19].

Let us analyze this relationship. A F-matrix is defined by 7 parameters (The matrix has 9 coefficients but is defined only up to 1 scale factor and is subject to 1 constraint $det(F_{ij}) = 0$). However, in our case we have entirely defined the F-matrix using (12) and with this definition the scale factor is known. It is however not measurable (local scale factor).

As for the Qs-representation, using equations (10), we have the following relations between

F-matrices in an image sequence :

$$\begin{array}{rcl} F_{ii} & = & 0 \\ F_{ij} & = & det(Q_{ij})\ F_{ji}^T \\ F_{ij} & = & det(Q_{ik}) \left[Q_{ki}^T \cdot F_{kj} + F_{ki}^T \cdot Q_{kj} \right] \\ & = & det(Q_{ik}) \left[Q_{ki}^T \cdot [\tilde{\mathbf{s}}_{kj} - \tilde{\mathbf{s}}_{ki}] \cdot Q_{kj} \right] \\ & = & det(Q_{ik}) \left[\frac{\tilde{\mathbf{s}}_{ik}}{det(Q_{ik})} \cdot Q_{ij} - Q_{ji}^T \cdot \frac{\tilde{\mathbf{s}}_{jk}}{det(Q_{jk})} \right] \end{array} \tag{13}$$

from which we deduce the fundamental relation :

$$\mathbf{s}_{ik}^T \cdot F_{ij} \cdot \mathbf{s}_{jk} = 0 \Leftrightarrow \mathbf{s}_{jk}^T \cdot F_{ji} \cdot \mathbf{s}_{ik} = 0 \tag{14}$$

since $\mathbf{s}_{ij} \wedge \mathbf{s}_{ik} = F_{ij} \cdot \mathbf{s}_{jk}$.

If we consider $N+1$ views we can define $\frac{N\,(N+1)}{2}$ F-matrices and $N\ (N+1)$ corresponding epipoles. However, considering equation (14) for any indices, we generate $\frac{N\,(N-1)\,(N-2)}{2}$ constraints, obviously not all independent.

Let us determine the exact number of parameters which determine a set of F-matrices in an image sequence.

In fact, since from equation (12) we observe that all F-matrices are generated by the Qs-representation we have, from Prop. 1 at most $11\ N-1$ parameters.

If we want to understand the relationships between the F-matrix and the Qs-representation, we need the following decomposition of a Q-matrix, given the corresponding s-vector :

$$Q_{ij} = S_{ij} + \mathbf{s}_{ij} \cdot \mathbf{r}_{ij}^T \quad \textbf{with} \quad \mathbf{s}_{ij}^T S_{ij} = 0 \quad \Leftrightarrow \quad \mathbf{r}_{ij} = \frac{Q_{ij}^T \cdot \mathbf{s}_{ij}}{||\mathbf{s}_{ij}||^2} \ ; \ S_{ij} = -\frac{\tilde{\mathbf{s}}_{ij}^2}{||\mathbf{s}_{ij}||^2} \cdot Q_{ij} \tag{15}$$

In this decomposition, the image of the s-vector through the transpose of the Q-matrix has been made explicit. In other words, the transpose of the Q-matrix is decomposed into its restriction to the vectorial line generated by $\mathbf{s}_{ij}$ and its restriction to $\mathbf{s}_{ij}^{\perp}$, represented by S_{ij}. The reader can easily verify, from (15), that this decomposition is one-to-one.

There is thus a one-to-one correspondence between F-matrices and S-matrices since we have :

$$F_{ij} = \tilde{\mathbf{s}}_{ij} \cdot S_{ij} \ ; \ \mathbf{s}_{ij}^T \cdot S_{ij} = 0 \Leftrightarrow S_{ij} = -\frac{\tilde{\mathbf{s}}_{ij}}{||\mathbf{s}_{ij}||^2} \cdot F_{ij} \ ; \ \mathbf{s}_{ij}^T \cdot F_{ij} = 0 \tag{16}$$

Moreover, although in this equation, both quantities are defined only up to scale factor, if we use the following two constraints : $||\mathbf{s}_{ij}||^2 = 1$ and $||S_{ij}||^2 = 1$, in addition to the 3 constraints given by $\mathbf{s}_{ij}^T \cdot S_{ij} = 0$, we have a $(3-1)_{s-vector} + (9-3-1)_{S-matrix} = 7$ parameters representation of F_{ij}, as expected [18]. A local parameterization of the F-matrix based on these relations will be proposed in a next section.

Knowing the S_{ij} matrix, we can estimate the related $\mathbf{s}_{ij}$ vectors since $\mathbf{s}_{ij}^T \cdot S_{ij} = 0$. We also can estimate $\mathbf{s}_{ji}$ and S_{ji} up to a scale factor, since from equation (10) we have :

$$\left\{ \begin{array}{rcl} 0 & = & S_{ij} \cdot \mathbf{s}_{ji} \\ S_{ji} & = & -det(Q_{ij})\ \frac{\tilde{\mathbf{s}}_{ji}}{||\mathbf{s}_{ji}||^2} \cdot S_{ij}^T \cdot \tilde{\mathbf{s}}_{ij} \end{array} \right.$$

If only two views are given there is no way to recover the Q-matrix from point correspondences, since only S_{ij} and $\mathbf{s}_{ij}$ are measurable, whereas $\mathbf{r}_{ij}$ is not. But considering S-matrices and s-vectors in an image sequence, we can relate all r-vectors by the following crucial relation,

obtained by making explicit the r-vectors in the formula $Q_{ij} = Q_{ik} \cdot Q_{kj}$ and using some useful relations between this elements[6] :

$$\mathbf{r}_{ij} = \mathbf{r}_{kj} + \mathbf{q}_{ijk} \quad ; \quad \mathbf{q}_{ijk} = \frac{(\mathbf{s}_{ik}^T \cdot \mathbf{s}_{ik})\, S_{kj}^T \cdot S_{ik}^T \cdot \mathbf{s}_{ij} + (\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ik})\, S_{ij}^T \cdot \mathbf{s}_{ik}}{(\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij})\,(\mathbf{s}_{ik}^T \cdot \mathbf{s}_{ik}) - (\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ik})^2} \tag{17}$$

Please note that this expression is coherent with respect to the different scale factors indeterminations as follow. If $Q_{ij} = S_{ij} + \mathbf{s}_{ij} \cdot \mathbf{r}_{ij}^T$ is defined up to a scale factor λ_{ij} and $\mathbf{s}_{ij}$ is defined up to a scale factor μ_{ij}, as discussed previously, S_{ij} is defined up to λ_{ij}, $\mathbf{r}_{ij}$ is defined up to $\frac{\lambda_{ij}}{\mu_{ij}}$ and F_{ij} is defined up to $\lambda_{ij}\,\mu_{ij}$. Introducing these scale factors in equation (17) it appears that $\mathbf{q}_{ijk}$ is defined up to a scale factor $\frac{\lambda_{ij}}{\mu_{ij}}$ like $\mathbf{r}_{ij}$. Moreover it is clear that we these scales are not independent because we must have $\mu_{ij} = \frac{\lambda_{ij}}{\lambda_{kj}}\ \mu_{kj}$ in coherence with the discussion of the previous paragraph.

From this relation, we can - knowing $\mathbf{r}_{kj}$ - recover $\mathbf{r}_{ij}$, thus Q_{ij} and Q_{kj} and Q_{ik}, and from frame to frame all Q-matrices. Finally, we have established that all r-vectors are entirely determined from the knowledge of the S-matrices or equivalently the F-matrices, up to one of them, say $\mathbf{r}_{01}$. This means that all F-matrices defines all Qs-representations, except 1 r-vector, so define at least $11\ N - 4$ parameters, and at most $11\ N - 1$ parameters.

Can we recover the 3 last parameters corresponding to the ultimate r-vector ? The answer is no, and this can be easily shown by induction : let us assume we have recovered all r-vectors considering $N + 1$ views and S-matrices and s-vectors, having $11\ N - 1$ parameters in this representation. Therefore, considering the N-first views we also must have been able to recover all r-vectors since we have at least 11 parameters less, in this case (a Q-matrix and a s-vector, up to a scale factor are undefined). By induction, for two views, i.e. N=1, we must have still $11\ N - 1 = 11$ parameters, which is not the case, a F-matrix being a function of 7 parameters, i.e. $\{11\ N - 4\}_{N+1=2}$, only. Therefore, 3 parameters, are indeed "missing".

Finally :

Proposition 3 *In a sequence of $N + 1$ views, the 2D-correspondences between points, when the depths have been eliminated, are defined by $11\ N - 4$ parameters when considering correspondences in the whole sequence. One r-vector of the Qs-representation is undefined, in this case.*

As a consequence, we must not only compute F-matrices between consecutive frames (i.e. F_{01}, F_{12}, F_{23}, etc...) which would provide no more than $7\ N$ parameters, but also F-matrices between non-consecutive frames (i.e. F_{02}, F_{03}, F_{13}, etc...), although very complex algebraic relations between these matrices occur. Therefore the representation of the motion using F-

---

[6] This algebraic construction is a very rich structure since we have, for instance :

$$\mathbf{r}_{ij}^T \cdot \mathbf{s}_{ji} = -1$$
$$\mathbf{r}_{ij}^T \cdot \mathbf{s}_{ij} = -\frac{\mathbf{s}_{ij}^t \cdot \mathbf{s}_{ji}}{\mathbf{s}_{ji}^t \cdot \mathbf{s}_{ji}}$$
$$\mathbf{r}_{ij}^T \mathbf{s}_{jk} = \frac{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ik}}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}} - 1$$
$$\mathbf{r}_{ji} = S_{ji}^T \cdot \mathbf{r}_{ij} - \frac{\mathbf{s}_{ij}}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}}$$
$$I = S_{ij} \cdot S_{ji} + \frac{\mathbf{s}_{ij} \cdot \mathbf{s}_{ij}^T}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}}$$
$$\left[I - \frac{\mathbf{s}_{ik} \cdot \mathbf{s}_{ik}^T}{\mathbf{s}_{ik}^T \cdot \mathbf{s}_{ik}}\right] \cdot \mathbf{s}_{ij} = S_{ik} \cdot \mathbf{s}_{kj}$$
$$\mathbf{r}_{kj} = S_{kj}^T \cdot \mathbf{r}_{ik} - \frac{(\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ik})\, S_{kj}^T \cdot S_{ik}^T \cdot \mathbf{s}_{ij} + (\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij})\, S_{ij}^T \cdot \mathbf{s}_{ik}}{(\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij})\,(\mathbf{s}_{ik}^T \cdot \mathbf{s}_{ik}) - (\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ik})^2}$$

all these relations being easily obtained introducing the S-matrix and s-vector in the relations of equations (10). They will be used for some technical developments, in this paper.

matrices is not always optimal in an image sequence. On the contrary, the Qs-representation up to some indetermination seems to be much simpler.

2.3 Application of the Qs-representation to other tokens

Using the Qs-representation for lines. Let us now compute the 2D correspondence between the two projection of a 3D line, in frame i and j. The 2D lines d_i and d_j are represented by their normal vectors $\mathbf{n}_i$ and $\mathbf{n}_j$. In fact, the same computation can be performed combining equation (5) with equation (8) rewritten in the reverse form : $\mathbf{N}_j = R_{ij}^T \cdot \mathbf{N}_i + \mathbf{t}_{ji} \wedge \mathbf{L}_j$. We thus have, performing a few algebra :

$$\alpha_j \, \mathbf{n}_j = \alpha_i \, Q_{ij}^T \cdot \mathbf{n}_i + \frac{\beta_j}{det(A_j)} \, \mathbf{s}_{ji} \wedge \mathbf{l}_j \tag{18}$$

Similarly, we have from $\mathbf{L}_i = R_{ij} \cdot \mathbf{L}_j$:

$$\beta_i \, \mathbf{l}_i = \beta_j \, Q_{ij} \cdot \mathbf{l}_j \tag{19}$$

So, in both cases (points and lines), the 2D correspondences between features related to the projection of a rigid object can be easily represented using the projection of the 3D features, and the Qs-representation.

Let us analyze these equations. From equation (18) and (5) we obtain :

$$\mathbf{l}_j = \frac{\underbrace{||A_j^{-1\,T} \cdot \mathbf{N}_j||}_{\alpha_j} \; det(A_j)}{\underbrace{||A_j \cdot \mathbf{L}||}_{\beta_j} \; (s_{ij}^T \cdot \mathbf{n}_i)} \left[Q_{ij}^T \cdot \mathbf{n}_i \wedge \mathbf{n}_j \right] \tag{20}$$

and replacing in equation (18) yields

$$0 = \left[\alpha_j \, (\mathbf{s}_{ji}^T \cdot \mathbf{n}_j) + \alpha_i \, (\mathbf{s}_{ij}^T \cdot \mathbf{n}_i) \right] \left[Q_{ij}^T \cdot \mathbf{n}_i - \mathbf{n}_j \right]$$

In other words we *only* recover $\mathbf{l}_j$ up to a scale factor and α_i/α_j from these equations.

Now, considering equations (18) and (19), we parametrize the line structure using α_j and $\mathbf{l}'_j = \beta_j \mathbf{l}_j$, with $0 = (\mathbf{l}_i'^T \cdot \mathbf{n}_i)$ (thus using 3 parameters), we parametrize the motion using Q_{ij} and $\mathbf{s}'_{ji} = \frac{1}{det(A_j)} \, \mathbf{s}_{ji}$ (12 parameters) and obtain the two following *normalized* equations :

$$\begin{cases} \alpha_j \, \mathbf{n}_j &= \alpha_i \, Q_{ij}^T \cdot \mathbf{n}_i + \mathbf{s}'_{ji} \wedge \mathbf{l}'_j \\ \mathbf{l}'_i &= Q_{ij} \cdot \mathbf{l}'_j \end{cases}$$

It is then obvious, using the same mechanism as for the motion of points, that due to the bilinearity of the equations, these unknowns are invariant upon several scalar transformations. Multiplying α_i, α_j, $\mathbf{l}'_j$ and $\mathbf{l}'_i$ by the same scale factor does not change the equations. More exhaustively the following scalar transformations leave the equations invariant :

1	$k\,\alpha_j$	$k\,\alpha_i$	$k\,\mathbf{l}'_j$	$k\,\mathbf{l}'_i$
2	$k\,\alpha_j$	$k\,\alpha_i$	$k\,\mathbf{s}'_{ji}$	
3	$k\,\alpha_j$	$k\,Q_{ij}$	$k\,\mathbf{l}'_j$	$k^2\,\mathbf{l}'_i$
4	$k\,\alpha_j$	$k\,Q_{ij}$	$k\,\mathbf{s}'_{ji}$	$k\,\mathbf{l}'_i$
5	$1/k\,\alpha_i$	$k\,Q_{ij}$	$k\,\mathbf{s}'_{ji}$	$1/k\,\mathbf{l}'_j$
6	$1/k\,\alpha_i$	$k\,Q_{ij}$	$k\,\mathbf{l}'_i$	
7	$k\,\mathbf{s}'_{ji}$	$1/k\,\mathbf{l}'_j$	$1/k\,\mathbf{l}'_i$	

This list is exhaustive. In order to avoid those indeterminations let us constrain our representation. Considering the scalar transformation number 6 in the previous table, we see that each frame i is subject to a "local" indetermination, which is canceled if we fix the scale of the Q-matrix, for instance with : $det(Q_{ij}) = 1$. Reciprocally, with this constraint, the indetermination 3, 4, 5 and 6 are canceled. The other indetermination 1, 2 and 7 are global for the whole sequence, since they affect both quantities in frame i and j and thus must be propagated in the whole sequence. Moreover, to fix only one value of either α or $||\mathbf{l}'||$ is not sufficient to avoid these three indeterminations. But reciprocally, if we fix one value of $||\mathbf{s}'||$, say $||\mathbf{s}'_{10}||$, indetermination 2 and 7 are canceled. The first indetermination remains, but it is not dependent upon the motion parameters, whereas it is only related to the structure of the line. So, considering an enumerable set of generic lines, we have $12 - 1$ parameters for the Qs-representation and 1 global scale factor. As a consequence, Prop. 1 is also true for lines, since the 2D-correspondences between lines is defined by $11\ N - 1$ parameters also. In addition, we are in the same situation as for Prop. 2. However, the parameters related to the structure of the line are defined up to a global scale factor and are not directly function of the Euclidean structure of the line.

We thus can conclude :

Proposition 4 *Prop. 1 and Prop. 2 are also true for line correspondences. Moreover, the parameters related to the uncalibrated structure of the line is defined by* $3\ N + 2$ *parameters, while* $N + 1$ *parameters where sufficient for points.*

It is clear, that there is a deep analogy between the motion of points and lines. Let us analyze in more details the reason of this analogy.

We consider 2 points m^a and m^b on a 2D-line d. In frame i the point m_i^a is in correspondence with a point in frame j of the form $m_j^a + \lambda^a\ \mathbf{l}_j$, for some unknown λ^a, since we can not locate its correspondent on the 2D-line, because we know it up to displacement on this 2D-line. We thus must consider a "sliding" point. Similarly, a point m_i^b is in correspondence with a point of the form $m_j^b + \lambda^b\ \mathbf{l}_j$, for some unknown λ^b.

Let us now assume that we know these two "partial" correspondences. This corresponds to taking only the *normal displacements* of these points into account. We thus can constrain Q_{ij} and $\mathbf{s}_{ij}$ since we can apply eq. (9) rewritten in the form :

$$\begin{cases} Z_i^a\ Q_{ij}^{-1} \cdot m_i^a &= Z_j^a \left[m_j^a + \lambda^a\ \mathbf{l}_j\right] - \mathbf{s}_{ji} \\ Z_i^b\ Q_{ij}^{-1} \cdot m_i^b &= Z_j^b \left[m_j^b + \lambda^b\ \mathbf{l}_j\right] - \mathbf{s}_{ji} \end{cases}$$

But from eq. (6) we have $\mathbf{n}_i = \rho_i(m_j^a \wedge m_j^b)$ while $\mathbf{l}_i = \kappa_i(Z_i^b\ m_j^b - Z_i^a\ m_j^a)$. using these relations, we can combine with the previous equations and obtain after some manipulations :

$$\begin{cases} \left[\frac{Z_j^a\ Z_j^b(1+(Z_i^b\ \lambda^b - Z_i^a\ \lambda^a)\kappa_j)}{\rho_j}\right]\ \mathbf{n}_j = \left[\frac{Z_i^a\ Z_i^b det(Q_{ji})}{\rho_i}\right]\ Q_{ji}^T \cdot \mathbf{n}_i + \left[Z_i^b\ \lambda^b - Z_i^a\ \lambda^a - \frac{1}{\kappa_j}\right]\ \mathbf{s}_{ji} \wedge \mathbf{l}_j. \\ \mathbf{l}_i = k\ Q_{ij}\ \mathbf{l}_j \end{cases}$$

which, in turn, can be identified with equations (18) and (19), after some additional algebra.

In other words, if we know these two "partial" correspondences, we can predict the "uncalibrated" retinal motion of the line. One important consequence is that when *considering any line we can replace its contribution to the evaluation of the uncalibrated motion by two normal correspondences between two points of the line.* We, for further use, will summarize this result as follow :

Proposition 5 *The retinal motion of a line is entirely determined by the normal displacement of two points of this line.*

The planar case In order to emphasis the generality of this approach, let us briefly develop the planar case. This will also help us understand some key ideas developed in the sequel.

In some situations, we can assume that all points belong to same 3D-plane S. We parameterized this plane considering its normal $\mathbf{N}_j$, with $||\mathbf{N}_j|| = 1$ and its distance to the origin d_j, these quantities being taken in frame j. We can write the plane equation as [4] :

$$M \in P \Leftrightarrow M^T \mathbf{N}_j = d_j \tag{21}$$

Combining this equation with equations (1) and (7) leads to :

$$Z_i\, m_i = Z_j \underbrace{\left[Q_{ij} + \mathbf{s}_{ij} \cdot \mathbf{n}_j^T\right]}_{H_{ij}} \cdot m_j \quad ; \quad \mathbf{n}_j = \frac{A_j^{-1T} \cdot \mathbf{N}_j}{d_j} \tag{22}$$

With this relation, we obtain the well-known fact that the retinal displacement of the projection of a planar patch is entirely defined by a collineation, and explicit the relations between this collineation and the Qs-representation. The $\mathbf{n}_j$ vector of this paragraph, resumes the information about the structure of the plane and corresponds to the "uncalibrated" plane equation since if m is the projection of a point of the plane P we obviously have $\mathbf{n}_j^T m = 1$.

As a consequence, using equation (22) and the first relation in footnote[6] for any collineation :

$$H_{ij} \cdot \mathbf{s}_{ji} = -\,\mathbf{s}_{ij}$$

All these relations prove that collineations are constrained by the motion parameters, as already used by [23].

Using this equation we can derive some useful relationships between the collineation and the fundamental matrix. We have :

$$\tilde{\mathbf{s}}_{ij} \cdot H_{ij} = \tilde{\mathbf{s}}_{ij} \cdot Q_{ij} = F_{ij} \tag{23}$$

for any collineation H_{ij}, while :

$$F_{ij}^T \cdot H_{ij} = F_{ij}^T \cdot Q_{ij} = S_{ij}^T \cdot \tilde{\mathbf{s}}_{ij} \cdot S_{ij} = \frac{\tilde{\mathbf{s}}_{ji}}{-det(Q_{ji})}$$

is a skew-symmetric matrix as already used by [23]. And finally using the previous decomposition of a Q-matrix we derive :

$$H_{ij} = S_{ij} + \mathbf{s}_{ij} \cdot [\mathbf{r}_{ij} + \mathbf{n}_{ij}]^T \tag{24}$$

This last equation is very important because it clearly shows, that given any collineation corresponding to an unknown plane, we measure $\mathbf{r}_{ij} + \mathbf{n}_{ij}$ only, and therefore the $\mathbf{r}_{ij}$ vector is not measurable in an independent way.

For the plane at infinity, $\frac{1}{d_j} = 0$ and : $H_{ij} = Q_{ij}$ as expected. Reciprocally, if Q_{ij} is known, the location of the plane at infinity is known since we can determine given point correspondences, whether they belong to this plane or not. This could also have been seen in equation (9), considering points with huge depths.

Moreover, we can provide a geometric interpretation of the $\mathbf{r}_{ij}$ vector, since the quantity $\mathbf{r}_{ij} + \mathbf{n}_{ij}$ is just equal to $\mathbf{r}_{ij}$ for the plane at infinity. Therefore this vector simply characterizes the affine structure of the plane at infinity.

Finally the S-matrix can also be interpreted as follows : because S_{ij} is singular, it is not a collineation, but still a correspondence between the two retinas induced by a plane, which contains one optical center (this explains why the transformation is singular). The image of a point m_j in the frame j is $m_i = \lambda\, \mathbf{s}_{ij} \wedge F_{ij} \cdot m_j$ in other words, the intersection of the epipolar associated to m_j with the line $\delta_{\mathbf{s}_{ij}}$ of equation $m \in \delta_{\mathbf{s}_{ij}} \Leftrightarrow m^T \cdot \mathbf{s}_{ij} = 0$. However the underlying geometric construction is out of the scope of this paper.

Considering now equation (23) we can make the following remark : this equation corresponds to the "motion equation for planar patches" in the sense that we have eliminated the parameters related to the structure and have summarized all information about motion, and independent of the structure of the plane, in this equation. This leads to the following conclusion :

Considering correspondences between coplanar structures does not allow to recover the Q-matrix but only the fundamental matrix. We thus must obtain local correspondences between features in these planes.

On the reverse, the Qs-representation being known, it is very easy to recover the uncalibrated structure of the plane since :

$$\mathbf{n}_j = \frac{\lambda H_{ij}^T \cdot \mathbf{s}_{ij} - \mathbf{r}_{ij}}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}} \quad ; \quad \lambda = \frac{||\tilde{\mathbf{s}}_{ij} \cdot Q_{ij}||}{||\tilde{\mathbf{s}}_{ij} \cdot H_{ij}||} \tag{25}$$

Moreover, combining equations (1) and (7) with equation (21) we calculate the evolution of the parameter related to the structure of the plane and easily obtain :

$$\mathbf{n}_j = \frac{Q_{ij}^T \cdot \mathbf{n}_i}{1 - \mathbf{s}_{ij}^T \cdot \mathbf{n}_i}$$

Therefore, as for points and lines, the uncalibrated parameters related to the structure of the plane are entirely defined by the Qs-representation.

From these relations and equations (10) we immediately deduce :

$$\begin{aligned} H_{ii} &= 0 \\ H_{ji} &= H_{ij}^{-1} \\ H_{ik} \cdot H_{kj} &= H_{ij} \end{aligned}$$

It is not a surprise that Q-matrices behave like all collineations (see equation (10)), because Q-matrices are particular collineations.

This digression about the planar case illustrates two key ideas of this paper :

1. Using the Qs-representation we can derive the equations about (1) motion, (2) structure from motion or (3) evolution of structure parameters as in the calibrated case, but using the "retinal projection" of the Euclidean parameters. We conjecture that this can be done for any set of primitives[7]

---

[7]Although this is not the purpose of this paper, we can offer to the reader -for instance - the formulas in the case of a planar conic. A planar conic is defined by the intersection of a plane and of quadric, say :

$$\begin{cases} M^T \cdot \mathbf{N}_i = d_i & \textit{plane equation} \\ M^T \cdot C_i \cdot M + \mathbf{d}_i^t \cdot M + e_i = 0 & \textit{quadric equation} \end{cases}$$

It can be easily shown that the retinal projection of this conic is in homogeneous coordinates :

$$m_i^T \cdot A_i^{-1\,T} \cdot \underbrace{\left[C_i + \mathbf{d}_i \cdot \frac{\mathbf{N}_i}{d_i}^T + e_i \frac{\mathbf{N}_i}{d_i} \cdot \frac{\mathbf{N}_i}{d_i}^T \right]}_{\Gamma_i} A_i^{-1} \cdot m_i = 0$$

The Γ_i matrix represents, with $\mathbf{n}_i = \frac{A_i^{-1\,T} \cdot \mathbf{N}_i}{d_i}$ the "uncalibrated" structure of the conic, the former being the $2D$ information (what is measure in the image) and the latter the $3D$ information (what should be estimated to recover the scene structure). The geometrical interpretation is that Γ_i defines both the retinal image of the conic and a $3D$ cone which contains the $3D$ conic, this curve being the intersection of the $3D$ cone with the $3D$ plane defined by $\mathbf{n}_i$. Therefore the structure from motion problem is to recover $\mathbf{n}_i$ as developed in this paragraph. Moreover, the prediction of Γ_i from one frame to another is given by the following formula :

$$\Gamma_j = Q_{ij}^T \cdot \Gamma_i \cdot Q_{ij} + Q_{ij}^T \cdot \Gamma_i \cdot \mathbf{s}_{ij} \cdot \mathbf{n}_j^T + \mathbf{n}_j \cdot \mathbf{s}_{ij}^T \cdot \Gamma_i \cdot Q_{ij} + \mathbf{n}_j \cdot \mathbf{s}_{ij}^T \cdot \Gamma_i \cdot \mathbf{s}_{ij} \cdot \mathbf{n}_j^T$$

and thus entirely defined by the Qs-representation, again.

2. Not all the parameters of the Qs-representation are computable from motion equations, but a subset, sufficient to predict the disparity between two consecutive frames, i.e. the correspondences between tokens in two consecutive frames. This corresponds to the projective parameters of the scene.

2.4 Using the Qs-representation to analyze the scene structure.

Structure from motion using the Qs-representation. In order to increase our understanding of the potentialities of this representation, let us explicit a very useful, but oversimple result. The formula is obtained from equation (7) very easily :

Proposition 6 *Using the Qs-representation, the 3D affine depth of an image point can be directly recovered, without an explicit knowledge of the intrinsic calibration (which can change at any time) and of the Euclidean displacements, since :*

$$Z_j \ [Q_{ij} \cdot m_j \wedge m_i] + [\mathbf{s}_{ij} \wedge m_i] = 0$$

This result means that we can directly build a depth map, the Q-matrix and s-vector being known. Since we obtain two equations for the depth from this equation (related to a cross-product, thus only defined in two directions of the space), the estimation can, in practice be weighted considering some knowledge about the 2D correspondence. In particular, if the retinal displacement $d_{ij} = m_i - m_j$ is only known in one direction, as it appears along curves for which only the normal displacement can be recovered, one can - in general - still evaluate the depth.

This result is very useful in practice for the following reason : if we are interested in building a depth map, we can deal with this Qs-representation, without recovering the Euclidean parameters of the system, which is generally not possible as given in Prop.2.

However, one must understand that we do not recover the Euclidean structure of the scene, since the depth Z_j is a function of m_j and not of M_j. We must recover the A-matrix, i.e. the intrinsic calibration parameters, since $m_j = \lambda\, A \cdot M_j$, to reconstruct the Euclidean depth map. If not, we only reconstruct the depth map, up to an *affine transformation*of the image, given by the matrix A, and this corresponds to an affine transformation of the 3D space.

If we do not known the Q-matrix but only the related S-matrix, as when considering F-matrices, we can decompose the Q-matrix and obtain :

Proposition 7 *When the r-vector of the Qs-representation is not known, the 3D depth of an image point can be directly recovered, up to a collineation of the depth map, without an explicit knowledge of the intrinsic calibration (which can change at any time) and of the Euclidean displacements, since :*

$$\underbrace{\frac{Z_j}{1 + Z_j\,(\mathbf{r}_{ij}^T \cdot m_j)}}_{\zeta_j} \ [S_{ij} \cdot m_j \wedge m_i] + [\mathbf{s}_{ij} \wedge m_i] = 0$$

The quantity ζ_j is related to the Euclidean depth through a special collineation. The corresponding coefficients are functions of the retinal location. In such a case we reconstruct the depth map, not only up to affine transformation of the image plane, but also up to projective transform of the depth. These transformations have been already made explicit [3], in the case of a stereo rig. Therefore, we can design the following strategy :

- compute a *precise* 3D reconstruction, up to an affine/projective transformation, *without calibration*, and then
- minimize the previous transformation from the knowledge of the calibration parameters

More generally, considering equation (9) and following the same method as in [13], when calibration was taken into account, we can make the following count of equations. Given $N + 1$

views of a monocular image sequence and P points, the problem of recovering the depths up to a scale factor can be solved only if we have more equations than unknowns, more precisely :

$$\underbrace{((1+N)\,P)_{depth} + (11\,N-1))_{QS-parameters}}_{unknowns} - (3\,N\,P)_{equations} \leq 0 \Leftrightarrow P \geq \frac{11\,N-1}{2\,N-1}$$

Let us explicit the minimum number P_{min} of points to solve equation (9) :

$N+1$	3	4	5	6	...	∞
P_{min}	7	$7-\frac{3}{5}$	$7-\frac{6}{7}$	6	...	$6-\frac{1}{2}$

We mention this result here because we now know it is a mistake to assume that with 2 views and 10 points (as given by the formula) one can recover the depths, whereas we will show in this paper, that using at least 3 views, it is. Good to know.

One application of this method is that for $P > P_{min}$ there exists some internal relations between the coordinates of the points, neither dependent on motion nor on depth, thus **invariant**. It is known that given more than 8 points in 2 views, there exists invariants [11]. The previous count allows us to conjecture that given more than 7 points in 3, 4 or 5 views or more than 6 points in 6 views or more, there exists invariants. These invariants, are related to the rigidity constraint, i.e. they are verified if and only if the points belong to the same rigid object.

Can we generalize Prop. 6 considering lines ? Yes. But this is less obvious. Let us explain how.

It is easily seen from equation (20) that it is not possible to recover $\mathbf{L} = A_j^{-1} \cdot \mathbf{l}_j$ and the distance from the line to the origin $||\mathbf{N}_j||$ - which would have characterized the Euclidean parameters of the line [30] - from the equations, without knowing the A-matrix. We, in fact, recover $\mathbf{l}_j$ only up to a scale factor.

However, we can write for a point m_i on the line d_i that there is a correspondent m'_j on the line d_j with $m'_j = m_j + \lambda\,\mathbf{l}_j$ for some unknown λ. Replacing in equation (9), using equation (20) and eliminating Z_j and λ we derive, after some straightforward simplifications, the depth Z_i^a of a point $m_i^a \in d_i$, as :

$$Z_i^a \left[m_i^{a\,T} \cdot Q_{ij} \cdot \mathbf{n}_j \right] + \left[\mathbf{s}_{ij}^T \cdot \mathbf{n}_j \right] = 0$$

This equation, as expected, corresponds to the use of the retinal disparity in a direction normal to the line. Finally, we thus can compute the depth of any point on a 2D line, using only the Qs-representation. Again, we recover the location of the line, up to a transformation, as discussed for points.

Relation between the Qs-representation and Euclidean parameters The Q-matrix is in deep relation with intrinsic parameters, the A-matrix, and the rotational component of the displacement. Let us explicit this relation.

If we write that $R_{ij} = A_i^{-1} \cdot Q_{ij} \cdot A_j$ is an orthogonal matrix we obtain, after some algebra :

$$Q_{ij} \cdot K_j \cdot Q_{ij}^T = K_i \quad \Leftrightarrow \quad Q_{ij}^T \cdot K_i^{-1} \cdot Q_{ij} = K_j^{-1} \tag{26}$$

where :

$$K_i = A_i \cdot A_i^T = \begin{pmatrix} \underbrace{\alpha_u^2 + \gamma^2 + u_0^2}_{b_u} & \underbrace{u_0\,v_0 + \gamma\,\alpha_v}_{b_c} & u_0 \\ \underbrace{u_0\,v_0 + \gamma\,\alpha_v}_{b_c} & \underbrace{\alpha_v^2 + v_0^2}_{b_v} & v_0 \\ u_0 & v_0 & 1 \end{pmatrix} \tag{27}$$

is a symmetric positive matrix in one-to-one correspondence with the A-matrix since :

$$\alpha_v = \sqrt{b_v - v_0^2} \quad ; \quad \gamma = \frac{b_c - u_0\,v_0}{\alpha_v} \quad ; \quad \alpha_u = \sqrt{b_u - u_0^2 - \gamma^2} \tag{28}$$

which is unambiguous because $\alpha_u > 0$ and $\alpha_v > 0$.

This matrix defines, in fact, the retinal image of the absolute conic at infinity which equation is $0 = m^T \cdot K^{-1} \cdot m$, that is the angular relations between each optical ray, or in other words, the intrinsic calibration or the Euclidean geometry of the system [19].

Then, if we know the matrices K_i and K_j we can calculate A_i and A_j, then $R_{ij} = A_i^{-1} \cdot Q_{ij} \cdot A_j$ and $\mathbf{t}_{ij} = A_i^{-1} \cdot \mathbf{s}_{ij}$. We thus obtain intrinsic and extrinsic (motion) parameters.

But the matrices K_i and K_j depend linearly on Q_{ij}. More precisely equation (26) provides 5 equations on the intrinsic parameters, because it involves an equality between 3x3 symmetric matrices, thus yielding 6 equations, but up to a scale factor. Usually Q is known up to a scale factor, but K is also defined up to a scale factor; therefore this indetermination does not affect the equation.

This is a very important fact that we can compute the intrinsic calibration parameters using a linear algorithm, while it was not the case in earlier works [18, 19]. In practice these equations are to be solved using a statistical framework and a model of the evolution of the parameters since we have more unknowns than measurement equations, as discussed now.

If we consider the system of equations, for $N + 1$ views, because for any i, j, k, we have :

$$Q_{ij} \cdot K_j \cdot Q_{ij}^T = K_i \quad ; \quad Q_{jk} \cdot K_k \cdot Q_{jk}^T = K_j \quad \Rightarrow \quad Q_{ki} \cdot K_i \cdot Q_{ki}^T = K_k \tag{29}$$

we obtain, if generic at most 10 linear equations in the 15 unknowns. By an obvious induction, it is clear that for $N + 1$ views we will generate at most 5 N equations while we have 5 $(N + 1)$ unknowns in the general case. We thus are left with 5 unknowns in coherence with Prop.1 and 2.

However, if the initial values of the parameters - say K_0 - are known, we thus can compute, using equation (29), K_1 from Q_{10}, K_2 from Q_{21}, etc... In other words, we have a positive result to oppose to Prop. 2, i.e. :

Proposition 8 *When studying the motion of points with an uncalibrated system having variables calibration parameters, we can recover the calibration and motion parameters from the observation of the retinal disparities if the initial values of the calibration parameters are known.*

It appears that we can recover the intrinsic parameters with the Qs-representation in an apparently simpler way, than using the projective representation using the F-matrix [19, 6]. Where does it comes from ? A very deep reason. As explained before, the Q-matrix corresponds to the collineation of the plane at infinity. From the section on the planar case, we see that if we know the collineation, we have the equation of this plane. But knowing the plane at infinity, means knowing the *affine geometry* of the scene [29]. In other words, *the Qs-representation is an intermediate representation of the motion in which the affine geometry of the scene has been made explicit.* A projective representation would have involved F-matrices only. A Euclidean representation would have involved rotation matrices, translation vectors and calibration parameters. An affine representation, now, involves Q matrices and s-vectors .

Moreover, it is possible, for a rather large class of real scenes, to infer the collineation of the plane at infinity, considering points "at the horizon", that is with a negligible depth [29]. This segmentation process, although not rigorous is quite efficient .

Let us finally discuss the case where calibration parameters are constant. In that case $K_i = K_j = K_k = K$ while $det(Q_{ij}) = 1$. The equation $Q_{ij} \cdot K \cdot Q_{ij}^T = K$ provides 5 linear equations in the five unknowns $(u_0, v_0, b_u, b_v, b_c)$. In fact, this equation can be solved explicitly, but, unfortunately, this does not allow to recover these five parameters. Let us explain why. Let us represent the rotations using the exponential of a skew-symmetric matrix, that is :

$$R_{ij} = e^{\tilde{\mathbf{r}}_{ij}}$$

where $\mathbf{r}$ is a vector aligned with the rotation axis and which magnitude is equal to the angle of the rotation (see for instance [4]).

We can write :

$$Q_{ij} = A \cdot R_{ij} \cdot A^{-1} = A \cdot e^{\tilde{\mathbf{r}}_{ij}} \cdot A^{-1} = e^{A \cdot \tilde{\mathbf{r}}_{ij} \cdot A^{-1}} = e^{K \cdot \tilde{\rho}_{ij}}$$

with $\rho_{ij} = \frac{A \cdot \mathbf{r}}{det(A)}$. This relation is only valid if $K_i = K_j = K$. This relations allows to compute $L_{ij} = K \cdot \tilde{\rho}_{ij}$ as the one logarithm matrix of Q_{ij} considering only the unique real solution, since $L_{ij} = K \cdot \tilde{\rho}_{ij}$ is a real matrix. Moreover it is clear that we cannot all components of K but only in a direction orthogonal to ρ_{ij}. Therefore *we must perform at least two rotations around two non-parallel axes to recover the intrinsic parameters.* Reciprocally, with two such rotations, if we can compute the corresponding Q_{ij} matrices, and thus the corresponding L_{ij} matrices, we can recover K from the two equations $L_{ij} = K \cdot \tilde{\rho}_{ij}$. This is coherent to what has been observed by [28].

So we can recover the intrinsic calibration parameters as soon as three views are given, and we have an overdetermined linear system, solvable in a statistical framework again.

Auto-calibration from motion Let us now discuss how to recover the calibration parameters of the system, when the Q-matrix is not entirely known.

This means that we do not have an affine geometry but only projective quantities.

If we assume that we know the F-matrices, i.e. S-matrices and s-vectors, we have to recover r-vectors and K-matrices. This problem can be analysed as follows.

Considering equation (29) in which we explicit the S-matrix, r-vector and s-vector, we obtain, after some simplifications, the following set of equations :

$$\left\{ \begin{array}{rcl} S_{ij} \cdot K_j \cdot S_{ij}^T & = & \left(I - \frac{\mathbf{s}_{ij} \cdot \mathbf{s}_{ij}^T}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}}\right) \cdot K_i \cdot \left(I - \frac{\mathbf{s}_{ij} \cdot \mathbf{s}_{ij}^T}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}}\right) \\ S_{ij} \cdot K_j \cdot \mathbf{r}_{ij} & = & \left(I - \frac{\mathbf{s}_{ij} \cdot \mathbf{s}_{ij}^T}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}}\right) \cdot K_i \cdot \frac{\mathbf{s}_{ij}}{\mathbf{s}_{ij}^T \cdot \mathbf{s}_{ij}} \end{array} \right. \tag{30}$$

since we have $\mathbf{r}_{ij}^T \cdot K_j \cdot \mathbf{r}_{ij} = \mathbf{s}_{ij}^T \cdot K_i \cdot \mathbf{s}_{ij}$.

These two equations allow to perform auto-calibration, let us explain how.

The first equation provides 3 linear homogeneous equations for the K-matrices. Because the matrices are symmetric, we have 6 equations, but this matrix expression is null in the direction of $\mathbf{s}_{ij}$ so 3 equations are always verified.

It is true that S_{ij} is known up to a scale factor only but K_j also, and this scale factor could be integrated in the definition of K_j since this matrix is only defined up to a scale factor, and the equation is coherent.

Finally, it is easy to remark that the matrix K_i cannot be recovered in the direction of $\mathbf{s}_{ij}$ since if we write : $K_i = \kappa_i \, \mathbf{s}_{ij} \cdot \mathbf{s}_{ij}^T + \Lambda_i$ the quantity κ_i disappears in the equation. Similarly, K_j cannot be recovered in the direction of $\mathbf{s}_{ji}$.

The second equation provides, if $\mathbf{r}_{ij}$ is known, an additional set of 2 linear homogeneous equations for the K-matrices, because this vectorial expression is null in the direction of $\mathbf{s}_{ij}$. The scale factors are coherent with respect to the first equation. If generic these $3 + 2 = 5$ equations are independent. We obtain in fact the 5 equations (29) again.

If $\mathbf{r}_{ij}$ is not known or partially known, but K_j has been estimated, the second equation provides a set of 2 linear homogeneous equations for this quantity. We have only 2 equations since this vectorial expression is null in the direction of $\mathbf{s}_{ij}$. Moreover, $\mathbf{r}_{ij}$ can not be recovered in the direction $K_j^{-1} \cdot \mathbf{s}_{ji}$. We only have homogeneous equations since the K-matrix and S-matrix are defined up to a scale factor.

Let us now apply these equations when the 3 motions between 3 views is given. From the first equation of (30), using S_{01}, S_{21}, $\mathbf{s}_{01}$ and $\mathbf{s}_{21}$, we obtain 6 equations to determine the K-matrices. In particular, if $K_0 = K_1 = K_2 = K$ the problem can be solved. More generally, having a model on the evolution of the intrinsic parameters, i.e. of the K-matrices one can combine it with this 6 measurement equations.

From the second equation of equations (30), using the estimation of K_0, K_2 and K_1, we recover $\mathbf{r}_{01}$ up to a scale factor and not in the direction $\mathbf{s}'_{01} = K_0^{-1} \cdot \mathbf{s}_{01}$, i.e. a vector $\mathbf{q}_0$ such

that $\lambda_0\,\mathbf{q}_0 = \mathbf{r}_{01} + \mu_0 \mathbf{s}'_{01}$. Similarly, we recover a vector $\mathbf{q}_2$ such that $\lambda_2\,\mathbf{q}_2 = \mathbf{r}_{21} + \mu_2 \underbrace{K_2^{-1} \cdot \mathbf{s}_{21}}_{\mathbf{s}'_{21}}$.

But we also have recovered $\mathbf{q}_{012} = \lambda_1\,\mathbf{q}_1$ up to a scale factor, i.e. we can write $\lambda_1\,\mathbf{q}_1 = \mathbf{r}_{01} - \mathbf{r}_{21}$ for some λ_1. Using the relation $\mathbf{r}_{ij} \cdot \mathbf{s}_{ji} = -1$ we can eliminate these scale factors and finally obtain :

$$\left\{\begin{array}{rcl} \mathbf{r}_{01} & = & -\frac{\mathbf{s}'_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}} - \frac{|\mathbf{q}_1, \frac{\mathbf{s}'_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}} - \frac{\mathbf{s}'_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}}, \frac{\mathbf{q}_2}{\mathbf{q}_2^{T}\cdot\mathbf{s}'_{12}} - \frac{\mathbf{s}'_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}}|}{|\mathbf{q}_1, \frac{\mathbf{s}'_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}} - \frac{\mathbf{q}_0}{\mathbf{q}_0^{T}\cdot\mathbf{s}'_{10}}, \frac{\mathbf{q}_2}{\mathbf{q}_2^{T}\cdot\mathbf{s}'_{12}} - \frac{\mathbf{s}'_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}}|}\,\frac{\mathbf{s}_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}} - \frac{\mathbf{q}_0}{\mathbf{q}_0^{T}\cdot\mathbf{s}'_{10}} \\ \mathbf{r}_{21} & = & -\frac{\mathbf{s}_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}} - \frac{|\mathbf{q}_1, \frac{\mathbf{s}'_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}} - \frac{\mathbf{q}_0}{\mathbf{q}_0^{T}\cdot\mathbf{s}'_{10}}, \frac{\mathbf{s}'_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}} - \frac{\mathbf{s}'_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}}|}{|\mathbf{q}_1, \frac{\mathbf{s}'_{01}}{\mathbf{s}'^{T}_{01}\cdot\mathbf{s}'_{10}} - \frac{\mathbf{q}_0}{\mathbf{q}_0^{T}\cdot\mathbf{s}_{10}}, \frac{\mathbf{q}_2}{\mathbf{q}_2^{T}\cdot\mathbf{s}'_{12}} - \frac{\mathbf{s}'_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}}|}\,\frac{\mathbf{s}_{21}}{\mathbf{s}'^{T}_{21}\cdot\mathbf{s}'_{12}} - \frac{\mathbf{q}_2}{\mathbf{q}_2^{T}\cdot\mathbf{s}'_{12}} \end{array}\right.$$

We thus can recover the **r**-vectors from the knowledge of the K-matrices, without an explicit recovery the Euclidean motion and the calibration parameters.

Since we also have recovered S and **s** up to a *common* scale factor, we have entirely recover the Q-matrix up to a scale factor. Unfortunately, it appears that we must compute the Euclidean calibration (i.e. the K-matrix) to obtain the **r** vectors.

Unfortunately again, this process is feasible if and only if, either the K-matrices are constant, or some other hypotheses are available on the calibration of the system.

2.5 Motion of points and lines in three views

Let us now study the problem of computing the motion parameters S_{ij}, $\mathbf{s}_{ij}$ and $\mathbf{q}_{ijk}$, considering points and/or lines correspondences.

In this section we are going to study the case of three views in detail for several reasons :

1. It is known that for lines, when calibration is given, 3 views are needed [15, 9], while we now know that at least 3 views are also needed considering point correspondences when calibration is not given.
2. We are going to derive linear and non-linear algorithms in this simple case, in order to analyze how complex would their implementations be.
3. As a control, we would like to verify that the results obtained in the previous sections are indeed valid, and we derive them again in this case.
4. The geometric approach will focus on the three views problem.
5. In the literature, most of the results are derived considering 2 or 3 views only, and this will allow to compare with previous works.

As discussed previously these quantities depends on : $(9+9)_{Q-matrices} + (3+3)_{s-vectors} = 24$ numbers but up to 3 scale factors and the indetermination of one r-vector, thus yielding 18 parameters.

Motion equations for points and lines. To obtain these equations, we simply eliminate parameters related to the 3D structure of the scene.

When point correspondences are given, eliminating Z_0, Z_1 and Z_2 in equation (9) and using equation (13), we have :

$$m_1^T \cdot \underbrace{\tilde{\mathbf{s}}_{10} \cdot Q_{10}}_{F_{10}} \cdot m_0 = 0 \quad ; \quad m_1^T \cdot \underbrace{\tilde{\mathbf{s}}_{12} \cdot Q_{12}}_{F_{12}} \cdot m_2 = 0 \quad ; \quad m_0^T \cdot \underbrace{Q_{10}^T \cdot [\tilde{\mathbf{s}}_{12} - \tilde{\mathbf{s}}_{10}] \cdot Q_{12}}_{F_{02}} \cdot m_2 = 0 \quad (31)$$

Please note that these three equations are independent, as already shown [18].

The two first equalities can be written, equivalently[8] :

$$0 = m_1 \wedge \left[\left[\underbrace{\mathbf{s}_{10} \wedge [Q_{10} \cdot m_0]}_{F_{10} \cdot m_0} \right] \wedge \left[\underbrace{\mathbf{s}_{12} \wedge [Q_{12} \cdot m_2]}_{F_{12} \cdot m_2} \right] \right] \tag{32}$$

and the last one :

$$|\,[Q_{10} \cdot m_0], \mathbf{s}_{12}, [Q_{12} \cdot m_2]\,| = |\,[Q_{10} \cdot m_0], \mathbf{s}_{10}, [Q_{12} \cdot m_2]\,| \tag{33}$$

This means that given a point correspondence ($m0 \leftrightarrow m2$) in two views, one can - knowing the Q-matrices and s-vectors - predict the location of the point in the third view. This means that we can perform "trinocular stereo, when the system is weakly calibrated" [22, 23], that is without recovering explicitly the Euclidean parameters of the system.

These results, considering point correspondences were already known. The corresponding results considering lines are not, and we derive them now.

Similarly, eliminating α_1, α_0, α_2 and $\frac{\beta_1}{det(A_1)}$ in equation (18), written for ($j = 1, i = 0$) and ($j = 1, i = 2$) we derive :

$$0 = \mathbf{n}_1 \wedge \left[\frac{Q_{01}^T \cdot \mathbf{n}_0}{(\mathbf{s}_{01}^T \cdot \mathbf{n}_0)} - \frac{Q_{21}^T \cdot \mathbf{n}_2}{(\mathbf{s}_{21}^T \cdot \mathbf{n}_2)} \right] \tag{34}$$

Then each line correspondence provides at least two equations, as in the calibrated case, while knowing a line correspondence between two views allows to compute the location of the line in the third view. Again, this shows that correspondences can be verified without recovering the Euclidean parameters.

This equation can easily be decomposed in the two equivalent scalar equations :

$$|\mathbf{n}_1, Q_{01}^T \cdot \mathbf{n}_0, Q_{21}^T \cdot \mathbf{n}_2| = 0 \quad ; \quad \frac{||\mathbf{n}_1 \wedge Q_{01}^T \cdot \mathbf{n}_0||}{(\mathbf{s}_{01}^T \cdot \mathbf{n}_0)} = \frac{||\mathbf{n}_1 \wedge Q_{21}^T \cdot \mathbf{n}_2||}{(\mathbf{s}_{21}^T \cdot \mathbf{n}_2)} \tag{35}$$

which correspond to the two equations found by Liu and Huang [15] in the case where calibration is given. They are now generalized.

In the Euclidean case, the geometrical interpretation of the first equation is that the three normals of the three projection planes are coplanar [30], and using the Qs-interpretation we can easily verify, that this interpretation is still valid. In the Euclidean case, the geometrical interpretation of the second equation is that the distance from the line to the origin, computed by triangulation using view 0 and 1 is the same as the distance from the line to the origin, computed by triangulation using view 2 and 1 [7], but this interpretation is no more valid because the concept is only Euclidean.

We can permute the indices 0, 1 and 2 in equation (34) and obtain two other equations which relate $\mathbf{n}_0$, $\mathbf{n}_1$ and $\mathbf{n}_2$, but contrary to the case of points, we will show that they are equivalent.

A first result on points and lines motion. Introducing the decomposition of a Q-matrix from equation (15) in equation (34) yields :

$$0 = \mathbf{n}_1 \wedge \left[\frac{S_{01}^T \cdot \mathbf{n}_0}{\mathbf{s}_{01}^T \cdot \mathbf{n}_0} - \frac{S_{21}^T \cdot \mathbf{n}_2}{\mathbf{s}_{21}^T \cdot \mathbf{n}_2} + \underbrace{\mathbf{r}_{01} - \mathbf{r}_{21}}_{\mathbf{q}_{012}} \right]$$

[8] In fact $m_1 \wedge \mathbf{X} = 0 \Leftrightarrow m_1 = \lambda \mathbf{X}$ for some λ. Therefore, m_1 is entirely defined from the correspondence m_0 and m_2, using (32), since it is an homogeneous quantity, as obtained in [22].

This shows that we cannot recover all parameters of the Q-matrices, but only the expected quantities.

This leads to the following result :

Proposition 9 *Considering correspondences between lines, the Qs-representation is a function of at most 18 parameters, and only two equations are available for each line correspondence.*

Proof

Let us multiply Q_{ij} by λ_{ij} and $\mathbf{s}_{ij}$ by μ_{ij} in equation (34) such that $\frac{\lambda_{01}}{\mu_{01}} = \frac{\lambda_{21}}{\mu_{21}}$. The equations are not modified. Therefore $Q_{01}, Q_{21}, \mathbf{s}_{01}, \mathbf{s}_{21}$ are only recovered up to three scale factors, leaving three parameters unknown.

Moreover the equations on lines are only function of $\mathbf{q}_{012} = \mathbf{r}_{01} - \mathbf{r}_{21}$ whereas the sum of these two vectors can not be found. Yet another triplet of parameters unknown. We thus do not have an explicit dependency with respect to 24 parameters, but at most $24 - 3 - 3 = 18$.

Let us now show that equation (34) is equivalent to any other equations relating $\mathbf{n}_0$, $\mathbf{n}_1$ and $\mathbf{n}_2$. Since the only way to generate an equation about the motion of the line which is independent of the structure of the line is to eliminate the parameters related to this structure, i.e. α_i and β_i, the only way to generate an equation about the motion of line between three views from equation (18), is to generate an equation of the form of (34), up to a permutation of the indices.

Let us consider, for instance, the equation :

$$0 = \mathbf{n}_2 \wedge \left[\frac{Q_{12}^T \cdot \mathbf{n}_1}{(\mathbf{s}_{12}^T \cdot \mathbf{n}_1)} - \frac{Q_{02}^T \cdot \mathbf{n}_0}{(\mathbf{s}_{02}^T \cdot \mathbf{n}_0)} \right]$$

obtained after a permutation of the indices. Any other permutation could have been chosen.

Multiplying on the left by Q_{21} yields

$$\lambda\, \mathbf{n}_1 = a\, Q_{21}^T \cdot \mathbf{n}_2 + b\, Q_{01}^T \cdot \mathbf{n}_0 \quad ; \quad \frac{\lambda}{b} = \frac{\mathbf{s}_{02}^T \cdot \mathbf{n}_0}{\mathbf{s}_{12}^T \cdot \mathbf{n}_1}$$

But $(\mathbf{s}_{02}^T \cdot \mathbf{n}_0) = (\mathbf{s}_{12}^T \cdot Q_{01}^T \cdot \mathbf{n}_0) - (\mathbf{s}_{10}^T \cdot Q_{01}^T \cdot \mathbf{n}_0)$ and $\lambda\, (\mathbf{s}_{12}^T \cdot \mathbf{n}_1) = b\, (\mathbf{s}_{12}^T \cdot Q_{01}^T \cdot \mathbf{n}_0) + a\, (\mathbf{s}_{12}^T \cdot Q_{21}^T \cdot \mathbf{n}_2)$. We thus obtain $\frac{a}{b} = -\frac{\mathbf{s}_{01}^T \cdot \mathbf{n}_0}{\mathbf{s}_{21}^T \cdot \mathbf{n}_2}$ and as a consequence equation (18). But this derivation is true for any permutation of the indices, thus can be done in the reverse way. These two equations are equivalent, and so are any equation about the motion of lines using three views.

End Of Proof

The demonstration of the reverse is given in the next sections, in Prop. 12.

More precisely since each line provides two equations, at least 9 generic lines are required to recover the motion parameters, from a set of 18 quadratic equations obtained from equation (34) and 6 constraints :

$$\begin{cases} 0 &= Q_{01}^T \cdot \mathbf{s}_{01} + Q_{21}^T \cdot \mathbf{s}_{21} \\ 1 &= ||Q_{01}||^2 \\ 1 &= ||Q_{21}||^2 \\ ||\mathbf{s}_{01}||^2 &= ||\mathbf{s}_{21}||^2 \end{cases}$$

in 24 unknowns.

This results leads to several comments : applying the Bezout Theorem means that we can find at most ... 2^{24} solutions !!! This must certainly be improved. The last constraint states a result known in the case of a calibrated system [30] but still valid here : *when studying the motion of lines, the relative scale of the translational component of the displacement is measurable*, whereas the scale indeterminate is present but for the whole motion only. This is also the case for points, as visible in the last equation of (31).

Let us now state the results for points, in coherence to what has been found by [19] :

Proposition 10 *Considering correspondences between points, the 3 fundamental matrices are functions of at most 18 parameters, and only three equations are available for each point correspondence.*

Proof If we have to estimate 3 fundamental matrices, defined by 7 parameters each, this involves 21 parameters at most. But there exists three constraints between these matrices, coming from equation (14) :

$$\mathbf{s}_{12}^T \cdot F_{10} \cdot \mathbf{s}_{02} = 0 \quad ; \quad \mathbf{s}_{10}^T \cdot F_{12} \cdot \mathbf{s}_{20} = 0 \quad ; \quad \mathbf{s}_{01}^T \cdot F_{02} \cdot \mathbf{s}_{21} = 0$$

Therefore we have at most $21 - 3$ independent parameters for this representation also.

We do not re-build the proof of the independency of the three equations, since this has been already established [18, 19]. **End Of Proof**

The demonstration of the reverse is also given in the next sections, in Prop. 12.

Thus, at least 6 generic points are required to recover the 18 parameters of the motion.

Let us now consider correspondences between points **and** lines at the same time. Are the dependencies with respect to the motion parameters complementary ? In fact no, because we can map one set of parameters onto another, as made explicit now.

Proposition 11 *Considering correspondences between points and lines, the motion parameters are functions of at most 18 parameters.*

Proof Let us assume we have computed the motion parameters from line correspondences. In that case we have estimated S_{01}, S_{21}, $\mathbf{s}_{01}$, $\mathbf{s}_{21}$ and $\mathbf{q}_1$ up to some scale factors. We thus can compute $k\, F_{10}^T = F_{01} = \tilde{\mathbf{s}}_{01} \cdot S_{01}$ and $k\, F_{12}^T = F_{21} = \tilde{\mathbf{s}}_{21} \cdot S_{21}$ but also $\mathbf{s}_{12}$ and S_{12} since :

$$\left\{ \begin{array}{rcl} 0 & = & F_{21} \cdot \mathbf{s}_{12} \\ S_{12} & = & k \frac{\tilde{\mathbf{s}}_{12}}{\|\mathbf{s}_{12}\|^2} \cdot S_{21}^T \cdot \tilde{\mathbf{s}}_{21} \end{array} \right. \tag{36}$$

Now it is quite painful to establish, but easy to verify, that :

$$\begin{array}{rcl} \mathbf{s}_{02} & = & S_{01} \cdot \mathbf{s}_{12} + (\mathbf{q}_{012}^T \cdot \mathbf{s}_{12})\, \mathbf{s}_{01} \\ F_{02} & = & \tilde{\mathbf{s}}_{02} \cdot \left[S_{01} \cdot S_{12} + \mathbf{s}_{01} \cdot \left[S_{12}^T \cdot \mathbf{q}_{012} + \frac{\mathbf{s}_{21}}{\mathbf{s}_{21}^T \cdot \mathbf{s}_{21}} \right]^T \right] \end{array} \tag{37}$$

These equations define F_{02} up to a scale factor, and are coherent because (1) S_{01} and $\mathbf{s}_{01}$ are recovered up to the same scale factor and (2) S_{12} and $\frac{\tilde{\mathbf{s}}_{21}}{\mathbf{s}_{21}^T \cdot \mathbf{s}_{21}}$ also. Therefore, we can obtain the 3 fundamental matrices knowing the motion parameters estimated from line correspondences.

Reciprocally, knowing the 3 fundamental matrices allows to recover, for $(i,j) \in \{0,1,2\}^2$, S_{ij}, $\mathbf{s}_{ij}$, up to some scale factors. Now, using equation (17), we can recover $\mathbf{q}_{012}$, as expected. Therefore we can compute the 18 parameters related the motion of lines from those related to the motion of points and do the reverse computation also.

As a consequence, when computing motion parameters from point correspondences and line correspondences, the same set of 18 parameters is estimated. **End Of Proof**

It is not an exaggeration to say that this proof, although perfectly valid, is painful and not very instructive. On the contrary, a geometrical construction will leads to the same conclusion, in a much more elegant way. However, we will see that this construction was mandatory to derive an effective algorithm of estimation.

Anyway, we can conclude that, considering three views and line and point correspondences :

1. The 3 fundamental matrices can be recovered considering either point correspondences or line correspondences or both. More precisely, the following minimum numbers of tokens in generic positions needed for the estimation are[9] :

Number of Lines	0	1	2	3	4	5	6	7	8	9
Number of Points	6	$6 - \frac{2}{3}$	$5 - \frac{1}{3}$	4	$4 - \frac{1}{3}$	$4 - \frac{2}{3}$	3	$2 - \frac{2}{3}$	$1 - \frac{1}{3}$	0

   and we note that,surprisingly perhaps, a line correspondence leads to 2 equations only, whereas a point correspondence leads to 3 equations, all this in the general case.

[9]The fractional numbers mean that among the three equations provided by a each point correspondence, one or two is not mandatory.

2. The 2 Q-matrices cannot be recovered considering either point correspondences, line correspondences or both. This is not surprising since this quantity corresponds to the structure of the plane at infinity which is not known from motion, but depends on the affine calibration of the system. More precisely, we have the following hierarchy of parameters :

Projective Motion Computation	P=18 parameters ($P = 11\ N - 4$ for $N + 1$ views)
Affine Motion Computation	A=P+3 parameters
Euclidean Motion Computation	M=A+5 parameters

The 3 and 5 "missing" parameters in the affine and Euclidean representation have the following geometrical interpretation :

- The affine calibration corresponds to the identification of the plane at infinity. A plane is defined by 3 parameters, as expected. This is represented by the r-vector in our equations. As soon as this quantity is known in one view, it can be predicted in the whole image sequence.
- The Euclidean calibration corresponds to the identification of the image of the absolute conic in the plane at infinity. The planar equation of a conic is defined by 5 parameters, as expected. This is represented by the K-matrix in our equations. As soon as this quantity is known in one view, it can be predicted in the whole image sequence. Knowing the K-matrix the r-vector can be computed, we thus do not have $5 + 3$ independent parameters.

About linear algorithms for the motion of points and lines. Let us improve our knowledge about the algebraic nature of this problem. Equation (18) can easily be written in the form[10] :

$$\forall l \in \{1,2,3\}, 0 = \sum_{h,i,j,k} n_1^h\, \epsilon_{hil}\, T^i_{jk}\, n_0^j\, n_2^k \tag{38}$$

By doing this, we make explicit the fact that the parameters related to the motion of lines can be collected in a $3 \times 3 \times 3$ tensor of 27 components.

In the case of points, considering two views, the similar algorithm is to estimate the nine components of the fundamental matrix as given is equation (9) under the constraints that F_{ij} is defined up to a scale factor, while $det(F_{ij}) = 0$ [18].

If we write : $Q_{ij} = \left(q^1_{ij}, q^2_{ij}, q^3_{ij}\right)$, i.e. explicit the columns of the Q-matrix this tensor can be seen as the concatenation of three matrices T^1, T^2, T^3 with :

$$T^i = q^i_{01} \cdot \mathbf{s}^T_{21} - \mathbf{s}_{01} \cdot q^{iT}_{21} \tag{39}$$

This equation has been simultaneously found by [11].

Of course, this tensor is subject to $27 - 18 = 9$ constraints which can be easily found, by summarizing the previous results to this new notation :

- (1) T is defined up to a scale factor,
- (2,3,4) for each matrix T^i, $det(T^i) = 0$,

---

[10]The Eddington (or Levi-Civita) symbol is defined by the following relations

$$\epsilon_{ijk} = \begin{cases} 0 & if \quad i=j \quad or \quad i=k \quad or \quad j=k \\ 1 & if \quad i<j<k \quad or \quad k<i<j \quad or \quad j<k<i \\ -1 & if \quad i<k<j \quad or \quad k<j<i \quad or \quad j<i<k \end{cases}$$

It is an antisymmetric tensor of type 3 and obviously $\mathbf{w} = \mathbf{u} \wedge \mathbf{v}$ can be computed as $\mathbf{w}_k = \sum_{i,j} \epsilon_{ijk} \mathbf{u}^i \mathbf{v}^j$, while $\tilde{u}_{jk} = \sum_i \epsilon_{ijk} \mathbf{u}^i$.

- (5) let $\mathbf{f}_{01}^i, i = 1,2,3$ be the 3 vectors of the kernel of ${T^i}^T$, i.e. $\mathbf{f}_{01}^{iT} \cdot T^i = 0$: these three vectors are coplanar, i.e. $|\mathbf{f}_{01}^1, \mathbf{f}_{01}^2, \mathbf{f}_{01}^3| = 0$; moreover we have $\mathbf{s}_{01}^T \cdot \mathbf{f}_{01}^i = 0$ which allows to determine $\mathbf{s}_{01}$. Moreover, the fundamental matrix is directly related to these vectors since, from equation (39) :

$$F_{01} = \left(\underbrace{\mathbf{s}_{01} \wedge \mathbf{q}_{01}^1}_{\lambda_1 \mathbf{f}_{01}^1} \quad \underbrace{\mathbf{s}_{01} \wedge \mathbf{q}_{01}^2}_{\lambda_2 \mathbf{f}_{01}^2} \quad \underbrace{\mathbf{s}_{01} \wedge \mathbf{q}_{01}^3}_{\lambda_3 \mathbf{f}_{01}^3} \right)$$

  This equation is due to [11][11].

- (6) let $\mathbf{f}_{21}^i, i = 1,2,3$ be the 3 vectors of the kernel of T^i, i.e. $T^i \cdot \mathbf{f}_{21}^i = 0$: these three vectors are coplanar. i.e. $|\mathbf{f}_{21}^1, \mathbf{f}_{21}^2, \mathbf{f}_{21}^3| = 0$; moreover we have $\mathbf{s}_{21}^T \cdot \mathbf{f}_{21}^i = 0$ which allows to determine $\mathbf{s}_{21}$.

- (7,8,9) for each matrix T^i, $\mathbf{s}_{01}^T \cdot T^i \cdot \mathbf{s}_{21} = 0$.

This result also allows us to recover, knowing the tensor T the components of the Qs-representation, and yields the following linear algorithm :

1. Given at least 13 lines, estimate the tensor T up to a scale factor, using equation (38) as a linear measurement equation.

2. For each matrix T^i find the closest matrix T^i which determinant is zero. This can be easily done using a singular value decomposition of T^i [12]. The vectors $\mathbf{f}_{01}^i$ and $\mathbf{f}_{21}^i$ define the kernel of T^i and ${T^i}^T$and are thus given by the computation of footnote (12).

3. Apply the same procedure on the matrices $F_0 = (\mathbf{f}_{01}^1, \mathbf{f}_{01}^2, \mathbf{f}_{01}^3)$ and $F_2 = (\mathbf{f}_{21}^1, \mathbf{f}_{21}^2, \mathbf{f}_{21}^3)$, since these two matrices must have a zero determinant, because the three vectors are coplanar. Computes the kernel of F_0, generated by $\mathbf{s}_{01}$ and the kernel of F_2, generated by $\mathbf{s}_{21}$. These two vectors are obtained, at this stage, up to a scale factor.

4. Compute the $\mathbf{p}_{01}^i = \lambda^0 (q_{01}^i + \lambda^i \mathbf{s}_{01})$ and $\mathbf{p}_{21}^i = \lambda^0 (q_{21}^i + \lambda^i \mathbf{s}_{21})$ from the linear equations :

$$\left\{ \begin{array}{rcl} \lambda^0 T^i \cdot \mathbf{s}_{21} & = & \mathbf{s}_{21}^T \cdot \mathbf{s}_{21}\, q_{01}^i - \mathbf{s}_{01} \cdot \mathbf{s}_{21}^T \cdot q_{21}^i \\ \lambda^0 {T^i}^T \cdot \mathbf{s}_{01} & = & \mathbf{s}_{21} \cdot \mathbf{s}_{01}^T \cdot q_{01}^i - \mathbf{s}_{01}^T \cdot \mathbf{s}_{01}\, q_{21}^i \end{array} \right.$$

   in which $\lambda^i, i = 0..3$ are undefined.

---

[11]The author uses this relation to recover the fundamental matrix. Since he obtains the column vectors of F_{01} up to a different scale factor for each, he briefly mentions a method which involves the computation of F_{01}^T, thus requires also the computation of two other tensors, while the present method will provide a direct estimate of two fundamental matrices, using only one linear estimation. The difference is due to the fact we have made explicit the algebraic constraints verified by the T-tensor.

[12] Let us write :

$$M = \left[\underbrace{(\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3)}_{U} \right] \cdot \begin{pmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \sigma_3 \end{pmatrix} \cdot \left[\underbrace{(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)}_{V} \right]^T$$

with $U \cdot U^T = U^T \cdot U = I$, $V \cdot V^T = V^T \cdot V = I$ and $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq 0$. This decomposition, called *singular value decomposition*, is unique and can be efficiently estimated numerically [21]. The reader can easily check that the matrix $M^* = U \cdot \begin{pmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot V^T$ verifies $det(M^*) = 0$ and minimizes $||M - M^*||$ considering the norm $||M|| = sup_{||\mathbf{u}||=1} ||M \cdot \mathbf{u}||$. Moreover $\mathbf{u}_3$ is the vector generating the kernel of ${M^*}^T$, i.e. $\mathbf{u}_3^T \cdot M^* = 0$ and $\mathbf{v}_3$ is the vector generating the kernel of M^*, i.e. $M^* \cdot \mathbf{v}_3 = 0$

5. From $S_0 = (\mathbf{p}_{01}^1, \mathbf{p}_{01}^2, \mathbf{p}_{01}^3)$ and $S_2 = (\mathbf{p}_{21}^1, \mathbf{p}_{21}^2, \mathbf{p}_{21}^3)$ which is obtained up to a scale factor, common to all the three columns vectors, we can estimate S_{01} and S_{21} up to a scale factor, thus F_{01} and F_{21} from equation (16). And we obtain also $\mathbf{q}_{012}$ up to a scale factor since we have :

$$S_0 = \lambda^0 \left[S_{01} + \mu\, \mathbf{s}_{01} \cdot \mathbf{q}_{012}^T\right] \quad ; \quad S_2 = \lambda^0 \left[S_{21} - \mu\, \mathbf{s}_{21} \cdot \mathbf{q}_{012}^T\right]$$

We first must notice that this "linear" algorithm does not require more lines than in the case where calibration is given [15].

However, as in [15], we have no guaranty that the estimation of T will verify the expected constraints, given a set of uncertain measurements. But the way we have enforced T to verify the constraints is not innocent. We have acted by successive reprojections, i.e. we have looked for "the closest" value which verifies the constraints. More generally, it has been demonstrated that an alternative to the non-linear statistical estimation of a quantity is to perform a submersion of the parameter space in a Euclidean space, in order to get linear equations, then find the best estimate in the Euclidean space under the constraints defining this parameter space [1]. This mechanism is equivalent to first find an unconstrained estimate, then reproject on the parameter space, as done here. We thus have an application of the general schema given in [1].

However, beside this practical application, the representation in terms of the tensor T allows us to establish that at least 18 parameters are required to parametrize our representation, as done in the next section.

Let us now compare with the case of points. Equation (9) is linear with respect to the components of the F-matrix, and having an estimate of the F-matrix from at least 8 point correspondences, we can find the closest matrix which determinant is zero as it is explained in footnote (12). However, in the case of more than 2 views, the F-matrices are not independent and integrating the constraints given in equation (14) is far from being obvious. In fact, we conjecture that when using linear algorithms to estimate motion from point correspondences in a sequence of $N+1$ views, the $7\,\frac{N(N+1)}{2} - (11\,N - 4)$ constraints are untractable because relations between F-matrices in the image sequence occur.

This was not the case when using only line correspondences, since the S-matrices, s-vectors and q-vectors are only locally constrained but not constrained all over the image sequence. For instance, the previous algorithm manages these constraints. However, if we want to use *both point and line correspondences* linear algorithms must be avoided. Let us now turn to non-linear algorithms, and -in parallel- terminate the analysis of our equations.

A parametric representation for the motion of lines and/or points. Let us now turn to the reverse problem, that is the fact we need at least 18 parameters for our representation. To show this point, we are going to construct a local parameterization of our representation. This parameterization will be very useful in some implementations, since it will allow to build a chart of the set of unknowns.

Let $\Phi = (\theta_0, \phi_0, a_0, b_0, c_0, d_0, e_0, f_0, \theta_2, \phi_2, a_2, b_2, c_2, d_2, e_2, f_2, \alpha, \beta)^T$ be a set of 18 numbers

such that :

$$\left\{\begin{array}{rcl}
\mathbf{s}_{01} & = & (\cos(\theta_0)\cos(\phi_0), \sin(\theta_0)\cos(\phi_0), \sin(\phi_0))^T \\
\mathbf{Q}_{01} & = & \underbrace{\left[\begin{pmatrix}\cos(\theta_0)\sin(\phi_0)\\ \sin(\theta_0)\sin(\phi_0)\\ -\cos(\phi_0)\end{pmatrix}\cdot\begin{pmatrix}a_0\\ b_0\\ c_0\end{pmatrix}^T + \begin{pmatrix}-\sin(\theta_0)\\ \cos(\theta_0)\\ 0\end{pmatrix}\cdot\begin{pmatrix}d_0\\ e_0\\ f_0\end{pmatrix}^T\right]}_{S_{01}} + \mathbf{s}_{01}\cdot\mathbf{q}_{012}^T \\
\mathbf{s}_{21} & = & (\cos(\theta_2)\cos(\phi_2), \sin(\theta_2)\cos(\phi_2), \sin(\phi_2))^T \\
\mathbf{Q}_{21} & = & \underbrace{\left[\begin{pmatrix}\cos(\theta_2)\sin(\phi_2)\\ \sin(\theta_2)\sin(\phi_2)\\ -\cos(\phi_2)\end{pmatrix}\cdot\begin{pmatrix}a_2\\ b_2\\ c_2\end{pmatrix}^T + \begin{pmatrix}-\sin(\theta_2)\\ \cos(\theta_2)\\ 0\end{pmatrix}\cdot\begin{pmatrix}d_2\\ e_2\\ f_2\end{pmatrix}^T\right]}_{S_{21}} - \mathbf{s}_{21}\cdot\mathbf{q}_{012}^T \\
\mathbf{q}_{012} & = & (\cos(\alpha)\cos(\beta), \sin(\alpha)\cos(\beta), \sin(\beta))^T
\end{array}\right. \tag{40}$$

It is easy to verify that $||\mathbf{s}_{01}|| = 1$, $||\mathbf{s}_{21}|| = 1$, $||\mathbf{q}_{012}|| = 1$. These three constraints avoid the three scale indetermination, found in Prop. 9. We also have $\mathbf{s}_{ij}^T \cdot S_{ij} = 0$ as required.

This parameterization has been constructed from a well known representation of unary vectors which is of the form : $\mathbf{u} = (\cos(\alpha)\cos(\beta), \sin(\alpha)\cos(\beta), \sin(\beta))^T$, singular if and only if $\mathbf{u} = (0,0,1)^T$. In such a case it is straightforward to find another parameterization by a permutation of two coordinates. This allows to build a chart of the sphere with two maps. A similar construction, modifying the parameterization of $\mathbf{s}_{01}$, $\mathbf{s}_{21}$, $\mathbf{q}_{012}$ when these vectors are close to $(0,0,1)^T$ allow us to construct a chart of our representation using $2 \times 2 \times 2 = 8$ maps.

When correspondences between points are given, the following parameterization also applies to F-matrices, since we have :

$$\begin{array}{rcl}
F_{01} & = & \left[\begin{pmatrix}-\sin(\theta_0)\\ \cos(\theta_0)\\ 0\end{pmatrix}\cdot\begin{pmatrix}a_0\\ b_0\\ c_0\end{pmatrix}^T - \begin{pmatrix}\cos(\theta_0)\sin(\phi_0)\\ \sin(\theta_0)\sin(\phi_0)\\ -\cos(\phi_0)\end{pmatrix}\cdot\begin{pmatrix}d_0\\ e_0\\ f_0\end{pmatrix}^T\right] \\
F_{21} & = & \left[\begin{pmatrix}-\sin(\theta_2)\\ \cos(\theta_2)\\ 0\end{pmatrix}\cdot\begin{pmatrix}a_2\\ b_2\\ c_2\end{pmatrix}^T - \begin{pmatrix}\cos(\theta_2)\sin(\phi_2)\\ \sin(\theta_2)\sin(\phi_2)\\ -\cos(\phi_2)\end{pmatrix}\cdot\begin{pmatrix}d_2\\ e_2\\ f_2\end{pmatrix}^T\right]
\end{array}$$

while the parameterization of F_{02} is obtained from :

$$\mathbf{s}_{12} = k\begin{pmatrix}a_2\\ b_2\\ c_2\end{pmatrix} \wedge \begin{pmatrix}d_2\\ e_2\\ f_2\end{pmatrix} \quad ; \quad S_{12} = k'\,\tilde{\mathbf{s}}_{12}\cdot F_{21}^T \tag{41}$$

and using equation (37). However, the last matrix F_{02} is quite a huge expression because, as for the linear algorithm, it is quite heavy to constraint these three interdependent F-matrices.

It might be surprising that in eq. (41) the F-matrix appears as being parameterized by 8 parameters, when 7 parameters are expected. In fact, this is $(a_0, b_0, c_0, d_0, e_0, f_0)$ are defined up to a scale factor, and this scale factor is determined by the fact we have constrained $||\mathbf{q}_{012}|| = 1$. It ensures that $det(F_{ij}) = 0$. Anyway, it is clear that this parameterization is not unique. We have make this choice for some reasons of simplicity, and we will use a very similar parameterization in the final implementation.

Despite this problem, we can build "non-linear" algorithms to estimate our representation. Given a set of measurements, we try to find the vector $\mathbf{\Phi}$ which minimizes the sum of any positive function of the measurement error (least-squares, robust estimators, etc...). If one component of the representation becomes close to a singularity, the algorithm must perform a "change of map". This is a very common procedure (see for instance [4], for vision applications), so not detailed here.

We now can express the tensor T as a function of Φ, and the three fundamental matrices also. Doing this, we obtain :

Proposition 12 *The rank of the Jacobian of the functions :*

$$\begin{array}{rccc} \phi : & \mathcal{R}^{18} & \rightarrow & \mathcal{R}^{27} \\ & \Phi & \hookrightarrow & T(\Phi) \end{array}$$

and

$$\begin{array}{rccc} \psi : & \mathcal{R}^{18} & \rightarrow & \mathcal{R}^{27} \\ & \Phi & \hookrightarrow & \left\{ \begin{array}{c} F_{01}(\Phi) \\ F_{21}(\Phi) \\ F_{02}(\Phi) \end{array} \right\} \end{array}$$

are 18 *almost everywhere.*

Do not panic. The proof has not been established by hand, but using the following piece of MapleV.2 code. The quantities Q_{01}, s_{01}, Q_{21}, s_{21}, q_{012}, F_{21}, F_{01} and F_{02} have been defined using the previous equations. We obtain :

```
# Definition of T as a 27 components vector
T:=array([''Q01[j,i]*s21[k]-Q21[k,i]*s01[j]'$i=1..3'$j=1..3'$k=1..3]):
# Definition of the vector of parameters Phi
Phi:=array([alpha,beta,theta0,phi0,a0,b0,c0,d0,e0,f0,
 theta2,phi2,a2,b2,c2,d2,e2,f2]):
# Computes the Jacobian of the function
J:=map(proc(x) simplify(x) end, jacobian(T,Phi)):
# Computes the rank of this matrix for a subset of points
#  (we restrict to this set, for Maple to be able to calculate)
r:=rank(map(simplify,subs({theta0=0,phi0=0,theta2=0,phi2=0,alpha=0,beta=0},
 op(J))));
                                    18
# Definition of F01-F21-F02 as a 27 components vector
F:=array([''F01[i,j]'$i=1..3'$j=1..3,''F21[i,j]'$i=1..3'$j=1..3,
 ''F02[i,j]'$i=1..3'$j=1..3]):
# Computes the Jacobian of the function
J:=map(proc(x) simplify(x) end, jacobian(F,Phi)):
# Computes the rank of this matrix for a subset of points
r:=rank(map(simplify,subs({theta0=0,phi0=0,theta2=0,phi2=0,alpha=0,beta=0},
 op(J))));
                                    18
```

This formal computation establishes that, for a subset of the map, the two functions have a rank equal to 18. Then :

Proof Since the rank is 18 in a particular subset, it cannot be less than 18 everywhere. So the rank at a generic point is at least 18. Of course, for some particular values of the parameters, a lower rank can occur. But these singularities, if generic, are a set of null measure. So, the result is true, almost everywhere. **End Of Proof**

This last proposition is a complement of Prop. 10 and Prop. 9 and demonstrates the reverse.

2.6 Conclusion on the three views motion problem

We now have the certitude that exactly 18 parameters are needed to estimate the motion of lines and/or points, 3 views being given, in the general case, where no calibration is given.

When establishing this result, we have also build two algorithms (one linear, one with a minimum set of parameters) to estimate this representation. We have made obvious the fact that it is very difficult to manage linear algorithms in this case, because the estimated parameters

verify a important number of complicated constraints. Previous studies on the computation of motion from point or line correspondences, when no calibration, do not integrate all these constraints [6, 11] although the existence of these constraints has been already established and used for point correspondences [18].

Although a non-linear parameterization is easily usable for three views, its generalization to a sequence of views is not obvious since token correspondences between non-consecutive frames must be taken into account, as established for three views. This means that, at the implementation level, we must try to avoid this level of complexity and build the algorithms on local correspondences only. This will be discussed in the last section.

But we had two other goals, considering this algebraic approach, one good, one bad. The good one was to obtain a much better understanding of several aspects of the motion of lines and points in the uncalibrated case.

The bad one was to illustrate how algebra developments are boring, as the reader must be convinced of, now. The situation is even more complicated, because we have omitted a lot of intermediate but trivial calculations, focusing on the results. Is there any possibility to avoid this tragedy ?

There is. No algebra any more, just geometry.

3 The motion of lines and points : geometric interpretation

In this section, although we will try to maintain the notations as coherent as possible with the previous section, we define again each object, and will not rely on any existing result.

Notations : we use the operator $+$ to define the geometrical object generated by two or three objects, i.e. if P, Q and R are three points and D is a line $P+Q$ is the line generated by P and Q, $P+Q+R$ the plane generated by this three points and $P+D$ the plane generated by the point and the line; we use the operator $\cap$ for intersection, for instance $D \cap P$ is a point, intersection of the line D and the plane P. The equality between two quantities is written $\equiv$ since it corresponds to an algebraic equality, up to a scale factor, between homogeneous vectors.

From the epipolar to the trifocal geometry. Let us consider a system of three cameras, in *general position* as shown in figure 2. Each camera is defined by its optical center C_i, $i = \{0,1,2\}$ and its retinal plane R_i. We use the usual pinhole model for the camera.

Let us now consider the plane $\mathcal{T} \equiv C_0 + C_1 + C_2$. This plane is called *the trifocal plane.* It has been materialized using dashed lines in figure 2. This plane intersects the retinal planes R_i along lines $t_i \equiv \mathcal{T} \cap R_i$ called *the trifocal lines.* The intersection of the line $C_i + C_j$ with each retinal plane defines *the epipoles* $\mathbf{s}_{ij} \equiv (C_i + C_j) \cap R_i$ and *the tripoles* $\mathbf{s}_{kk} \equiv (C_i + C_j) \cap R_k$, when $i \neq k \neq j$. For instance, looking at figure 2 and considering the line $C_0 + C_2$ one can see that the intersection with the retinal planes R_1, R_0 and R_2 are : $\mathbf{s}_{11}$ which is a tripole and $\mathbf{s}_{02}$, $\mathbf{s}_{20}$ which are two epipoles.

From this construction it appears that all **the trifocal lines, the epipoles and the tripoles belong to the trifocal plane.** Moreover, it is straightforward to establish, combining the previous definitions, that in **each retinal plane the epipoles and the tripole belong to the trifocal line**, or, using our notations, $\mathbf{s}_{ij} \in d_j$ for $i = \{0,1,2\}$. In particular :

$$\forall i \in \{0,1,2\} \quad , \quad t_i \equiv \mathbf{s}_{i1} + \mathbf{s}_{i0} + \mathbf{s}_{i2} \quad ; \quad |\mathbf{s}_{i1}, \mathbf{s}_{i0}, \mathbf{s}_{i2}| = 0 \tag{42}$$

and these three points form a projective basis of the trifocal line. As a consequence, the trifocal lines are defined by the knowledge of the epipoles.

Of course, the epipoles defined here correspond to the epipoles defined, in the case of the motion of points, in the epipolar geometry.

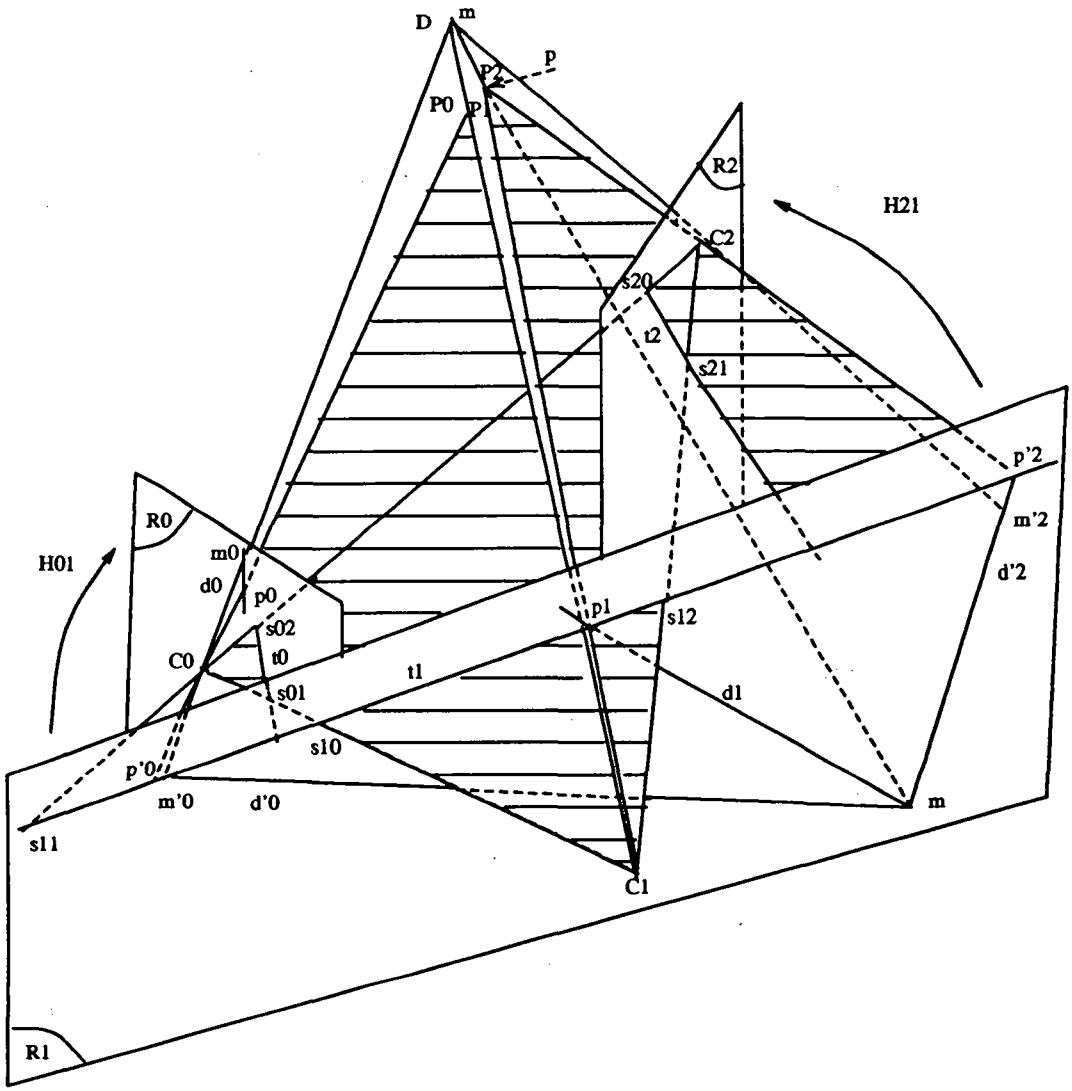

Figure 2: The trifocal geometry, see text for a complete description.

Let us finally explicit useful transformations from one retinal plane to another. We consider the special collineation, H_{ij}, from the retinal plane R_j on the retinal R_i which is a perspectivity through the optical center C_i. By definition, the image of a point $m_j \in R_j$ is the point m_i, intersection of the line $m_j + C_i$ with the retinal plane R_i, i.e. $m_i \equiv R_i \cap (C_i + m_j) \equiv H_{ij} m_j$. It induces a dual transformation H_{ij}^T which maps lines of R_i onto lines of R_j. By definition, the image d_j of a line d_i in R_i is the set of points of R_j which images through H_{ij} belong to d_i, i.e $d_j \equiv H_{ij}^T d_i$.

The dual of this perspectivity can be decomposed as follow. Given any plane $\Pi^\bullet$ containing C_i and any pairs of lines $d_i \in R_i$ and $d_j \in R_j$ with $d_j \equiv H_{ij}^T d_i$ the present perspectivity induces a collineation $h^\bullet$ between $t_i^\bullet \equiv \Pi^\bullet \cap R_i$ and $t_j^\bullet \equiv \Pi^\bullet \cap R_j$ by taking the restriction of the application to these two lines, i.e. for $p_i \in t_i^\bullet$ and $p_j \in t_j^\bullet$: $p_j \equiv (H_{ij}(p_i + q)) \cap t_j^\bullet$, q being a generic point not in $t_i^\bullet$. Reciprocally, given two planes Π^a and Π^b containing C_i and two collineations h^x, $x = \{a, b\}$ from $t_i^x \equiv \Pi^x \cap R_i$ to $t_j^x \equiv \Pi^x \cap R_j$ the dual of the perspectivity is entirely defined since $d_j \equiv H_{ij}^T d_i \equiv (h^a(d_i \cap t_i^a)) + (h^b(d_i \cap t_i^b))$.

In our situation, if we consider H_{01} and H_{21}, it appears that to decompose these two per-

spectivities we can take the trifocal plane Π and any other plane $\Pi^\lambda \neq \Pi$ containing both C_0 and C_2. This corresponds to choosing a plane Π_λ in the pencil of plane containing $C_0 + C_2$, thus defined by 1 parameter. With these choices, the two perspectivities are entirely defined by 4 collineations $h_{01} : t_0 \rightarrow t_1$, $h_{21} : t_2 \rightarrow t_1$, $h_{01}^\lambda : t_0^\lambda \rightarrow t_1$, $h_{21}^\lambda : t_2^\lambda \rightarrow t_1^\lambda$ given as above and involving 3 parameters each. Therefore, H_{01} *and* H_{21} *are defined by* $1 + 4 \times 3 = 13$ *parameters.*

Because the epipoles and tripoles belong to lines passing through the optical center, and because the trifocal lines contain those epipoles, it is straightforward to see that these quantities are related through the previous perspectivity. If we consider H_{01} and H_{21} we obtain immediately :

$$\begin{array}{rcl} \mathbf{s}_{01} & \equiv & H_{01}\mathbf{s}_{10} \\ \mathbf{s}_{21} & \equiv & H_{21}\mathbf{s}_{12} \\ \mathbf{s}_{02} & \equiv & H_{01}\mathbf{s}_{11} \\ \mathbf{s}_{20} & \equiv & H_{21}\mathbf{s}_{11} \\ t_1 & \ni & H_{01}^{-1}\mathbf{s}_{00} \\ t_1 & \ni & H_{21}^{-1}\mathbf{s}_{22} \end{array} \tag{43}$$

from which we deduce :

$$\begin{array}{rcl} t_1 & \equiv & H_{01}^T t_0 \\ t_1 & \equiv & H_{21}^T t_2 \end{array} \tag{44}$$

Epipolar correspondences for a 3D point. Because we have to analyze also the motion of points, let us briefly review the epipolar geometry. We consider the correlation F_{ij} which maps a point m_j in the retinal plane R_j onto a (epipolar) line d_i in R_i. This line is the intersection of the (epipolar) plane, containing the optical ray $m_j + C_j$ and the other optical center C_i, with the retinal plane R_i, i.e. $d_j \equiv (m_j + C_j + C_i) \cap R_i$. This corresponds to the projection of the optical ray in view j onto the retinal plane i, and this transformation is obviously undefined for any points on $C_i + C_j$ such as epipoles and tripoles. From this construction, we easily see that the epipolar line defined by a point m_j contains (1) the epipole $\mathbf{s}_{ij}$ since it is the projection of the optical center in R_i, i.e. $\mathbf{s}_{ij} = (C_j + C_i) \cap R_i$, (2) and the inverse image of m_j through the perspectivity H_{ji}. We thus have :

$$F_{ij}m_j \equiv H_{ji}^{-1}m_j + \mathbf{s}_{ij} \tag{45}$$

and we can easily verify that for any point of the trifocal line :

$$m \in t_j \Rightarrow t_i \equiv F_{ij}m$$

while by construction $F_{ij} \equiv F_{ji}^T$.

The projection of a point $m_i \in R_i$ must belong to the epipolar line defined by its correspondent m_j which yields the fundamental equation :

$$m_i \in F_{ij}m_j$$

Trifocal correspondences for a 3D line. Let us consider a 3D-line D, as shown also in figure 2. The projections d_i of this line onto the three retinal planes $R_i, i = 1, 2, 3$ is defined by the intersection of the "projection" plane $P_i \equiv D + C_i$, defined by the 3D-line D and the optical center C_i, with the retinal plane R_i, i.e. $d_i \equiv P_i \cap R_i$.

Let us construct the projection of the line in one retinal plane, say R_1, from the projections in the two others.

Because, by definition, these three planes meet in a line, presently D, their intersection with any plane P is a set of 3 concurrent lines, the intersection point being $m \equiv P \cap D$. For instance if we consider the retinal plane R_1, the intersections of the three projection planes with R_i are $d_0' \equiv R_1 \cap P_0$, $d_1 \equiv R_1 \cap P_1$ and $d_2' \equiv R_1 \cap P_2$. These three lines thus meet at a point m. Moreover,

by construction, d'_0 is the image of d_0 through the dual perspectivity H_{01}^T, i.e. $d'_0 \equiv H_{01}^T d_0$, and d'_2 is the image of d_2 through the dual perspectivity H_{21}^T, i.e. $d'_2 \equiv H_{21}^T d_2$. We can express the fact these three lines are concurrent through :

$$|d_1, H_{21}^T d_2, H_{01}^T d_0| = 0$$

This equation is nothing else than the first equation of Liu and Huang [15] on line motion, but given now in a general case, as in equation (35).

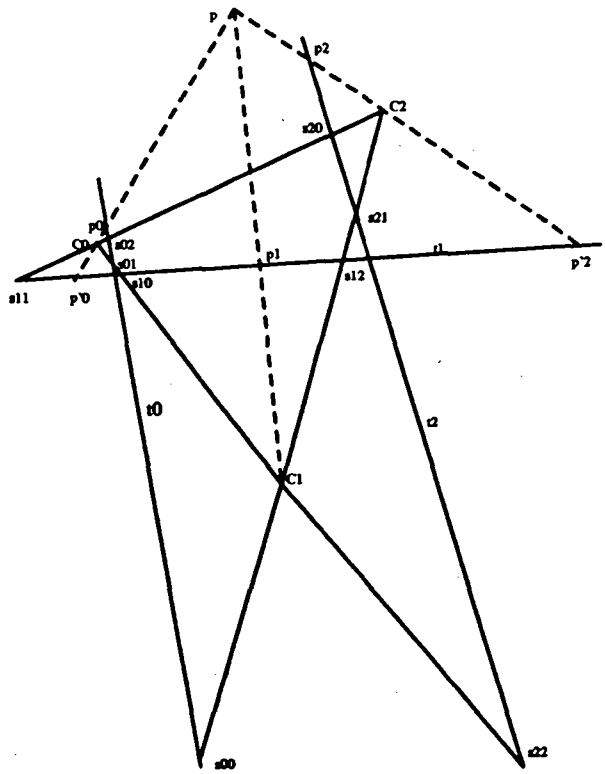

Figure 3: The trifocal plane geometry, see text for a complete description.

On one hand, we have a point m which belongs to d_1, on the other hand, we can also calculate the intersection, p_1, of d_1 with the trifocal line t_1, since we can derive the following relations considering the geometry of trifocal plane, see figure 3 :

$$\begin{aligned}
p'_0 &\equiv (H_{01}^T d_0) \cap t_1 \\
p'_2 &\equiv (H_{21}^T d_2) \cap t_1 \\
C_0 &\equiv (s_{02} + s_{20}) \cap (s_{01} + s_{10}) \\
C_1 &\equiv (s_{10} + s_{01}) \cap (s_{12} + s_{21}) \\
C_2 &\equiv (s_{21} + s_{12}) \cap (s_{20} + s_{02}) \\
p &\equiv (p'_0 + C_0) \cap (p'_2 + C_2) \\
p_1 &\equiv (C_1 + p) \cap t_1
\end{aligned}$$

all these relations being defined only considering the perspectivity H_{01}, the perspectivity H_{12} and the 6 epipoles.

By doing this we entirely defines d_1.

This equation is nothing else than the second equation of Liu and Huang [15] on line motion, but given now in a general case.

In the projective case, this corresponds to a trilinearity which relates the intersections of the lines d_1, d'_0 and d'_2 with the trifocal line t_1.

We can explicit this trilinearity if we choose coordinates for the quantity given in the retinal plane R_1. Let us take : $t_1 = (0,0,1)^T$, $\mathbf{s}_{1i} = (\sigma_i, 1, 0)^T$, $H_{i1}^T d_i = (-1, \nu_i, 0)^T$ and $p_i = (\nu_i, 1, 0)^T$. In that case equation (34) corresponds to the following relation :

$$\begin{pmatrix} \frac{\nu_0 - \nu_1}{\nu_0 - \sigma_0} & - & \frac{\nu_2 - \nu_1}{\nu_2 - \sigma_2} \\ & 0 & \\ & 0 & \end{pmatrix} = 0$$

which relates $\nu_i, i \in \{0,1,2\}$ through a trilinear relation, as expected.

The trifocal and epipolar geometries are defined by 18 parameters. From the previous developments we obtain immediately the following three results :

1. The correlations F_{ij} are entirely defined, through equation (45), by the two perspectivities and the epipoles.
2. The two perspectivities are defined by $2 \times 8 - 3 = 13$ parameters.
3. All epipoles and trifocal lines are defined, through equations (43), by the knowledge of the two perspectivities, the epipoles $\mathbf{s}_{10}$, $\mathbf{s}_{12}$, and the tripole $\mathbf{s}_{11}$.
4. The three points $\mathbf{s}_{10}$, $\mathbf{s}_{12}$ and $\mathbf{s}_{11}$ are defined by $3 \times 2 - 1 = 5$ parameters since these 3 points are collinear.

Finally, the trifocal and epipolar geometries are defined by $5 + 13 = 18$ parameters. What about a simpler result ?

4 Implementation and experimental results

4.1 Implementation of the motion module

Let us now discuss the implementation of a module which takes point and/or line correspondences as input, and output an estimation of the affine motion parameters.

Thank's to several other developments in the field such as [17, 30, 24, 20, 29] we do not have to discuss again how to implement such a module in great details but simply can base our work on previous experiences. The main features to be taken into account are the following :

- Point and line correspondences are always defined with a certain uncertainty, often represented by a covariance matrix, and estimation criteria must weight their estimates using this uncertainty.
- It is always more robust and reliable to have a criterion based on a retinal measurement error (i.e. a retinal disparity or a image related quantity), even in the uncalibrated case [17], because this quantity corresponds to the physical measure. Obviously, when the retinal disparity has been canceled, all information about the motion has been extracted.
- It is always possible - and sometimes more efficient [30, 12] - to compute motion *and structure* at the same time, instead of eliminating structure parameters to estimate motion and then structure from motion. We will follow this track here.

Using retinal disparities as measurement error. Let us consider a set of P point correspondences in a sequence of $N+1$ views. We index the points with $k \in \{1..P\}$ and the different views with $i \in \{0..N\}$.

The main source of error, considering such point correspondences, is related to the uncertainty in the pixel localization and the uncertainty in the correspondence, the latter being more important than the former. A simple way to represent this uncertainty is to consider for each

point m_i^k that the value of m_j^k is given with a certain uncertainty corresponding to a covariance $\Lambda_{m_j^k}$. The uncertainty is "thus carried by m_j^k", while m^k is taken as a deterministic value, neglecting its localization error in the image, i.e. $\Lambda_{m_i^k} = 0$ and $\Lambda_{m_j^k} = \Lambda_j^k$. Such a model corresponds, for instance, to the uncertainty of a correlation operator [5].

More precisely, when the covariance is used to weight the contribution of a local correspondence in a criterion, we use the inverse of the covariance, also called *information matrix* :

$$\left(\Lambda_j^k\right)^{-1} = \begin{pmatrix} I_{uu} & I_{uv} \\ I_{uv} & I_{vv} \end{pmatrix}$$

and, moreover, we not always assume that this symmetric positive matrix is definite i.e. invertible. On the contrary, if the retinal disparity is defined in only one direction, say $\mathbf{n}_j^k = (\cos(\varsigma), \sin(\varsigma))^T$, whereas in the orthogonal direction $\mathbf{t}_j^k = (-\sin(\varsigma), \cos(\varsigma))^T$ the retinal correspondence is undefined, we can integrate this constraint very easily by defining an information matrix of the form :

$$\left(\Lambda_j^k\right)^{-1} = \frac{1}{(\sigma_j^k)^2} \mathbf{n}_j^k \cdot \mathbf{n}_j^{k\,T} \tag{46}$$

which interpretation is that the variance of the measure in the direction where the retinal correspondence is undefined is undefined is infinite (notion of generalized inverse) [4].

We are going to apply this powerful mechanism now.

From line or curve correspondences to point correspondences. Let us consider a 2D-line d_i with equation $m \in d_i \Leftrightarrow 0 = \mathbf{n}_i^T \cdot m_i$ represented by an homogeneous vector $\mathbf{n}_i$ in correspondence with a 2D-line d_j represented by an homogeneous vector $\mathbf{n}_j$. We can reduce this equation on line correspondences to equations involving points correspondences, using Prop. 5, as developed now.

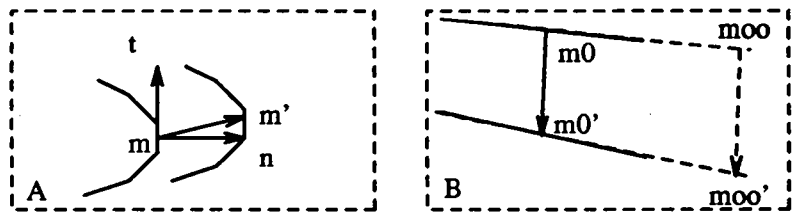

Figure 4: Using correspondences between non punctual primitives; A : correspondence related to the normal motion of a curve, B : correspondence related to a line segment

Let us choose two points m_0 and m_1 on this 2D-line. In practice, we are going to detect line-segments in an image. It has been shown [29] that we better choose the center of the segment m_0 and the direction of the segment, or point at infinity $m_1 = m_\infty$. This will minimize the covariance of the measures. However, any other pair of points could be used.

Each point is in correspondence with a point for which we cannot determine the exact location but only its location up to a displacement on the line, i.e. a "sliding point".

In other words, if we have a line correspondence, *constraining the collineation using the equation on the line parameter* $\mathbf{n}$ *or using the normal correspondences between two points on this line is equivalent.* We thus do not need to implement specifically the case of lines, but can limit our calculations to the case of points, with appropriate information matrices.

Similarly, a 2D-correspondence between a point m_i on a 2D-curve and another point m_j on the corresponding 2D-curve after a retinal displacement can be integrated in our framework. Along a curve, the tangential displacement is generally ambiguous (aperture problem) [4], except

for points with a high curvature (corners, junctions). Let $\mathbf{t}_j^k$, $||\mathbf{t}_j^k|| = 1$, be the tangent of the curve on m, and $\mathbf{n}_j^k$, $||\mathbf{n}_j^k|| = 1$, its normal, as illustrated in figure 4.

In this situation, the planar Euclidean displacement δm is obtained in the normal direction only, written $\beta = (\mathbf{n}_j^{k\,T} \cdot \delta m)$, whereas $(\mathbf{t}_j^{k\,T} \cdot \delta m)$ is undefined. The uncertainty for this equation is easily computable. Since only the disparity in the direction of $\mathbf{n}$ influences the measure, we can calculate the uncertainty as : $(\sigma_j^k)^2 = \mathbf{n}_j^{k\,T} \cdot \Lambda_j^k \cdot \mathbf{n}_j^k$, where Λ_j^k is the covariance of the correspondence.

We can integrate this situation in our framework, by simply choosing a information matrix, as given in eq. (46).

The case of line correspondences is a particular case for which only two correspondences are to be used.

As a conclusion, we do not need to integrate specifically line correspondences in our algorithm but can simply use point correspondences. In addition to that, we could also have integrated local correspondences along any curve as studied in [8].

Dealing with local correspondences. What we have implicitly assumed from now on is that each point has been matched *all along the sequence.* This is by no-means a realistic assumption and we must be able to integrate the fact that a point correspondence is valid only during a few frames.

This is very easy to do, using our information matrices again. Let us consider that the point correspondences have been established from view j_1 to view j_2 only, then we set $(\Lambda_j^k)^{-1} = 0$ for all $j < j_1$ and $j \geq j_2$ and the related equations, which are undefined, will not be taken into account because their weight is zero.

This allows to carry on our study, considering P points in the $N+1$ views but with suitable information matrices which integrate the fact that for each the correspondences are not established all along the sequence.

In any case, it has been established that we need at least three views for a correspondence to constraint the Qs-representation, if it corresponds to a sliding point on a line.

Parameterization of the uncalibrated motion and structure Let us consider again a sequence of $N+1$ views and P point-correspondences in this sequence.

We are going to represent the uncalibrated motion in the sequence by Q-matrices and s-vectors between *two consecutive frames*, i.e. $Q_{i+1\,i}$ and $\mathbf{s}_{i+1\,i}$. All other Q-matrices and s-vectors are entirely defined by these quantities, using eq. (10).

As discussed in this paper, the Qs-representation is not entirely determined in the image sequence but is only determined up to :

1. *A global scale factor for the image sequence.* We fix this scale factor by setting $||\mathbf{s}_{10}|| = \sigma_s$, i.e. by constraining the uncalibrated translation in the two first consecutive images.
   A reasonable default value is $\sigma_s = 1$. This corresponds to a reconstruction of the depths up to a scale factor as always the case in a monocular image sequence.
2. *One r-vector in the sequence.* We fix $\mathbf{r}_{10} = (\sigma_0, \sigma_1, \sigma_2)^T$.
   A reasonable default value is $(\sigma_0, \sigma_1, \sigma_2) = (0, 0, 1)$. This corresponds to assuming that the plane at infinity is fronto-parallel, an affine retinal frame of reference being taken into account.
3. *An expansion factor between each pair of consecutive frames.* We note the expansion factor between frame i and frame $i+1$: $\{\mathcal{Z}_{i+1\,i}\}_{i=0..N-1}$.
   A reasonable default value is $\forall i, \mathcal{Z}_{i+1\,i} = 1$, especially when the camera is not performing a zoom, since it corresponds to assuming that the relative scales between two consecutive frames are left unchanged.
   As a consequence of this last indetermination, we have seen that :
   - *The twelve components of each Q-matrix and s-vector are only defined up to a scale*

factor. In order to avoid this scale factor indetermination we will simply choose $Q^{22}_{i+1\,i} = 1$, i.e. fix the last component of each Q-matrix. This constraint is not far from the reality since in an image sequence we expect the rotational component of the motion between two frames to be rather small and the calibration matrix not to be much modified. As a consequence, the matrix $Q_{i+1\,i} = A_{i+1} \cdot R_{i+1\,i} \cdot A_i^{-1}$ is expected to be close to the identity.

We can either determine these parameters from other perceptual processes (Euclidean calibration, odometric cues, etc...) or set them to the proposed default values. In that case, we are not going to estimate the "true" parameters but the depths and displacements up to some affine and projective transformations as discussed in this paper.

These $N + 4$ parameters will be called *external parameters* in the sequel.

If we take these $N + 4$ parameters as inputs of our algorithm, we must parametrize our Qs-representation such that $\mathbf{Q}_{10}$ and $\mathbf{s}_{10}$ verify the previous constraints, and such that $Q^{22}_{i+1\,i} = 1$. Following these requirements, we can parametrize the motion as follows :

Let us consider the first view.

$$For \quad n = 0 \quad : \quad \mu_0 = (\theta_0, \phi_0, a_0, b_0, d_0, e_0, f_0)^T \in \mathcal{R}^7$$

such that :

$$\left\{ \begin{array}{lcl} \mathbf{s}_{10} & = & \sigma_s \, (\cos(\theta_0)\cos(\phi_0), \sin(\theta_0)\cos(\phi_0), \sin(\phi_0))^T \\ \mathbf{Q}_{10} & = & \underbrace{\left[\begin{pmatrix} \cos(\theta_0)\sin(\phi_0) \\ \sin(\theta_0)\sin(\phi_0) \\ -\cos(\phi_0) \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ b_0 \\ c_0 \end{pmatrix}^T + \begin{pmatrix} -\sin(\theta_0) \\ \cos(\theta_0) \\ 0 \end{pmatrix} \cdot \begin{pmatrix} d_0 \\ e_0 \\ f_0 \end{pmatrix}^T \right]}_{S_{10}} + \mathbf{s}_{10} \cdot \begin{pmatrix} \sigma_0 \\ \sigma_1 \\ \sigma_2 \end{pmatrix}^T \\ with & & c0 = \frac{\sigma_2 \sigma_s \sin(\phi_0) - 1}{\cos(\phi_0)} \end{array} \right.$$

as obtained using equation (40).

This parameterization does not cover all configurations and is undefined if $\mathbf{s}^0_{10} = \mathbf{s}^1_{10} = 0$. But if we use two other maps obtained by permuting the trigonometric factors in the definition of $\mathbf{s}_{10}$, we parametrize all configurations as already discussed in this paper.

This mechanism also provides a parameterization of the F-matrix, as been made explicit in eq. (41), and we have :

$$F_{10} = \left[\begin{pmatrix} -\sin(\theta_0) \\ \cos(\theta_0) \\ 0 \end{pmatrix} \cdot \begin{pmatrix} a_0 \\ b_0 \\ c_0 \end{pmatrix}^T - \begin{pmatrix} \cos(\theta_0)\sin(\phi_0) \\ \sin(\theta_0)\sin(\phi_0) \\ -\cos(\phi_0) \end{pmatrix} \cdot \begin{pmatrix} d_0 \\ e_0 \\ f_0 \end{pmatrix}^T \right]$$

As expected, $Q^{22}_{10} = 1$, $\mathbf{s}^T_{10} \cdot S_{10} = 0$, $||\mathbf{s}_{10}|| = \sigma_s$ and $\frac{\mathbf{Q}^T_{10} \cdot \mathbf{s}_{10}}{||\mathbf{s}_{10}||^2} = \mathbf{r}_{10} = (\sigma_0, \sigma_1, \sigma_2)^T$. We thus verify the required constraints. Please note that $c0$ as been chosen so that $Q^{22}_{10} = 1$.

Let us now consider the other views.

$$For \quad 0 < n < N \quad : \quad \mu_n = \begin{array}{c} (\mathbf{s}^0_{i+1\,i}, \mathbf{s}^1_{i+1\,i}, \mathbf{s}^2_{i+1\,i}, \\ Q^{00}_{i+1\,i}, Q^{01}_{i+1\,i}, Q^{02}_{i+1\,i}, Q^{10}_{i+1\,i}, Q^{11}_{i+1\,i}, Q^{12}_{i+1\,i}, Q^{20}_{i+1\,i}, Q^{21}_{i+1\,i})^T \in \mathcal{R}^{11} \end{array}$$

while again $Q^{22}_{i+1\,i} = 1$.

By doing this we use $7 + 11\ (N - 1) = 11\ N - 4$ parameters as expected and verify the constraints of this representation.

The structure of the scene is very simply parameterized by the inverse of the affine depths of the points, called *proximity*, .i.e :

$$\pi^k_i = \frac{1}{Z^k_i}$$

for a point $k \in \{1..p\}$ in frame $i \in \{0..N\}$.

A criterion to estimate the Qs-representation. Let us now define the measurement equations. As stated in this papers, all measurement equations, considering point correspondences, are given by eq. (11). In this equation we have decomposed the equations in (a) a prediction of the retinal disparity and (b) a prediction of the depth. This is exactly the decomposition we need, since we would like :

1. to minimize the retinal disparities, between consecutive frames :

$$\epsilon_i^k(\mu_i, \pi_i^k) = \left| \begin{array}{ll} u_{i+1} & - \frac{Q^{00}_{i+1\,i}\,u_i + Q^{01}_{i+1\,i}\,v_i + Q^{02}_{i+1\,i} + \frac{1}{Z_i^k}\,s^0_{i+1\,i}}{Q^{20}_{i+1\,i}\,u_i + Q^{21}_{i+1\,i}\,v_i + Q^{22}_{i+1\,i} + \frac{1}{Z_i^k}\,s^2_{i+1\,i}} \\ v_{i+1} & - \frac{Q^{10}_{i+1\,i}\,u_i + Q^{11}_{i+1\,i}\,v_i + Q^{12}_{i+1\,i} + \frac{1}{Z_i^k}\,s^1_{i+1\,i}}{Q^{20}_{i+1\,i}\,u_i + Q^{21}_{i+1\,i}\,v_i + Q^{22}_{i+1\,i} + \frac{1}{Z_i^k}\,s^2_{i+1\,i}} \end{array} \right|$$

   weighted by the inverse of their covariances $(\Lambda_i^k)^{-1}$,
2. but also the errors between predicted and estimated depths, i.e. :

$$\xi_i^k(\mu_i, \pi_{i+1}^k, \pi_i^k) = \frac{1}{Z_{i+1}^k} - \left[\frac{1}{Z_i^k} \frac{Z_{i+1\,i}}{Q^{20}_{i+1\,i}\,u_i + Q^{21}_{i+1\,i}\,v_i + Q^{22}_{i+1\,i} + \frac{1}{Z_i^k}\,s^2_{i+1\,i}} \right]$$

   in which we have introduced the zoom factor $Z_{i+1\,i}$, this last quantity being weighted by its information $V_{\xi_i^k}^{-1}$ computed as follows[13] :

$$\begin{aligned} V_{\xi_i^k}^{-1} &= \frac{\partial \pi_i^k}{\partial \xi_i^k}^T \cdot V_{\pi_i^k}^{-1} \cdot \frac{\partial \pi_i^k}{\partial \xi_i^k} \\ V_{\pi_i^k}^{-1} &= \frac{\partial \epsilon_i^k}{\partial \pi_i^k}^T \cdot (\Lambda_i^k)^{-1} \cdot \frac{\partial \epsilon_i^k}{\partial \pi_i^k} \\ \frac{\partial \pi_i^k}{\partial \xi_i^k} &= \frac{Q^{20}_{i+1\,i}\,u_i + Q^{21}_{i+1\,i}\,v_i + Q^{22}_{i+1\,i} + \frac{1}{Z_i^k}\,s^2_{i+1\,i}}{Z_{i+1\,i}} \\ \frac{\partial \epsilon_i^k}{\partial \pi_i^k} &\simeq \frac{1}{Q^{20}_{i+1\,i}\,u_i + Q^{21}_{i+1\,i}\,v_i + Q^{22}_{i+1\,i} + \frac{1}{Z_i^k}\,s^2_{i+1\,i}} \underbrace{\begin{pmatrix} s^0_{i+1\,i} - u_{i+1}\,s^2_{i+1\,i} \\ s^1_{i+1\,i} - v_{i+1}\,s^2_{i+1\,i} \end{pmatrix}}_{\nu_i^k} \end{aligned}$$

   so that, finally :

$$V_{\xi_i^k}^{-1} = \frac{1}{Z^2_{i+1\,i}}\,\nu_i^{k\,T} \cdot (\Lambda_i^k)^{-1} \cdot \nu_i^k$$

Combining these measures we obtain the following criterion :

$$J(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{k=1..P, i=0..N}) = \sum_{i=0}^{N-1} \sum_{k=1}^{P} \epsilon_i^{k\,T} \cdot (\Lambda_i^k)^{-1} \cdot \epsilon_i^k + \sum_{i=0}^{N-1} \sum_{k=1}^{P} \xi_i^k\,V_{\xi_i^k}^{-1}\,\xi_i^k$$

The problem of computing motion and structure in a monocular image sequence when no calibration is given has been reduced to the present optimization problem.

---

[13] In an equation of the form $\mathbf{y} = \mathbf{f}(\mathbf{x})$ we can compute the inverse of the covariance of $\mathbf{x}$ from the inverse of the covariance of $\mathbf{y}$ from the well-known formula obtained applying the Implicit Function Theorem :

$$\Lambda_{\mathbf{x}}^{-1} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}^T \cdot \Lambda_{\mathbf{y}}^{-1} \cdot \frac{\partial \mathbf{f}}{\partial \mathbf{x}}$$

Implementation of the minimization process. In order to proceed to the minimization of this rather huge criterion, we decompose the minimization into two subproblems :

- *Motion from structure* : We minimize with respect to the motion parameters $\{\mu_i\}_{i=0..N-1}$, the structure parameters $\{\pi_i^k\}_{k=1..P,i=0..N}$ being constant. In that case the previous criterion is the sum of N independent terms :

$$\begin{aligned} J(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{k=1..P,i=0..N}) &= \sum_{i=0}^{N-1} J_{\mu_i}(\mu_i, \{\pi_i^k\}_{k=1..P,i=0..N}) \\ J_{\mu_i}(\mu_i, \{\pi_i^k\}_{k=1..P,i=0..N}) &= \sum_{k=1}^{P} \epsilon_i^{kT} \cdot (\Lambda_i^k)^{-1} \cdot \epsilon_i^k + \xi_i^k \, V_{\xi_i^k}^{-1} \, \xi_i^k \end{aligned}$$

  and - considering $\{\pi_i^k\}_{k=1..P,i=0..N}$ is constant, the problem reduces to the independent minimization of the N sub-criteria J_{μ_i}. This corresponds to a minimization of the sum of $3\,P$ terms[14] with respect the parameters of μ_i, either 7 (for $i = 0$) or 11 (for $i > 0$). Such a minimization is quite easy to perform.
- *Structure from motion* : We minimize with respect to the structure parameters $\{\pi_i^k\}_{k=1..P,i=0..N}$, the motion parameters $\{\mu_i\}_{i=0..N-1}$ being constant. In that case the previous criterion is the sum of P independent terms :

$$\begin{aligned} J(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{k=1..P,i=0..N}) &= \sum_{k=1}^{P} J_{\pi_k}(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{i=0..N}) \\ J_{\pi_k}(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{i=0..N}) &= \sum_{i=0}^{N-1} \epsilon_i^{kT} \cdot (\Lambda_i^k)^{-1} \cdot \epsilon_i^k + \xi_i^k \, V_{\xi_i^k}^{-1} \, \xi_i^k \end{aligned}$$

  and - considering $\{\mu_i\}_{i=0..N-1}$ is constant, the problem reduces to the independent minimization of the P sub-criteria J_{π_k}. This corresponds to a minimization of the sum of $3\,(N-1)$ terms with respect to the $N+1$ parameters $\{\pi_i^k\}_{i=0..N}$. Such a minimization is again very easy to perform.

Obviously, this decomposition is sub-optimal, but dramatically simplifies the amount of calculations. In order to limit this drawback and to accelerate the convergence of the estimation process, we used a very common strategy (see for instance [30]) in which we decompose the estimation process into two *phases*, a (1) boot-strapping phase and several (2) steady-state phases :

- During **the boot-strapping phase**, we attempt to obtain a initial estimate of a subset of the parameters in order to start the minimization as close as possible to the final estimate. We must obtain an estimation of motion, without any knowledge on the structure, which corresponds to *the estimation of the fundamental matrix* $F_{10}(\mu_0)$. The corresponding criterion can be derived exactly as the previous one, and we have, very briefly :

$$\begin{aligned} J'(\mu_0) &= \sum_{k=1}^{P} \xi_0^k V_{\xi_0^k}^{-1} \xi_0^k + \sum_{k=1}^{P} \xi_1^k V_{\xi_1^k}^{-1} \xi_1^k \\ \xi_0^k &= \frac{m_1^T \cdot F_{10}(\mu_0) \cdot m_0}{\sqrt{(F_{10}(\mu_0) \cdot m_0)^T \cdot U \cdot (F_{10}(\mu_0) \cdot m_0)}} \\ V_{\xi_0^k}^{-1} &= \frac{1}{\frac{\partial \xi_0^k}{\partial m_1} \cdot (\Lambda_i^k) \cdot \frac{\partial \xi_0^{k\,T}}{\partial m_1}} = \frac{(F_{10}(\mu_0) \cdot m_0)^T \cdot U \cdot (F_{10}(\mu_0) \cdot m_0)}{(F_{10}(\mu_0) \cdot m_0)^T \cdot (\Lambda_i^k) \cdot (F_{10}(\mu_0) \cdot m_0)} \\ \xi_1^k &= \frac{m_1^T \cdot F_{10}(\mu_0) \cdot m_0}{\sqrt{(F_{10}(\mu_0)^T \cdot m_1)^T \cdot U \cdot (F_{10}(\mu_0)^T \cdot m_1)}} \\ V_{\xi_1^k}^{-1} &= \frac{1}{\frac{\partial \xi_1^k}{\partial m_0} \cdot (\Lambda_i^k) \cdot \frac{\partial \xi_1^{k\,T}}{\partial m_0}} = \frac{(F_{10}(\mu_0)^T \cdot m_1)^T \cdot U \cdot (F_{10}(\mu_0)^T \cdot m_1)}{(F_{10}(\mu_0)^T \cdot m_1)^T \cdot (\Lambda_i^k) \cdot (F_{10}(\mu_0)^T \cdot m_1)} \\ \text{with } U &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \end{aligned}$$

[14] Each correspondence provides 2 equations (eventually 1 vanishes) for the disparity and 1 for the depth prediction.

following the method developed and discussed by previous authors [17, 18]. These quantities can be interpreted as follows : (a) We have chosen as measurement errors ξ_0^k and ξ_1^k which are the Euclidean distances from a point to its expected epipolar line, (b) in a symmetric way, since we have average the previous quantity in both views, (c) and weighted these quantities by the uncertainty of each correspondence. Please refer to [17] for a discussion.
In any case, please note that this estimation cannot be done using line correspondences (we use 2 views) but point correspondences only. This appears in the fact that we use the covariance matrix (Λ_i^k) and not its inverse which is undefined for normal correspondences. This is a limitation of the present implementation that we do not use line correspondences for the boot-strapping phase, although a straightforward implementation of the previous equations and a mechanism using three views is indeed possible.
The estimation of the fundamental matrix $F_{10}(\mu_0)$ provides an initial estimate of the motion parameters between view 0 and 1. From this initial estimate we can obtain a first estimate of the structure parameters from the "structure from motion" paradigm, minimizing the criteria $J_{\pi_k}(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{i=0..N})$ with respect to $\{\pi_i^k\}_{i=0..N}$ for all k.

- During **the steady-state phases**, we re-estimate alternatively "motion from structure" and "structure from motion", first considering frame 0 only, then integrating frame 0, 1, etc...
When this mechanism has been executed up to view N, the estimation is finished.

This finally leads to a non-trivial architecture, illustrated in figure 5. We must report that we have tried and experimented several alternatives and variations of the proposed paradigm in order to obtain an efficient algorithm, for which we are going to report some experimental results in the next paragraph. This version of the implementation seems to be a good compromise between several constraints, not all being easy to formalize.

We conjecture that several other possibilities must also been experimented in the future, but this is - for sure - out of the scope of this already very long paper. Let us simply describe an alternative.

Fast implementation of the algorithm. In the proposed implementation, each minimization is performed using a comprehensive algorithm for finding the minimum of a sum of squares [10]. No derivatives are required, which is an advantage (less computations, adaptive estimation of either the Gauss-Newton directions or the Newton directions depending on the numerical stability of the algorithm, etc...) with respect to methods requiring the analytic gradient. The method is designed to ensure that steady progress is made whatever the starting point, and to have a rapid ultimate convergence, i.e. super-linear.

However this mechanism is not adapted to a real-time implementation for instance, but we can modified a bit our criterion to obtain a linear algorithm. Let us explain how.

If we consider the quantity :

$$\begin{aligned}
\varepsilon_i^k(\mu_i, \pi_i^k) &= \left(Q_{i+1\,i}^{20}\, u_i + Q_{i+1\,i}^{21}\, v_i + Q_{i+1\,i}^{22} + \tfrac{1}{Z_i^k}\, \mathbf{s}_{i+1\,i}^2\right) \epsilon_i^k(\mu_i, \pi_i^k) \\
&= \left| \begin{array}{l} \left(Q_{i+1\,i}^{20}\, u_i + Q_{i+1\,i}^{21}\, v_i + Q_{i+1\,i}^{22} + \tfrac{1}{Z_i^k}\, \mathbf{s}_{i+1\,i}^2\right) u_{i+1} - Q_{i+1\,i}^{00}\, u_i + Q_{i+1\,i}^{01}\, v_i + Q_{i+1\,i}^{02} + \tfrac{1}{Z_i^k}\, \mathbf{s}_{i+1\,i}^0 \\ \left(Q_{i+1\,i}^{20}\, u_i + Q_{i+1\,i}^{21}\, v_i + Q_{i+1\,i}^{22} + \tfrac{1}{Z_i^k}\, \mathbf{s}_{i+1\,i}^2\right) v_{i+1} - Q_{i+1\,i}^{10}\, u_i + Q_{i+1\,i}^{11}\, v_i + Q_{i+1\,i}^{12} + \tfrac{1}{Z_i^k}\, \mathbf{s}_{i+1\,i}^1 \end{array} \right.
\end{aligned}$$

we can observe that this error is linear with respect to Q, $\mathbf{s}$ and the proximity $\pi = 1/Z$. Similarly, if we consider the quantity :

$$\bar{\nu}_i^k = \left(\bar{Q}_{i+1\,i}^{20}\, u_i + \bar{Q}_{i+1\,i}^{21}\, v_i + \bar{Q}_{i+1\,i}^{22} + \frac{\bar{1}}{Z_i^k}\, \bar{\mathbf{s}}_{i+1\,i}^2\right)$$

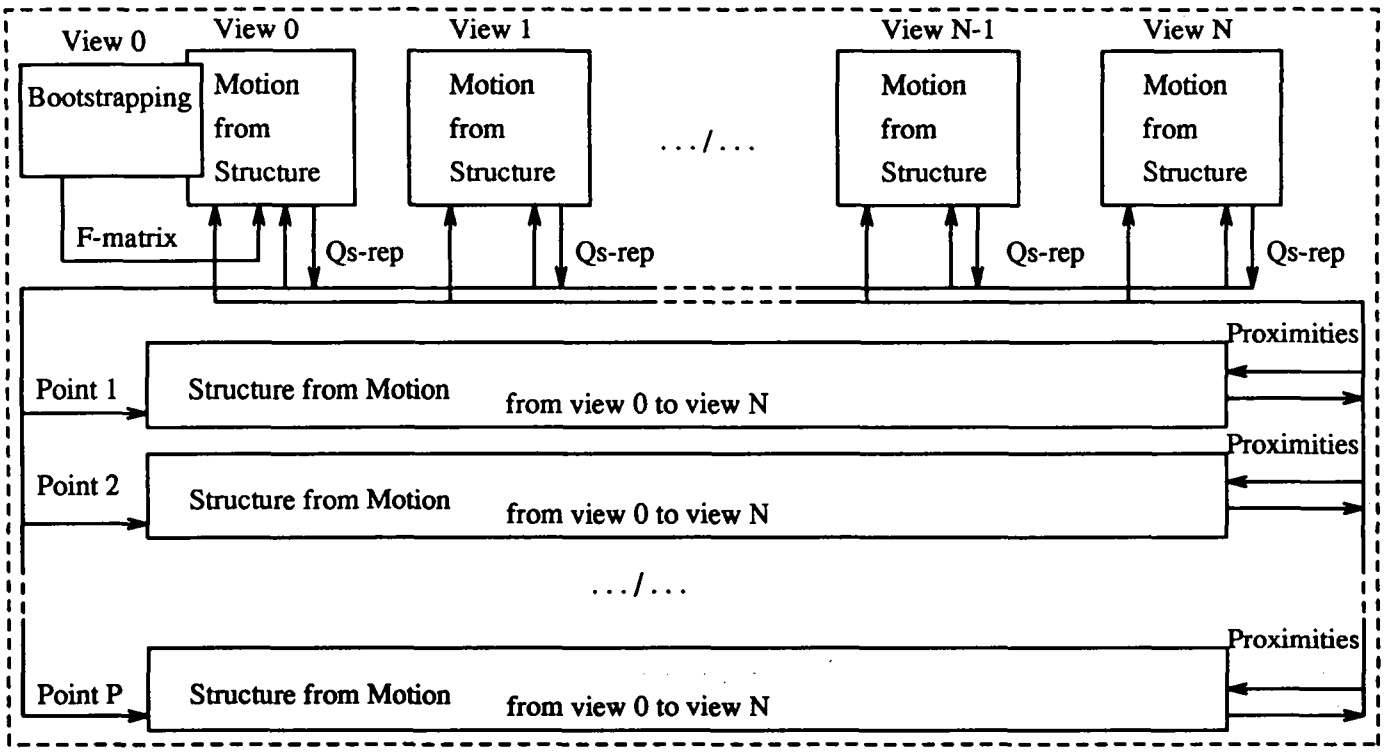

Figure 5: Architecture of the minimization process

estimated *a-priori*, i.e. the value of $\bar{\nu}_i^k$ is not determined by the algorithm, but input and constant, using a-priori information on motion and structure, the quantity :

$$\zeta(\pi_{i+1}^k, \pi_i^k) = \left(Q_{i+1\,i}^{20}\, u_i + Q_{i+1\,i}^{21}\, v_i + Q_{i+1\,i}^{22} + \frac{1}{Z_i^k}\, s_{i+1\,i}^2 \right) \xi_i^k(\mu_i, \pi_{i+1}^k, \pi_i^k) = \bar{\nu}_i^k \frac{1}{Z_{i+1}^k} - \left[\frac{1}{Z_i^k}\, Z_{i+1\,i} \right]$$

is no more dependent on the motion, but only on the structure and is linear with respect to proximity.

As a consequence the modified criterion :

$$\begin{aligned} J'(\{\mu_i\}_{i=0..N-1}, \{\pi_i^k\}_{k=1..P, i=0..N}) &= \sum_{i=0}^{N-1} \sum_{k=1}^{P} \varepsilon_i^k T \cdot \frac{(\Lambda_i^k)^{-1}}{(\bar{\nu}_i^k)^2} \cdot \varepsilon_i^k + \sum_{i=0}^{N-1} \sum_{k=1}^{P} \zeta_i^k \frac{V_{\xi_i^k}^{-1}}{(\bar{\nu}_i^k)^2} \zeta_i^k \\ &= \sum_{i=0}^{N-1} \underbrace{\sum_{k=1}^{P} \varepsilon_i^k T \cdot \frac{(\Lambda_i^k)^{-1}}{(\bar{\nu}_i^k)^2} \cdot \varepsilon_i^k}_{J'_{\mu_i}(\mu_i, \{\pi_i^k\}_{k=1..P, i=0..N})} + \sum_{i=0}^{N-1} \sum_{k=1}^{P} \zeta_i^k \frac{V_{\xi_i^k}^{-1}}{(\bar{\nu}_i^k)^2} \zeta_i^k \\ &= \sum_{k=1}^{P} \underbrace{\sum_{i=0}^{N-1} \varepsilon_i^k T \cdot \frac{(\Lambda_i^k)^{-1}}{(\bar{\nu}_i^k)^2} \cdot \varepsilon_i^k + \zeta_i^k \frac{V_{\xi_i^k}^{-1}}{(\bar{\nu}_i^k)^2} \zeta_i^k}_{J'_{\pi_k}(\{\pi_i^k\}_{i=0..N})} \end{aligned}$$

is, on one hand, quadratic with respect to the components of the Q matrix and s vector and, on the other hand, quadratic with respect to proximity. Therefore, the minimization of $J'_{\mu_i}(\mu_i, \{\pi_i^k\}_{k=1..P, i=0..N})$ and $J'_{\pi_k}(\{\pi_i^k\}_{i=0..N})$ leads to linear normal equations.

With this strategy, each step of the minimization corresponds to solving a linear system of equations, except for μ_0 because of the trigonometric functions. Nevertheless, in this last case,

the equations are linear with respect to $(a_0, b_0, d_0, e_0, f_0)$. As a consequence we can avoid using non-linear minimization except for one module and use only fast algorithms.

In reality, we will verify, that considering the experimental results, the quantity $\bar{\nu}_i^k$ is not far from 1 so that the proposed approximation is valid. It is so true, that even the original criterion yields "almost linear" normal equations. Therefore, the non-linear minimization algorithm converges in one or two steps.

We are not sure, depending on the application, that using linear equations is always a good choice because, although each step is rather quick to execute, the local minimization is less efficient and the global convergence might be affected, if we are far from the minimum. This problem is thus data dependent and will not be further discussed here.

4.2 Experimental results

Using a method to control the obtained values. We have chosen a monocular sequence of 9 frames containing 128 point correspondences. We had to choose an image sequence for which the results of the computation of the motion parameters (rigid motion + calibration) and structure parameters are computable using another well established method. Therefore we have chosen a sequence of views containing a calibration grid, for which all these parameters can be easily computed as often used in the past [22, 18].

A typical image sequence is shown in figure 6, for three consecutive images. The order of magnitude of the disparity between two frames is 2 to 4 pixels, corresponding to rotations of the object of about 5 *deg*. This corresponds to an acquisition at video rate.

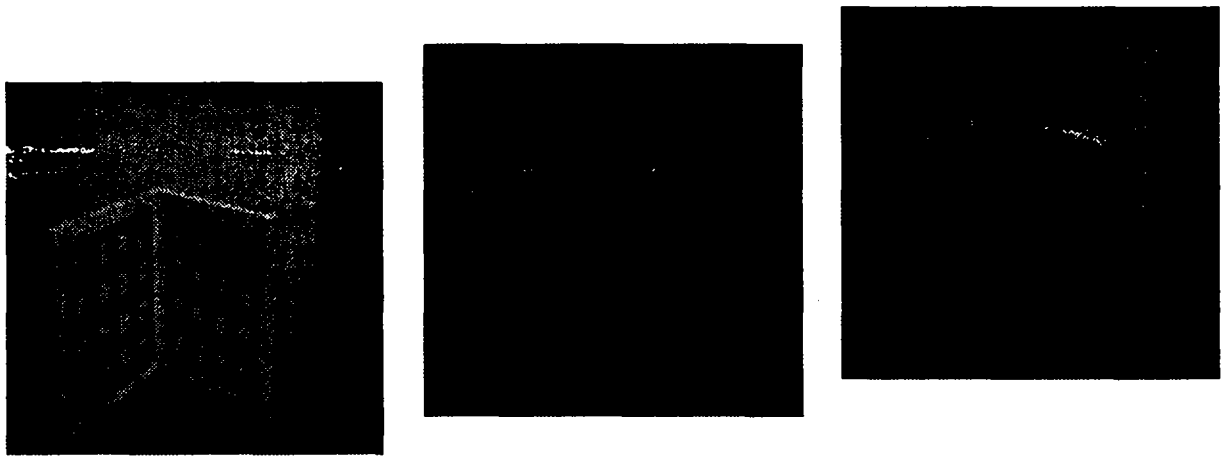

Figure 6: Three images of the sequence used for the experimentation

For each frame, we have computed the usual calibration parameters, i.e the 3×3 matrix P_i and 3×1 vector $\mathbf{p}_i$ which defines the projection of a 3D point M, given in an absolute frame of reference onto a retinal point $m_i = (u_i, v_i, 1)^T$ using the relation [4] :

$$Z_i\, m_i = P_i \cdot M + \mathbf{p}_i$$

If we write R_i and $\mathbf{t}_i$ for the rotation matrix and for the translation vector from the absolute frame of reference to the retinal frame of reference, we immediately verify that :

$$P_i = A_i \cdot R_i \quad ; \quad \mathbf{p}_i = A_i \cdot \mathbf{t}_i$$

and the P-matrix must verify the following constraint $(P_i \cdot P_i^T)^{22} = 1$, since $P_i \cdot P_i^T = A_i \cdot A_i^T = K_i$ while $K_{ii}^{22} = 1$ from eq. (27).

Now if we eliminate M in these equations taken in frame i and j we obtain :

$$Q_{ij} = P_i \cdot P_j^{-1} \quad ; \quad \mathbf{s}_{ij} = \mathbf{p}_i - Q_{ij} \cdot \mathbf{p}_j$$

In fact P_i can be considered as the collineation between the plane at infinity and the retinal plane corresponding to frame i, as easily verified considering a point $M = ||M|| \frac{\mathbf{u}}{||\mathbf{u}||}$ for huge $||M||$. The evolution of these parameters is also governed by our Qs-representation, since :

$$\begin{aligned} P_i &= Q_{ij} \cdot P_j \\ \mathbf{p}_i &= Q_{ij} \cdot \mathbf{p}_j + \mathbf{s}_{ij} \end{aligned}$$

as obtained from the previous relation.

As a consequence, we have an estimate of all the motion and structure parameters using a well established method and can compare these results with the outputs of our module.

Experimental measure of the external parameters. As discussed in the previous section, we can estimate all parameters in this situation, and in particular estimate the external parameters which are inputs of our algorithms. We have obtained :

$$\begin{aligned} ||\mathbf{s}_{10}|| &= 17406.948594 \\ \mathbf{r}_{10} &= (\ -3.3e\text{-}06\ -5.4e\text{-}05\ -0.001\ ) \\ \{Z_{i+1\,i}\}_{i=0..7} &= \{\ 0.999\ 0.989\ 1.035\ 0.992\ 0.994\ 0.997\ 0.992\ 0.970\ 0.986\ \} \end{aligned}$$

which yields the following comments : (1) as expected, the zoom factors are very close to 1 (there were no zoom during this experimentation) and (2) the direction of the r-vector is very close to $(0,0,1)^T$ also, since the plane at infinity is - in standard conditions - almost a fronto parallel plane. The magnitudes of $\mathbf{s}_{10}$ and $\mathbf{r}_{10}$ are of-course very far to one, but that was also expected.

These magnitudes are well explained if we observe that a typical value of the Qs-representation is :

```
      (  0.959767  0.009758 -27.106785)        (  23802.317544)
Q_{08}=( -0.005585  0.945462  19.148278) s_{08}=( -20696.781141)
      (  0.000068  0.000002   0.981526)        (    -58.490636)
```

and it is obvious that the normalization based on the constraint $Q_{i+1\,i}^{22} = 1$ does not modify the order of magnitude of the obtained quantities.

However it appears that the obtained quantities have not the same order of magnitude and this can lead to some numerical instabilities in the algorithms. We have avoided such a problem by taking this order of magnitude into account in the numerical routines so that quantities expected to be very small or very high have been roughly normalized.

Let us now verify if our assumption about the fact that, considering a fast implementation, we can consider $\bar{\nu}_i^k = \left(Q_{i+1\,i}^{20}\, u_i + Q_{i+1\,i}^{21}\, v_i + Q_{i+1\,i}^{22} + \frac{1}{Z_i^k}\, \mathbf{s}_{i+1\,i}^2\right)$ as almost constant is valid. Here are the values obtained along this typical image sequence :

$$\begin{aligned} \bar{\nu}_0^k &= +0.000021u^k - 0.000021v^k + 0.999955 - 5.6e+01\pi^k \\ \bar{\nu}_1^k &= +0.000045u^k - 0.000003v^k + 0.989326 - 9.3e+00\pi^k \\ \bar{\nu}_2^k &= -0.000189u^k + 0.000017v^k + 1.035403 + 5.0e+01\pi^k \\ \bar{\nu}_3^k &= +0.000022u^k + 0.000008v^k + 0.992263 - 4.2e+00\pi^k \\ \bar{\nu}_4^k &= +0.000026u^k - 0.000004v^k + 0.994453 - 1.5e+00\pi^k \\ \bar{\nu}_5^k &= +0.000017u^k - 0.000007v^k + 0.997433 - 9.0e+00\pi^k \\ \bar{\nu}_6^k &= +0.000033u^k - 0.000003v^k + 0.992366 - 1.8e+01\pi^k \\ \bar{\nu}_7^k &= +0.000092u^k + 0.000020v^k + 0.970632 - 1.2e+00\pi^k \end{aligned}$$

for values of π^k of about 10^{-3}, while $(u^k, v^k) \in [0..512[\times[0..512[$. It is clear that this quantity, although not constant, has a value very close to one so that the modified criterion proposed in as a fast implementation of our original criterion is realistic.

We thus have verified that the different assumptions made in this paper on the order of magnitude or the approximate values corresponds to what is effectively measured in a real monocular sequence.

Performances of the algorithm. Let us also very briefly report how *fast* (or slow !) was our implementation.

The "motion from structure" modules converges in about 16 iterations for μ_0, and only 2 to 3 iterations for the other motion parameters $\mu_n, n > 0$ and the convergence is obtain, from scratch, in about $1.5sec$ CPU-time on a Sun-II for 9 views and 128 points.

The "structure from motion" modules converges in about 5 to 10 iterations and the convergence is obtain, from scratch, in about $4.5sec$ CPU-time on a Sun-II for 9 views and 128 points.

When activated together these two sets of modules take about 3 to $5sec$ CPU-time on a Sun-II for 9 views and 128 points (quicker because we never compute from scratch except the first module) for each iteration and the global convergence is obtained after 5 to 10 calls to each modules. In fact if we iterate a huge amount of time, we still gain something but this is peanuts. If we start close to the expected value (as in a steady-state mode) the convergence is obtain after 2 to 3 calls.

As a consequence, it is clear that the fact we have used an almost quadratic criterion for all parameters except the μ_0 yields to a very fast convergence, even-if to maximize the robustness of the estimate we have not used the fast implementation.

We have not yet optimized the code thus CPU-times are still quite long, and must not be taken as definitive values. Previous experience in the field let us predict that a careful coding can decrease these times by a factor of 2 to 10.

Precision of the algorithm, simulation. We have **simulated a projection of the points of the calibration** and we have added a white Gaussian noise to the retinal locations of each points, the standard deviation of this noise being $\sigma_{\mathcal{N}}$. The precision of the estimate is measured considering the distance between the 11 components of the expected and estimated Qs-representations in the parameter space.

We have obtained the following experimental results :

Noise Level	Motion from Structure	Structure from Motion
$\sigma_{\mathcal{N}} = 0$	$1.3\ 10^{-10}$	$6.23\ 10^{-15}$
$\sigma_{\mathcal{N}} = 0.1$	$1.8\ 10^{-2}$	$5.08\ 10^{-5}$
$\sigma_{\mathcal{N}} = 0.2$	$3.6\ 10^{-2}$	$9.80\ 10^{-5}$
$\sigma_{\mathcal{N}} = 0.5$	$8.2\ 10^{-2}$	$2.12\ 10^{-4}$
$\sigma_{\mathcal{N}} = 1.0$	$1.8\ 10^{-1}$	$5.06\ 10^{-4}$
$\sigma_{\mathcal{N}} = 2.0$	$3.5\ 10^{-1}$	$9.8\ 10^{-4}$
$\sigma_{\mathcal{N}} = 5.0$	$8.4\ 10^{-1}$	$2.5\ 10^{-3}$

which are quite accurate values since the reconstruction, obtained with a level of noise of $\sigma_{\mathcal{N}} = 1.0$ is given in figure 7.

The experimentation with $\sigma_{\mathcal{N}} = 0$ corresponds to a set of "perfect data". The obtained results should 0 ideally and the residual errors correspond to rounding errors of the machine. This also shows that the experimentation program has normally no bug !

We have also simulated the fact we were using line correspondences instead of point correspondences and obtain very similar results, for instance :

Noise Level	Motion from Structure	Structure from Motion
$\sigma_{\mathcal{N}} = 1.0$	$2.7\ 10^{-1}$	$7.12\ 10^{-4}$
$\sigma_{\mathcal{N}} = 2.0$	$6.5\ 10^{-1}$	$1.13\ 10^{-3}$

These results have been obtained considering 128 lines containing the 128 points and we thus have generated 256 normal correspondences between those points. This validates our mechanism of integration of line correspondences.

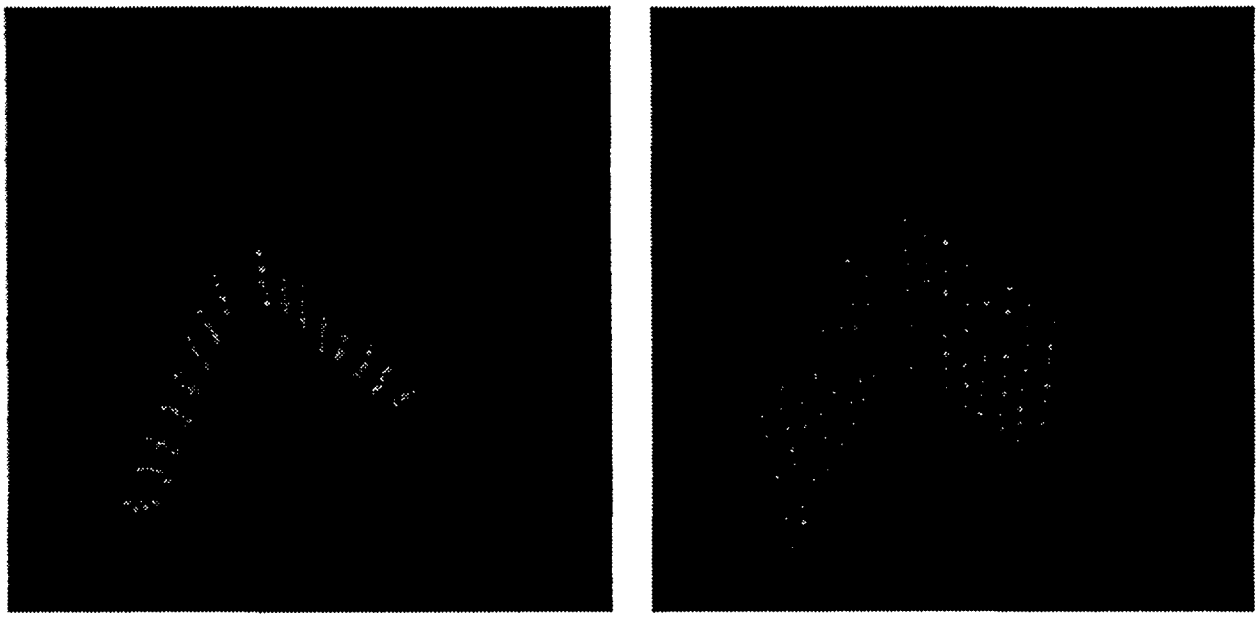

Figure 7: Two perspective views of the 3D reconstruction, simulation with a noise of 1 pixel standard deviation

Precision of the boot-strapping phase We have studied the precision of the boot-strapping phase, which corresponds to the estimation of a fundamental matrix as in [17]. Starting from scratch, the minimization is obtained in about 30 iterations and 1.5*sec* CPU-time in the same conditions as before. Considering the same error measure as above we have obtained :

Noise Level	Motion Only (Boot-Strapping)
$\sigma_{\mathcal{N}} = 0.1$	$9.82\ 10^{-2}$
$\sigma_{\mathcal{N}} = 0.2$	$9.64\ 10^{-2}$
$\sigma_{\mathcal{N}} = 0.5$	$1.04\ 10^{-1}$
$\sigma_{\mathcal{N}} = 1.0$	$1.07\ 10^{-1}$
$\sigma_{\mathcal{N}} = 2.0$	$3.40\ 10^{-1}$
$\sigma_{\mathcal{N}} = 5.0$	$6.35\ 10^{-1}$

which shows that the precision has the same order of magnitude than what has been obtained by the rest of the module; it is more sensitive to small errors because this estimation is definitely not linear, but less sensitive to high errors because a smaller number of parameters (7 instead of 11) is to be estimated.

Radius of convergence of the algorithm, simulation .

As a non-linear algorithm, the proposed module might not converge to the correct solution, if the initial value is too far from what has been expected.

We have observed that we could start from any motion values if the proximity had a relative variation of 20% with respect to their true values, and similarly we could start from any proximity values if the motion values had a relative variation of 10% with respect to their true values.

Moreover the algorithm is still converging when both motion and proximity values relative variations were about 50% of their true values.

This radius of convergence is rather small, and it is then of primary importance to use bootstrapping mechanisms to ensure the convergence of the method.

Precision of the algorithm, real data. We finally have checked our algorithm on the real data and have obtained the following precision :

Real Data	Motion from Structure	Structure from Motion
	$7.39\ 10^{-1}$	$3.16\ 10^{-3}$

which is again quite accurate, although it corresponds to a noise level of about five pixels. Some points have been quite badly located as visible on the reconstruction given in figure 8. When we have observed this fact, we were not disappointed because this was an opportunity to check whether the algorithm could deal with important errors, and it did.

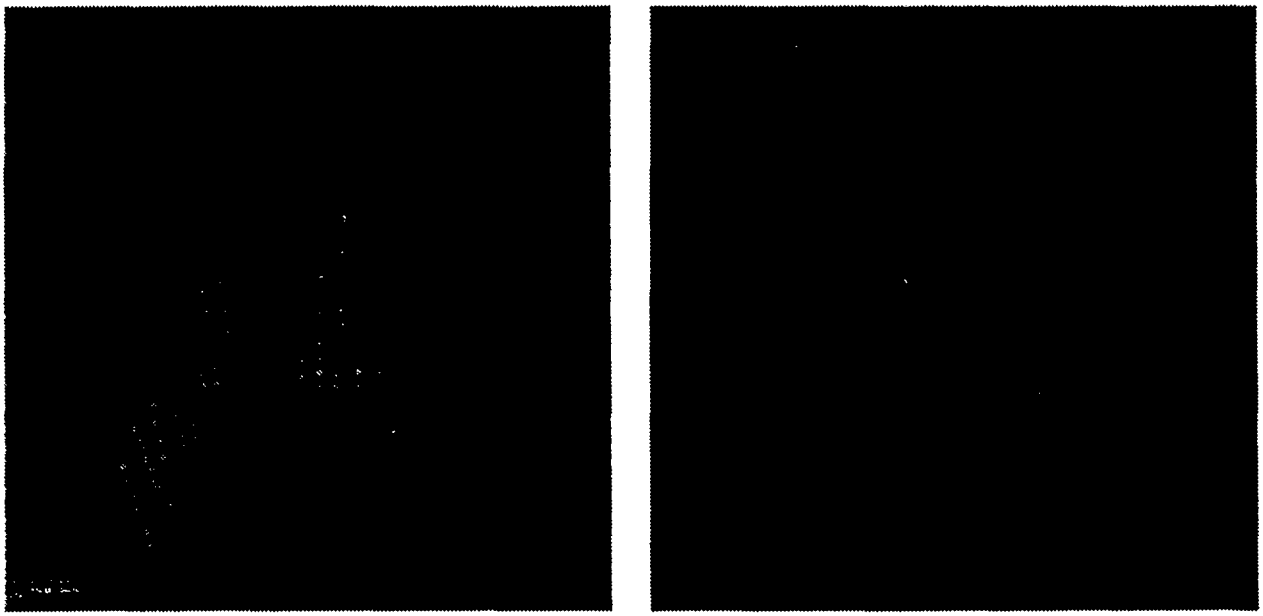

Figure 8: Two perspective views of the 3D reconstruction, real data

5 Conclusion

Considering the problem of computing structure and motion, given a set of point and/or line correspondences, in a monocular image sequence, when the camera is **not** calibrated, we have analyzed the underlying algebraic problem and have proposed a representation of this motion, the Qs-representation which seems to be the simplest generalization of what is well known in the calibrated case.

We have verified that this representation is in deep relation with what has been recently studied in the field : the Fundamental matrix, the 3D reconstruction up to an affine or projective transform and the problem of auto-calibration.

Considering the case of 3 views we have studied in detail the different aspects and possible algorithms of estimation and have developed a geometric interpretation in this case.

At a first glance, the obtained results are quite negative : no possibility of auto-calibration in the general case unless some additional hypotheses are taken into account, apparent increase of the number of constraints between the different parameters involved in this representation when the number of views increases, difficulties to derive complete algorithms for N views and point+line correspondences, etc...

However, using the Qs-representation and accepting some indetermination in this representation it is possible to avoid these difficulties and to estimate what is estimable, that is : $11\ N-4$ parameters and the proximity of the points, corresponding to their affine depths.

If we follow this method, we can obtain a very accurate model of the retinal correspondences and the corresponding numerical values are much more accurate than what has been obtained

in the calibrated case. This might be due to the fact we use more parameters, and the right ones, so that we do not have to trust the very unstable values of a calibration module.

It is clear that this paper does not modify the conclusions obtained by previous researchers who have already analyzed visual data when calibration was not given, but we have tried to extend previous studies in the field in three directions : (1) variable calibration parameters, (2) line and point correspondences, (3) not only 2/3 views but many.

Finally, since the reader is waiting for our conclusion about the dramatic dilemma between algebra and geometry, we state :

Algebra is painful, geometry is beautiful. In a misguided attempt to sell the former, we might have tried to maintain that to derive algorithms algebra is useful, even if awful. Fool ! The goal of Vision is neither to obtain engineering tools, nor good experimental results, but to build Science, and be cool.

Acknowledgements *We are especially thankful to* **Olivier FAUGERAS** *for his help during the elaboration of this work, and for his powerful ideas which are at the origin of this work. The trifocal construction is entirely due to some of his unpublished work. Many concepts have been clarified, thank's to his advises.*

References

[1] Y. Bar Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic-Press, Boston, 1988.

[2] R. Deriche and O. D. Faugeras. Tracking Line Segments. In *Proceedings of the 1st ECCV, Antibes*, pages 259–269. Springer-Verlag, Berlin, 1990.

[3] O. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig ? In *2nd ECCV*, Genoa, 1992.

[4] O. Faugeras. *Three-dimensional Computer Vision: a geometric viewpoint*. MIT Press, Boston, 1993.

[5] O. Faugeras, B. Hotz, H. Mathieu, T. Viéville, Z. Zhang, P. Fua, E. Théron, L. Moll, G. Berry, J. Vuillemin, P. Bertin, and C. Proy. Real time correlation-based stereo: algorithm, implementations and applications. Technical report, INRIA, 1993. in press.

[6] O. Faugeras, Q. T. Luong, and S. Maybank. Camera self-calibration : Theory and experiment. In *2nd ECCV*, Genoa, 1992.

[7] O. Faugeras, N. Navab, and T. Viéville. Ambiguity in motion estimation from line matches. In *Proceedings of ICCV93*, Berlin, 1993.

[8] O. D. Faugeras. On the motion of 3-D curves and its relationship to optical flow. In *Proceedings of the 1st ECCV, Antibes*, pages 107–118. Springer-Verlag, Berlin, 1990.

[9] O. D. Faugeras, F. Lustman, and G. Toscani. Motion and structure from point and line matches. In *Proceedings of the First International Conference on Computer Vision, London*, pages 25–34, June 1987.

[10] P. E. Gill and W. Murray. Algorithms for the solution of non-linear least squares problem. *SIAM Journal on Numerical Analysis*, 15:977–992, 1978.

[11] R. Hartley. Invariants of points seen in multiple images. In preparation for P.A.M.I., 1993.

[12] J. Heel. Temporally integrated surface reconstruction. In *Proceedings of the 3rd ICCV, Osaka*, 1990.

[13] T. Huang and A. Netravali. Linear and polynomial methods in motion estimation. In L. Auslander, T. Kailath, and S. Mitter, editors, *Signal Processing, Part I: Signal Processing Theory*. Springer Verlag, 1990.

[14] J. Lavest, G. Rives, and M. Dhome. 3D reconstruction by zooming. In *I.A.S.*, 1993.

[15] Y. Liu and T. S. Huang. Estimation of rigid body motion using straight line correspondences. In *Proceedings Workshop on Motion : Representation and Analysis*, pages 47–51, Charleston, South California, May 1986.

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

[17] Q. Luong, R. Deriche, O. Faugeras, and T. Papadopoulo. On determining the fundamental matrix: analysis of different methods and experimental results. Technical Report RR-1894, INRIA, Sophia, France, 1993.

[18] T. Luong. *Matrice Fondamentale et Calibration Visuelle sur l'Environnement.* PhD thesis, Université de Paris-Sud, Orsay, 1992. PhD thesis.

[19] S. Maybank and O. Faugeras. A theory of self-calibration of a moving camera. *The International Journal of Computer Vision*, 8, 1992.

[20] A. Mitiche, S. Seida, and J. K. Aggarwal. Interpretation of structure and motion using straight line correspondences. In *Proceedings of the 8th ICPR*, pages 1110–1112, Paris, France, October 1986.

[21] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling. *Numerical recipes, the art of scientific computing.* Cambridge University Press, Cambridge, U.S.A., 1988.

[22] L. Robert. *Stéréovision : de la mise en correspondence de courbes à l'analyse photogrammétrique de la scène.* PhD thesis, Ecole Polytechnique, Palaiseau. France, 1992. PhD thesis.

[23] L. Robert and O. Faugeras. Relative 3d positionning and 3d convex hull computation from a weakly calibrated stereo pair. In H. Nagel, editor, *4th I.C.C.V., Berlin.* IEEE Computer Society Press, Los Alamitos, California, 1993.

[24] M. Stephens, R. Blisset, D. Charnley, E. Sparks, and J. Pike. Outdoor vehicle navigation using passive 3d vision. In *Computer Vision and Pattern Recognition*, pages 556–562. IEEE Computer Society Press, 1989.

[25] N. A. Thacker. On-line calibration of a 4-dof robot head for stereo vision. In *British Machine Vision Association meeting on Active Vision*, London, 1992.

[26] H. Trivedi. Semi-analytic method for estimating stereo camera geometry from matched points. *Image and Vision Computing*, 9, 1991.

[27] R. Y. Tsai. Synopsis of recent progress on camera calibration for 3D machine vision. *Robotics Review*, 1:147–159, 1989.

[28] T. Viéville. Autocalibration of visual sensor parameters on a robotic head. *Image and Vision Computing*, 1993. in press.

[29] T. Viéville, P. Facao, and E. Clergue. Building a depth and kinematic 3D-map form visual and inertial sensors using the vertical cue. In H. Nagel, editor, *4th I.C.C.V., Berlin.* IEEE Computer Society Press, Los Alamitos, California, 1993.

[30] T. Viéville and O. Faugeras. Feed forward recovery of motion and structure from a sequence of 2D-lines matches. In S. Tsuji, A. Kak, and J.-O. Eklundh, editors, *Third International Conference on Computer Vision, Osaka*, pages 517–522. IEEE Computer Society Press, Los Alamitos, California, 1990.

Unité de Recherche INRIA Sophia Antipolis
2004, route des Lucioles - B.P. 93 - 06902 SOPHIA ANTIPOLIS Cedex (France)

Unité de Recherche INRIA Lorraine Technopôle de Nancy-Brabois - Campus Scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 VILLERS LES NANCY Cedex (France)
Unité de Recherche INRIA Rennes IRISA, Campus Universitaire de Beaulieu 35042 RENNES Cedex (France)
Unité de Recherche INRIA Rhône-Alpes 46, avenue Félix Viallet - 38031 GRENOBLE Cedex (France)
Unité de Recherche INRIA Rocquencourt Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 LE CHESNAY Cedex (France)

EDITEUR
INRIA - Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 LE CHESNAY Cedex (France)

ISSN 0249 - 6399