International Journal of Computer Vision, 25(3), 203–229 (1997)


# A Framework for Uncertainty and Validation of 3-D Registration Methods based on Points and Frames

XAVIER PENNEC, JEAN-PHILIPPE THIRION

Xavier.Pennec@sophia.inria.fr

*INRIA, B.P. 93, 2004 route des Lucioles, 06902 SOPHIA ANTIPOLIS Cedex, FRANCE*

*Received October 10, 1995; Revised May 19, 1996*

**Abstract.** In this paper, we propose and analyze several methods to estimate a rigid transformation from a set of 3-D matched points or matched frames, which are important features in geometric algorithms. We also develop tools to predict and verify the accuracy of these estimations. The theoretical contributions are: an intrinsic model of noise for transformations based on composition rather than addition; a unified formalism for the estimation of both the rigid transformation and its covariance matrix for points or frames correspondences, and a statistical validation method to verify the error estimation, which applies even when no "ground truth" is available. We analyze and demonstrate on synthetic data that our scheme is well behaved. The practical contribution of the paper is the validation of our transformation estimation method in the case of 3-D medical images, which shows that an accuracy of the registration far below the size of a voxel can be achieved, and in the case of protein substructure matching, where frame features drastically improve both selectivity and complexity.

## 1. Introduction

Many algorithms in Computer Vision concern *matching* tasks, whose aim is to find the correspondence between two representations of an object. Matching tasks are closely related, but not equivalent, to *registration* processes, which involve the evaluation of a geometric transformation. The main problem of matching methods is generally to reduce the complexity of associating features, such as in (Ayache & Faugeras, 1986) or (Huttenlocher & Ullman, 1987) for Alignment, (Grimson, 1992) for Interpretation Trees, (Lamdan & Wolfson, 1988; Wolfson, 1990) or (Rigoutsos & Hummel, 1993) for Geometric Hashing, (Besl & McKay, 1992) or (Zhang, 1994) for Iterative Closest Point (ICP) methods.

In the following, we do not discuss matching methods per se, but the estimation of the 3-D motion. The traditional approach is to apply a least squares method using, for example, the singular value decomposition in (Arun *et al.*, 1987) and (Umeyama, 1991), or the quaternion representation in (Horn, 1987). Some authors, such as (Zhuang & Huang, 1994), concentrate on the robustness of that estimation.

Uncertainty handling is a central topic in several works, like (Durrant-Whyte, 1988a; Durrant-Whyte, 1988b) or (Ayache, 1991; Zhang & Faugeras, 1992). There are fewer studies, however, dealing with the precision of the estimated motion, which is our main concern here. Early experimental works can be found in (Fang & Huang, 1984), (Snyder, 1989), or (Haralick *et al.*, 1989). A theoretical evaluation of the errors on the rotation estimation is introduced in (Kanatani, 1993) and (Kanatani, 1994). The present paper is the continuation of the work presented in (Pennec & Thirion, 1995).

As previous studies deal principally with point-to-point correspondences, one of our contributions is a method which applies also to frame-to-frame correspondences (a frame is a local coordinate system, i.e. a point with a trihedron). This raises some interesting problems in error modeling: a common assumption is that errors in geometric features can be modeled with an additive error on the parameters of the feature representation. We demonstrate that some important invariance properties are not conserved with this type of noise and we strongly believe that the assumption of additivity should be replaced by the composition with an error motion. This idea leads us to the devel-

opment of a formalism which unifies the handling of the error for the application, composition and estimation of rigid motions, in the case of point-to-point or frame-to-frame correspondences.

Section 2 investigates rigid motions and the handling of uncertainty on geometric features, from the probabilistic point of view. Section 3 describes the statistical point of view, in particular the noise model which should be used. We present a method to estimate a rigid motion and its covariance matrix from a set of matched points or a set of matched frames. In a similar framework, we are able to estimate the fusion of several rigid motions. Those estimates are used to evaluate the noise process on the features used, and sort matched primitives (points or frames) in order to reject outliers, that is, matched primitives which are "obviously" incompatible with the estimated motion.

Section 4 describes our robust motion estimation algorithm based on the theoretical results of section 2. It can be placed after, or better, embedded within your favorite matching algorithm. Essentially, this algorithm is the iteration of three processes: motion estimation, noise process estimation and outliers rejection. It gives, from an initial set of matched primitives with or without a priori knowledge of their covariance, a final rigid motion and its covariance matrix, an estimation of the noise process on features, and the sorted set of matched primitives compatible with that motion.

Another contribution is presented in section 5, where we describe an a posteriori method to validate statistically our motion estimation algorithm. This statistical validation applies even when no "ground truth" is available.

Finally, we present in section 6 experimental results showing the performance of our registration algorithm and its statistical validation. Those results are given for synthetic data, and also for real data, namely 3D medical images, for which the evaluation of the precision of the registration is vital. Another application example is presented with 3D protein substructure matching.

## 2. Probabilistic features and rigid motions

### 2.1. Representation of rigid motion and frames

Let $\mathcal{B} = \{\mathbf{o}, \mathbf{i}, \mathbf{j}, \mathbf{k}\}$ be the canonical right-handed orthonormal coordinate system of Euclidean space $\mathbb{R}^3$ (we have thus $\mathbf{o} = (0,0,0)^\top$ and $[\mathbf{i}, \mathbf{j}, \mathbf{k}] = \mathbf{I_d}$). Let $\mathcal{F} = \{\mathbf{t}, \mathbf{i}', \mathbf{j}', \mathbf{k}'\}$ be another right-handed orthonormal coordinate system (i.e. a frame), which coordinates are expressed in $\mathcal{B}$. The rigid motion from $\mathcal{B}$ to $\mathcal{F}$ can be written in homogeneous coordinates

$$\mathbf{M} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \quad \text{with} \quad \mathbf{R} = [\mathbf{i}', \mathbf{j}', \mathbf{k}']$$

Let $\mathbf{R}$ be a rotation of angle $\theta$ around the unit axis $\mathbf{n}$. Using the rotation vector $\mathbf{r} = \theta\mathbf{n}$ (see (Ayache, 1991) and appendix A.1), we can also represent the rigid motion $\mathbf{M}$ by the six dimensional vector $\mathbf{f}^\top = (\mathbf{r}^\top, \mathbf{t}^\top)$.

We note that $\mathbf{f}$ represents both the rigid motion $\mathbf{M}$ and the frame $\mathcal{F}$, with respect to the canonical coordinate system $\mathcal{B}$. Other representations could have been chosen for the rotation, e.g. quaternions, Euler's angles... but $\mathbf{f}$ would still represent the same rigid motion or feature frame. For convenient notation within the text, we write $\mathbf{f} = (\mathbf{r}, \mathbf{t})$ and call it indifferently a frame or a rigid motion when there is no ambiguity.

### 2.2. Utility of frames in geometric computations

The use of frame features instead of points is motivated by three main reasons. Firstly, frames are natural features emerging from some geometric problems. For instance, extremal points (defined in (Thirion & Gourdon, 1995)) are points on an iso-surface optimizing a differential geometry criterion. Since they lie on a surface, they are provided with the two principal curvature directions $\mathbf{t}_1, \mathbf{t}_2$ and the surface normal $\mathbf{n}$ (see Fig. 1), which forms a frame.

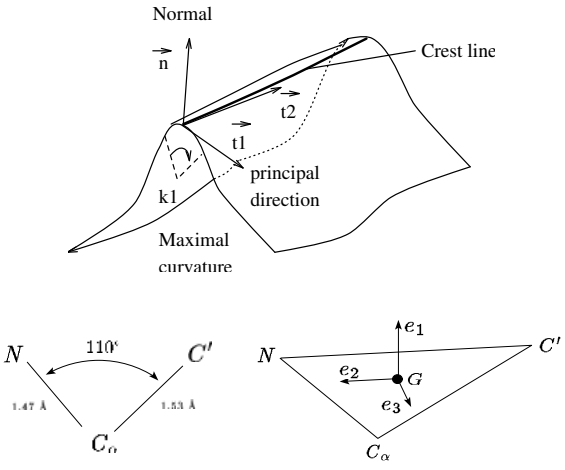


*Fig. 1.* Frame features naturally arise from geometric problems. Top: (extremal) points on iso surfaces. Bottom: geometric description of an amino acid.

In protein substructure matching, an amino acid location is characterized by the position of 3 atoms in a specific geometry (see figure 1). Since the constraints are identical for every amino acid, the shape of the triangle is not discriminant and the only geometric descriptor of an amino acid is its pose, which is again described by a frame. For computer vision applications, we could also think to model each vertex of a polyhedral scene by a frame.

The second reason is that more information is available, thus higher selectivity and/or accuracy (for complexity reasons, however, the extra information should be statistically relevant in order to be useful). The introduction of normals in (Feldmar *et al.*, 1994) allows, for instance, an easy search for seed matches by the use of bitangents, and an improvement of the accuracy in the pose estimation. In the same way, using frame features instead of points allows one to exploit the maximum information about the features (see section 6), and can lead to a drastic reduction of the problem complexity as in (Pennec & Ayache, 1994). Last but not least, frames and rigid motions are equivalent objects from a mathematical viewpoint, and knowing how to handle uncertainty on rigid motions is also necessary.

### 2.3. Operations on frames and motions

Rigid motions can be composed, inverted and applied to points: the composition (operator $\circ$), inverse ($^{(-1)}$) and apply ($\star$) operations on rotation vectors (see appendix A.1), can be easily extended to motions in the following way.

- Application of $\mathbf{f} = (\mathbf{r}, \mathbf{t})$ to point $\mathbf{x}$:

$$\mathbf{y} = \mathbf{f} \star \mathbf{x} = \mathbf{r} \star \mathbf{x} + \mathbf{t}$$

- Composition of $\mathbf{f}_1 = (\mathbf{r}_1, \mathbf{t}_1)$ by $\mathbf{f}_2 = (\mathbf{r}_2, \mathbf{t}_2)$:

$$\mathbf{f} = \mathbf{f}_2 \circ \mathbf{f}_1 = \left| \begin{array}{l} \mathbf{r}_2 \circ \mathbf{r}_1 \\ \mathbf{r}_2 \star \mathbf{t}_1 + \mathbf{t}_2 \end{array} \right.$$

- Inverse of $\mathbf{f} = (\mathbf{r}, \mathbf{t})$:

$$\mathbf{f}^{(-1)} = \left| \begin{array}{l} \mathbf{r}^{(-1)} \\ \mathbf{r}^{(-1)} \star (-\mathbf{t}) \end{array} \right.$$

Due to the equivalence of frames and motions, the application of a rigid motion $\mathbf{f}$ to a frame $\mathbf{f_m}$ is simply the composition $\mathbf{f_s} = \mathbf{f} \circ \mathbf{f_m}$. We note that in this formula, $\mathbf{f_m}$ and $\mathbf{f_s}$ have the same interpretation (frame or motion), but $\mathbf{f}$ is always a motion.

### 2.4. Probabilistic features

Estimated or measured values of geometric features are corrupted by measurement errors. Let $\mathbf{x}$ be the exact representation of such a feature and $\hat{\mathbf{x}}$ the measured one. From a probabilistic point of view, $\hat{\mathbf{x}}$ is the observation or realization of a $n$-D random vector characterized by its probability density function (pdf) $\rho_{\mathbf{x}}$. For computational reasons, however, a common assumption is to retain only the first and second order centered moments of the pdf, i.e. the expectation vector and covariance matrix of the *representation* considered as a random vector. This approach was mainly developed in robotics (Smith & Cheeseman, 1987; Smith *et al.*, 1987; Durrant-Whyte, 1988a; Durrant-Whyte, 1988b) with some applications in computer vision (Zhang & Faugeras, 1992). We note that no assumption is made about the physical noise process. The expectation and the covariance matrix of an "uncertain feature" are defined by

$$\begin{aligned} \bar{\mathbf{x}} &= E(\hat{\mathbf{x}}) = \int \mathbf{y}.\rho_{\mathbf{x}}(\mathbf{y}).d\mathbf{y} \\ \mathbf{\Sigma_{xx}} &= E\left((\hat{\mathbf{x}} - \bar{\mathbf{x}})(\hat{\mathbf{x}} - \bar{\mathbf{x}})^\top\right) \end{aligned}$$

The probabilistic feature $\hat{\mathbf{x}}$ is then treated as the couple $(\bar{\mathbf{x}}, \mathbf{\Sigma_{xx}})$. A common hypothesis is that the noise is *centered*, i.e. $E(\hat{\mathbf{x}}) = \bar{\mathbf{x}} = \mathbf{x}$. In this case, a probabilistic feature is the couple $(\mathbf{x}, \mathbf{\Sigma_{xx}})$ and a deterministic feature has a null covariance matrix.

It is now interesting to see how to generalize our group operations on probabilistic features. For that purpose, we use the classical first order approximation of Jacobians.

*Propagation through an explicit function* Let $\mathbf{x} = (\bar{\mathbf{x}}, \mathbf{\Sigma_{xx}})$ be an $m$-D random vector and $h$ a $p$-dimensional $C^1$ function acting on $\mathbf{x}$. Then the $p$-D random vector $\mathbf{z} = h(\mathbf{x})$ is determined, up to the first order, by $\mathbf{z} = (\bar{\mathbf{z}}, \mathbf{\Sigma_{zz}})$ where

$$\left\{ \begin{array}{l} \bar{\mathbf{z}} = h(\bar{\mathbf{x}}) \\ \mathbf{\Sigma_{zz}} = \mathbf{J}_h.\mathbf{\Sigma_{xx}}.\mathbf{J}_h^\top \end{array} \right. \tag{1}$$

with

$$\mathbf{J}_h = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_p}{\partial x_1} & \cdots & \frac{\partial h_p}{\partial x_m} \end{bmatrix}$$

The Jacobian matrix $\mathbf{J}_h$ generally depends on the point $\mathbf{x}$ where it is computed, and we denote by $\mathbf{J}_h|_{\mathbf{x}}$ its value whenever the application point needs to be specified. For a function $h(\mathbf{x}, \mathbf{y})$ of two independent parameters, the mean is simply $\bar{\mathbf{z}} = h(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ and the covariance matrix

$$\mathbf{\Sigma_{zz}} = \mathbf{J}_{h_\mathbf{x}} \mathbf{\Sigma_{xx}} \mathbf{J}_{h_\mathbf{x}}^\top + \mathbf{J}_{h_\mathbf{y}} \mathbf{\Sigma_{yy}} \mathbf{J}_{h_\mathbf{y}}^\top$$

with

$$\mathbf{J}_{h_\mathbf{x}} = \frac{\partial h(\mathbf{x}, \mathbf{y})}{\partial \mathbf{x}} \quad \text{and} \quad \mathbf{J}_{h_\mathbf{y}} = \frac{\partial h(\mathbf{x}, \mathbf{y})}{\partial \mathbf{y}}$$

The composition of functions is often used to simplify the calculus. Let $h_1$, $h_2$ and $h$ be three multi-dimensional functions such as $h(\mathbf{x}) = (h_1 \circ h_2)(\mathbf{x}) = h_1(h_2(\mathbf{x}))$. The Jacobian of $h$ is then

$$\mathbf{J}_h\Big|_\mathbf{x} = \mathbf{J}_{h_1}\Big|_{h_2(\mathbf{x})} . \mathbf{J}_{h_2}\Big|_\mathbf{x}$$

*Propagation through an implicit function* Let $\mathbf{x} = (\bar{\mathbf{x}}, \mathbf{\Sigma_{xx}})$ be a $m$-D random vector and $\Phi : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}^p$ a $C^1$ function. We want to investigate the $p$-dimensional random vector $\mathbf{y}$ implicitly defined by $\Phi(\mathbf{x}, \mathbf{y}) = 0$. From the implicit function theorem, $\mathbf{y} = \varphi(\mathbf{x})$ exists around a given point $(\mathbf{x_0}, \mathbf{y_0})$ such as $\Phi(\mathbf{x_0}, \mathbf{y_0}) = 0$ if and only if $\dfrac{\partial \Phi}{\partial \mathbf{y}}$ can be inverted at this point and we have in this case

$$\frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \frac{\partial \varphi}{\partial \mathbf{x}} = -\left(\frac{\partial \Phi}{\partial \mathbf{y}}\right)^{-1} \frac{\partial \Phi}{\partial \mathbf{x}}$$

Using a first order Taylor series expansion, $\bar{\mathbf{y}}$ is implicitly defined by $\Phi(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = 0$ and the covariance matrix is given by

$$\mathbf{\Sigma_{yy}} = \left(\frac{\partial \Phi}{\partial \mathbf{y}}\right)^{-1} \frac{\partial \Phi}{\partial \mathbf{x}} \, \mathbf{\Sigma_{xx}} \, \frac{\partial \Phi}{\partial \mathbf{x}}^\top \left(\frac{\partial \Phi}{\partial \mathbf{y}}\right)^{-\top} \tag{2}$$

*Minimizing a criterion* Let $C$ be a function of class $C^2$ from $\mathbb{R}^m \times \mathbb{R}^p$ to $\mathbb{R}^+$ (i.e. a criterion). We now define the $p$-dimensional random vector $\hat{\mathbf{y}}$ as the argument for which the criterion is minimum for a given $\mathbf{x}$:

$$\hat{\mathbf{y}} = \text{ArgMin}_\mathbf{y} \left(C(\mathbf{x}, \mathbf{y})\right)$$

A necessary condition to obtain a minimum is

$$\Phi(\mathbf{x}, \hat{\mathbf{y}}) = \frac{\partial C}{\partial \mathbf{y}}^\top \Bigg|_{(\mathbf{x}, \hat{\mathbf{y}})} = 0$$

and

$$\mathbf{H} = \frac{\partial^2 C}{\partial \mathbf{y}^2}\Bigg|_{(\mathbf{x}, \hat{\mathbf{y}})} \quad \text{positive definite}$$

We are then back to the case of an implicit function and since the Hessian matrix $\mathbf{H}$ is symmetric, the covariance matrix of $\hat{\mathbf{y}}$ at the minimum is

$$\mathbf{\Sigma}_{\hat{\mathbf{y}}\hat{\mathbf{y}}} = \mathbf{H}^{-1} \widehat{\frac{\partial \Phi}{\partial \mathbf{x}}} \, \mathbf{\Sigma_{xx}} \, \widehat{\frac{\partial \Phi}{\partial \mathbf{x}}}^\top \mathbf{H}^{-1} \tag{3}$$

This propagation scheme will be used in section 2 for the rigid motion minimizing the least squares criteria.

### 2.5. *Propagation of uncertainty for rigid motions*

To use the above framework on frames and rigid motions, we need to determine the Jacobians of the three basic operations: composition, inversion of frames, and application of a motion to a point. Appendix A.1 describes the tedious computations related to rotations, in particular the Jacobian of the composition of rotation vectors.

*Application of a frame* $(\mathbf{f}, \mathbf{\Sigma_{ff}})$ *to a point* $(\mathbf{x}, \mathbf{\Sigma_{xx}})$ Let $\mathbf{J}_\star$ be the Jacobian of $\mathbf{y} = \mathbf{f} \star \mathbf{x}$ with respect to $\mathbf{f} = (\mathbf{r}, \mathbf{t})$ and $\mathbf{R}$ the rotation matrix associated to the rotation vector $\mathbf{r}$.

$$\begin{array}{c} \mathbf{J}_\star = \frac{\partial (\mathbf{f} \star \mathbf{x})}{\partial \mathbf{f}} = \frac{\partial (\mathbf{r} \star \mathbf{x} + \mathbf{t})}{\partial (\mathbf{r}, \mathbf{t})} = \left[ \frac{\partial (\mathbf{r} \star \mathbf{x})}{\partial \mathbf{r}} \; ; \; \mathbf{I}_3 \right] \\ \frac{\partial (\mathbf{f} \star \mathbf{x})}{\partial \mathbf{x}} = \frac{\partial (\mathbf{r} \star \mathbf{x} + \mathbf{t})}{\partial \mathbf{x}} = \mathbf{R} \end{array}$$

The computation of the Jacobian $\frac{\partial (\mathbf{r} \star \mathbf{x})}{\partial \mathbf{r}}$ is detailed in appendix A.1.4. The covariance matrix of $\mathbf{y}$ is then

$$\mathbf{\Sigma_{yy}} = \mathbf{J}_\star \, \mathbf{\Sigma_{ff}} \, \mathbf{J}_\star^\top + \mathbf{R} \, \mathbf{\Sigma_{xx}} \, \mathbf{R}^\top \tag{4}$$

*Composition of frame* $(\mathbf{f}_1, \mathbf{\Sigma}_{11})$ *by* $(\mathbf{f}_2, \mathbf{\Sigma}_{22})$ Let $\mathbf{J}_1$ (resp. $\mathbf{J}_2$) be the Jacobian of $\mathbf{f} = \mathbf{f}_2 \circ \mathbf{f}_1$ with respect to $\mathbf{f}_1 = (\mathbf{r}_1, \mathbf{t}_1)$ (resp. $\mathbf{f}_2 = (\mathbf{r}_2, \mathbf{t}_2)$). Then

$$\mathbf{J}_1 = \frac{\partial (\mathbf{f}_2 \circ \mathbf{f}_1)}{\partial \mathbf{f}_1} = \begin{bmatrix} \frac{\partial (\mathbf{r}_2 \circ \mathbf{r}_1)}{\partial \mathbf{r}_1} & 0 \\ 0 & \mathbf{R}_2 \end{bmatrix}$$

$$\mathbf{J}_2 = \frac{\partial (\mathbf{f}_2 \circ \mathbf{f}_1)}{\partial \mathbf{f}_2} = \begin{bmatrix} \frac{\partial (\mathbf{r}_2 \circ \mathbf{r}_1)}{\partial \mathbf{r}_2} & 0 \\ \frac{\partial (\mathbf{r}_2 \star \mathbf{t}_1)}{\partial \mathbf{r}_2} & \mathbf{I}_3 \end{bmatrix}$$

The computation of the Jacobian of $\mathbf{r_1} \circ \mathbf{r_2}$ with respect to the rotation vectors $\mathbf{r_1}$ and $\mathbf{r_2}$ are detailed in appendix A.1.4. The covariance on $\mathbf{f}$ is simply given by

$$\mathbf{\Sigma_{ff}} = \mathbf{J}_1 \, \mathbf{\Sigma}_{11} \, \mathbf{J}_1^\top + \mathbf{J}_2 \, \mathbf{\Sigma}_{22} \, \mathbf{J}_2^\top \tag{5}$$

*Inversion of frame* $(\mathbf{f}, \mathbf{\Sigma_{ff}})$ Let $\mathbf{J}_I$ be the Jacobian of $\mathbf{f}^{(-1)} = (\mathbf{r}^{(-1)}, \, \mathbf{r}^{(-1)} \star (-\mathbf{t}))$ with respect to $\mathbf{f} = (\mathbf{r}, \mathbf{t})$. As $\mathbf{r}^{(-1)} = -\mathbf{r}$, we have:

$$\mathbf{J}_I = - \begin{bmatrix} \mathbf{I}_3 & 0 \\ \frac{\partial (\mathbf{r}^{(-1)} \star \mathbf{t})}{\partial \mathbf{r}^{(-1)}} & \mathbf{R}^\top \end{bmatrix}$$

using once again the Jacobian of the application of a rotation vector to a point (appendix A.1.4). The covariance of $\mathbf{f}^{(-1)}$ is then

$$\mathbf{\Sigma}_{\mathbf{f}^{(-1)}\mathbf{f}^{(-1)}} = \mathbf{J}_I \, \mathbf{\Sigma_{ff}} \, \mathbf{J}_I^\top \tag{6}$$

We have now completed the computations needs to propagate the covariance matrices through the three basic operations on frames and points.

### 2.6. The noise model

The problem is now slightly different. In the previous sections, we considered that the pdf (or the mean value and the covariance matrix) of the probabilistic geometric features were known. In the real world, this can be a very strong assumption, in particular when we begin with simple measurements without any idea of their noise. In such a case, a natural assumption is to say that all measurements are corrupted by the same noise process. This is the notion of Identically Independently Distributed (IID) measurements (Anderson, 1958). For points, the common assumption is to impose the same covariance matrix on all point measurements, which corresponds effectively to the same additive noise. Regarding frames, we have to be more careful.

Some paradoxes in standard geometric probabilities (see for instance the Bertrand paradox in (Kendall & Moran, 1963)) can be avoided by defining the uniform pdf over a differential manifold (i.e. the set of geometric features of a given type) as the pdf which remains invariant by the action of any transformation of a given (Lie) group. The transformations considered can be, for instance, the rigid motions: if we have a uniform distribution over our manifold, this distribution has to remain uniform when we apply the same rigid motion to all the features (for other examples, see (Pennec & Ayache, 1996)).

For our purpose, we have to impose a similar invariance property on the noise process. Let $\hat{\mathbf{f}}_1$ and $\hat{\mathbf{f}}_2$ be measurements of two frames $\mathbf{f}_1$ and $\mathbf{f}_2$, corrupted by the same noise process. Then $\hat{\mathbf{f}}'_1 = \mathbf{f} \circ \hat{\mathbf{f}}_1$ and $\hat{\mathbf{f}}'_2 = \mathbf{f} \circ \hat{\mathbf{f}}_2$ are measurements of $\mathbf{f'}_1 = \mathbf{f} \circ \mathbf{f}_1$ and $\mathbf{f'}_2 = \mathbf{f} \circ \mathbf{f}_2$ and have to be corrupted by the same noise process (possibly different from the first one). This simply means that two identical (or comparable) distributions remain identical by a change of the reference frame, which seems reasonable. A related assumption is usually required for criterions: if $\mathbf{y}$ minimizes $C(\mathbf{x}, \mathbf{Y})$, it is desirable to have a criterion invariant by a given group of transformation (i.e. $C(\mathbf{f} \star \mathbf{x}, \mathbf{f} \star \mathbf{Y}) = C(\mathbf{x}, \mathbf{Y})$ for any transformation $\mathbf{f}$) in order to obtain the transformed result $\mathbf{f} \star \mathbf{y}$ minimizing the criterion $C(\mathbf{f} \star \mathbf{x}, \mathbf{Y})$ on transformed data.

*Why additive noise is not adapted* Let $\hat{\mathbf{f}}_1$ and $\hat{\mathbf{f}}_2$ be measurements of the frames $\mathbf{f}_1$ and $\mathbf{f}_2$, corrupted (independently) by the same noise process of covariance $\mathbf{\Sigma}$. If the two noises are centered, this means that $\hat{\mathbf{f}}_i = \mathbf{f}_i + \delta\mathbf{f}_i$ with $E(\delta\mathbf{f}_i) = \mathbf{0}$ and

$$E\left(\delta\mathbf{f}_i . \, \delta\mathbf{f}_i^\top\right) = \mathbf{\Sigma}$$

If we change our view-point, i.e. apply a global motion $\mathbf{f}$ to exact and measured values, we obtain $\hat{\mathbf{f}}'_i = \mathbf{f} \circ \hat{\mathbf{f}}_i = \mathbf{f'}_i + \delta\mathbf{f'}_i$ with $E(\delta\mathbf{f'}_i) = \mathbf{0}$. The first order approximation of Jacobians gives us the propagation of covariance matrices:

$$E\left(\delta\mathbf{f'}_i . \, \delta\mathbf{f'}_i^\top\right) = \mathbf{J}_i . \mathbf{\Sigma} . \mathbf{J}_i^\top$$

with

$$\mathbf{J}_i = \frac{\partial (\mathbf{f} \circ \mathbf{f}_i)}{\partial \mathbf{f}_i}$$

In general $\mathbf{J}_i$ depends on $\mathbf{f}$ and $\mathbf{f}_i$, therefore $\mathbf{J}_1 \neq \mathbf{J}_2$: the covariance matrix $\mathbf{\Sigma}'_{11}$ of $\hat{\mathbf{f}}'_1$ is thus different from the covariance matrix $\mathbf{\Sigma}'_{22}$ of $\hat{\mathbf{f}}'_2$. The two measurements are no longer corrupted by the same noise process. As we can see, using an additive noise hypothesis can lead to paradoxes since the result of the computations depends upon the chosen reference frame.

*A simple example with rotation vectors* We consider an example with rotations only, represented by their ro-

tation vectors. Let $\mathbf{r}_1 = (\frac{\pi}{2}, 0, 0)^\top$ and $\mathbf{r}_2 = (0, 0, 0)^\top$ be two rotation vectors with the same covariance:

$$\mathbf{\Sigma}_{11} = \mathbf{\Sigma}_{22} = \mathbf{\Sigma} = \sigma^2 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

If we apply a global clockwise rotation of $\pi/2$ around the $x$ axis, represented by the rotation vector $\mathbf{r} = (-\frac{\pi}{2}, 0, 0)^\top$, we obtain the new rotation vectors $\mathbf{r'}_1 = (0, 0, 0)^\top$ and $\mathbf{r'}_2 = (-\frac{\pi}{2}, 0, 0)^\top$. Computing the Jacobians and propagating covariances, we find that $\mathbf{\Sigma}'_{11} = \frac{8}{\pi^2}\mathbf{\Sigma}$, which is significantly different from $\mathbf{\Sigma}'_{22} = \frac{\pi^2}{8}\mathbf{\Sigma}$ (the ratio is approximately 1.5: see Fig. 2).

It is normal that covariance matrices change according to the motion, but since they change differently at different positions, we cannot define identical distributions as additive noises (with different means) sharing the same covariance matrix: if these matrices are *identical* in one coordinate system, they are not in another one. Since the IID hypothesis is used in most usual statistical methods (and will be used in the following), it is important to define it well.

*A compositive noise model* In order to satisfy the invariance constraint, we propose to model the measurement process by the composition of the exact value with a noise (motion): let $\mathbf{f}$ be the exact representation of a frame and $\hat{\mathbf{f}}$ the measured one, then $\hat{\mathbf{f}} = \mathbf{f} \circ \mathbf{e}$ where $\mathbf{e}$ is a small rigid motion around the identity. In this process, $\mathbf{f}$ is a deterministic frame, whereas $\mathbf{e}$ and $\hat{\mathbf{f}}$ are probabilistic.

With this noise model, the invariance property is automatically verified: if $\mathbf{g}$ is the global motion to be applied, $\hat{\mathbf{f}}' = \mathbf{g} \circ \hat{\mathbf{f}} = (\mathbf{g} \circ \mathbf{f}) \circ \mathbf{e} = \mathbf{f}' \circ \mathbf{e}$. We note that this compositive noise model corresponds to the left composition $\mathbf{f} \circ \mathbf{e}$ by the exact frame. We could not have used the right composition $\mathbf{e} \circ \mathbf{f}$ since in this case, the transformation of $\check{\mathbf{f}} = \mathbf{e} \circ \mathbf{f}$ by $\mathbf{g}$ gives $\check{\mathbf{f}}' = \mathbf{g} \circ \mathbf{e} \circ \mathbf{f} \neq \mathbf{e} \circ \mathbf{f}'$. We can then characterize a noise process on frames by a "random frame" $\mathbf{e}$ measuring the identity (i.e. the canonical frame). A centered noise corresponds to $E(\mathbf{e}) = \mathbf{I_d}$, and is entirely defined by its covariance matrix: $\mathbf{e} = (\mathbf{I_d}, \mathbf{\Sigma})$. From now on, the noise on frames is a compositive noise model, and we use the probabilistic measurement $\mathbf{e}$ of identity to represent it.

We note that this is also an additive noise *in the local frame*, or more specially in the tangent space of the manifold at the expected point.

*Validity of the additive noise* It is interesting to note that, for 2D rotations (with angular representation) or for translations (i.e. points), the composition corresponds to the addition. In these cases, compositive and additive noise models are identical, which have provided a wrong intuition for more complex cases. More generally, the additive noise model corresponds to the compositive one as soon as the action of the transformation group on features (or on itself) is linear *in the considered representation*, in which case the tangent space corresponds to the space itself. There exists in this case a matrix $\mathbf{F}$ for each motion $\mathbf{f}$ such that $\mathbf{f} \star \mathbf{x} = \mathbf{F}.\mathbf{x}$ (or $\mathbf{f} \circ \mathbf{g} = \mathbf{F}.\mathbf{g}$) and the Jacobian $\partial(\mathbf{f} \star \mathbf{x})/\partial\mathbf{x} = \mathbf{F}$ is thus independent of the position $\mathbf{x}$ of the feature.

The relevance of the compositive noise model to real cases will be shown in section 7 with the analysis of the estimated noise on extremal points.

## 3. Uncertainty and frames: the statistical point of view

In the first part, we focus on how to model *a priori* information about uncertainty, and how to propagate it through the usual geometric computations. However,

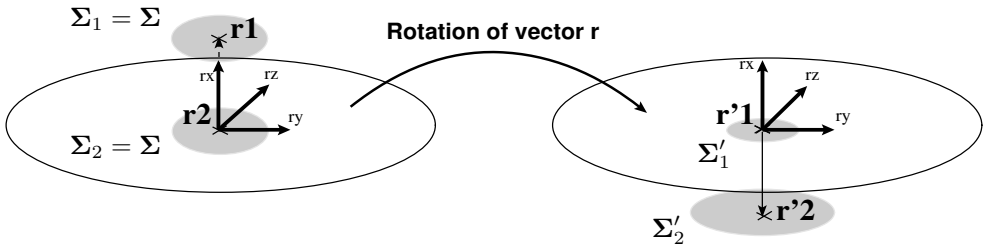


*Fig. 2.* The same covariance matrix on two rotation vectors in one reference frame generally leads to different covariance matrices in another reference frame. The circle is the intersection of the definition domain of the rotation vector with the $(ry, rz)$ plane.

in real cases, we often need to estimate this information *a posteriori* from data. This is the goal of statistics.

The first section is devoted to the estimation of a rigid motion between two sets of matched features, while considering that the noise process on features is known. The following question is thus: how can we estimate this noise process on features? The last section of this part deals with outliers detection.

### *3.1. Estimation of a rigid motion*

In this section, we address two main problems of estimation. The first is the estimation of a rigid motion $(\mathbf{f}, \mathbf{\Sigma_{ff}})$ from two sets of matched geometric features, namely points or frames. This is the registration problem. The second is the fusion of several measurements $(\mathbf{f}_i, \mathbf{\Sigma}_{\mathbf{f}_i\mathbf{f}_i})$ of the same motion in order to get a more accurate estimation. This is a data fusion problem.

Many papers on pose estimation focus on closed-formed solutions in the case of point-to-point correspondences (see the references in the introduction and especially (Kanatani, 1993)), but fewer deal with the uncertainty estimation of the pose found (see however (Csurka *et al.*, 1995)). Moreover, we also want to estimate the rigid motion from frame-to-frame correspondences, and no classical method easily applies to such a problem.

However, a unified framework for estimation with uncertainty has been developed using the Extended Kalman Filter (EKF) in (Ayache, 1991). We recall in section 2 the basic principles of this algorithm and introduce the notations and framework for motion estimation. In the following sections, we apply it to rigid motion estimation from point matches, from frames correspondences, and from a set of measurements of this motion. A special section (2) will be devoted to the computation of the uncertainty of the motion estimated by a standard least-squares on points.

*Kalman Filtering* Assume we have a set of measurements (or data) $\{\boldsymbol{\chi}_i\}$ and we search a state variable $\mathbf{g}$ such that, for each exact data $\boldsymbol{\chi}_i$, we have the vectorial relation

$$\mathbf{z}_i(\boldsymbol{\chi}_i, \mathbf{g}) = \mathbf{0}$$

This is called the measurement equation. In our case, the state $\mathbf{g}$ is the sought rigid motion and the data are couples of matched points or frames, or simple measurements of this rigid motion (section 2). Since we are working with noisy data, we only know the measured values of data $\hat{\boldsymbol{\chi}}_i = \boldsymbol{\chi}_i + \boldsymbol{\omega}_i$ (the observation). The additive noises $\boldsymbol{\omega}_i$ are assumed to be independent, white and centered with a known covariance $\mathbf{\Omega}_i$:

$$E(\boldsymbol{\omega}_i) = \mathbf{0} \qquad E(\boldsymbol{\omega}_i . \boldsymbol{\omega}_i^\top) = \mathbf{\Omega}_i$$

$$\text{and} \qquad E(\boldsymbol{\omega}_i . \boldsymbol{\omega}_j^\top) = \mathbf{0} \qquad \text{for } i \neq j$$

The measurement equations are generally not linear, but assuming we know a good estimate $\hat{\mathbf{g}}$ of the state $\mathbf{g}$, we can linearize them around the estimates and solve the problem with standard linear optimization techniques, namely here Kalman Filtering. This is the basis of the Extended Kalman Filtering technique. Since Kalman Filtering is a recursive filter, we assume that we have at each step $i$ an estimate $\hat{\mathbf{g}}_{i-1}$ of the state vector. We can then linearize the measurement equation around $(\hat{\boldsymbol{\chi}}_i, \hat{\mathbf{g}}_{i-1})$ with a first order Taylor series expansion. Taking the following notations:

$$\hat{\mathbf{z}}_i = \mathbf{z}_i(\hat{\boldsymbol{\chi}}_i, \hat{\mathbf{g}}_{i-1})$$

$$\mathbf{M}_i = \widehat{\frac{\partial \mathbf{z}_i}{\partial \mathbf{g}}} = \left.\frac{\partial \mathbf{z}_i}{\partial \mathbf{g}}\right|_{(\hat{\boldsymbol{\chi}}_i, \hat{\mathbf{g}}_{i-1})}$$

$$\widehat{\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}} = \left.\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}\right|_{(\hat{\boldsymbol{\chi}}_i, \hat{\mathbf{g}}_{i-1})}$$

the Taylor expansion of the measurement equation $\mathbf{z}_i(\boldsymbol{\chi}_i, \mathbf{g}) = \mathbf{0}$ gives

$$\hat{\mathbf{z}}_i + \widehat{\frac{\partial \mathbf{z}_i}{\partial \mathbf{g}}} . (\mathbf{g} - \hat{\mathbf{g}}_{i-1}) + \widehat{\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}} . (\boldsymbol{\chi}_i - \hat{\boldsymbol{\chi}}_i) \simeq 0$$

This equation can be re-written in the linear form $\mathbf{M}_i \,.\, \mathbf{g} = \boldsymbol{\gamma}_i + \boldsymbol{\nu}_i$ where $\boldsymbol{\gamma}_i$ is the linearized measurement $\boldsymbol{\gamma}_i = \mathbf{M}_i . \hat{\mathbf{g}}_{i-1} - \hat{\mathbf{z}}_i$ and $\boldsymbol{\nu}_i$ is a centered noise

$$\boldsymbol{\nu}_i = \widehat{\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}} . (\hat{\boldsymbol{\chi}}_i - \boldsymbol{\chi}_i)$$

of covariance $\mathbf{\Sigma}_{ii}$:

$$\mathbf{\Sigma}_{ii} = \left(\widehat{\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}}\right) \mathbf{\Omega}_i \left(\widehat{\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}}\right)^\top$$

Assuming that an initial estimate $(\hat{\mathbf{g}}_0\ ,\ \hat{\mathbf{G}}_0)$ of the state is given, a reasonable (weighted) least-squares criterion to minimize can be constructed on the basis of Mahalanobis distances:

$$C = \frac{1}{2}(\hat{\mathbf{g}}_0 - \mathbf{g})^\top \hat{\mathbf{G}}_0^{-1}(\hat{\mathbf{g}}_0 - \mathbf{g}) + \frac{1}{2}\sum_i (\boldsymbol{\gamma}_i - \mathbf{M}_i.\mathbf{g})^\top \boldsymbol{\Sigma}_{ii}^{(-1)}(\boldsymbol{\gamma}_i - \mathbf{M}_i.\mathbf{g})$$

The recursive solution for the minimization of this criteria is called the Kalman Filter ((Jazwinsky, 1970; Ayache, 1991)). At each step, the input is an estimate $(\hat{\mathbf{g}}_{i-1}, \hat{\mathbf{G}}_{i-1})$ of the state and the linearized measurement $(\boldsymbol{\gamma}_i, \boldsymbol{\Sigma}_{ii})$, and the output is the updated estimation of the state $(\hat{\mathbf{g}}_i, \hat{\mathbf{G}}_i)$. We just recall here the equations of the filter:

$$\begin{aligned} \mathbf{K}_i &= \hat{\mathbf{G}}_{i-1}.\mathbf{M}_i^\top.(\boldsymbol{\Sigma}_{ii} + \mathbf{M}_i.\hat{\mathbf{G}}_{i-1}.\mathbf{M}_i^\top)^{-1} \\ \hat{\mathbf{g}}_i &= \hat{\mathbf{g}}_{i-1} - \mathbf{K}_i.\hat{\mathbf{z}}_i \\ \hat{\mathbf{G}}_i &= (\ \mathbf{I_d} - \mathbf{K}_i.\mathbf{M}_i).\hat{\mathbf{G}}_{i-1} \end{aligned} \tag{7}$$

For linear transformation of Gaussian variables, the Kalman Filter produces the optimal estimate of the state (in the sense of minimum error variance), which turns out to be also the maximum likelihood estimate. Moreover, it preserves the Gaussian nature of the random variables and thus there is no loss of information in keeping only the mean value and covariance matrix.

If the Gaussian assumption is removed, the Filter remains the best *linear* estimator, but is no longer the best one amongst all non-linear estimators. In the general case of non-linear transformations with any type of noise (which is our case in this article), the Extended Kalman Filter only represent a sub-optimal non linear estimator, but appears to provide accurate estimates in practice. However, some care has to be taken about the initial state and the order of the measurements.

*Estimation from matched points* Assume that we have two sets of matched points $\mathbf{x}_i$ and $\mathbf{y}_i$, one transformed into the other with a rigid motion $\mathbf{f}$ such that $\mathbf{y}_i = \mathbf{f} \star \mathbf{x}_i$. During the measurement process, these points are corrupted by additive noise: we only measure $\hat{\mathbf{x}}_i = \mathbf{x}_i + \delta\mathbf{x}_i$ and $\hat{\mathbf{y}}_i = \mathbf{y}_i + \delta\mathbf{y}_i$ and we want to estimate the motion $\mathbf{f}$ and the covariance $\boldsymbol{\Sigma}_{\mathbf{ff}}$ of this estimation. We assume that the errors $\delta\mathbf{x}_i$ and $\delta\mathbf{y}_i$ are independent with covariances $\boldsymbol{\Sigma}_{\mathbf{x}_i\mathbf{x}_i}$ and $\boldsymbol{\Sigma}_{\mathbf{y}_i\mathbf{y}_i}$.

The error vector is in this case the classical difference in position $\mathbf{z}_i = \hat{\mathbf{y}}_i - \mathbf{f} \star \hat{\mathbf{x}}_i$ (and the measurement equation is $\mathbf{z}_i = \mathbf{0}$). The sought state is a rigid motion $\mathbf{g} = (\mathbf{r}, \mathbf{t})$, and the observation vector is $(\hat{\boldsymbol{\chi}}_i, \boldsymbol{\Omega}_i)$:

$$\hat{\boldsymbol{\chi}}_i = \left| \begin{matrix} \hat{\mathbf{x}}_i \\ \hat{\mathbf{y}}_i \end{matrix} \right. \qquad \boldsymbol{\Omega}_i = \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{x}_i\mathbf{x}_i} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{\mathbf{y}_i\mathbf{y}_i} \end{bmatrix}$$

If $\mathbf{R}$ is the rotation matrix corresponding to the rotation vector $\mathbf{r}$ of the motion $\mathbf{g}$, we have $\partial\mathbf{z}_i/\partial\boldsymbol{\chi}_i = [-\mathbf{R};\ \mathbf{I_d}]$ and hence

$$\boldsymbol{\Sigma}_{\mathbf{z}_i\mathbf{z}_i} = \boldsymbol{\Sigma}_{\mathbf{y}_i\mathbf{y}_i} + \mathbf{R}\boldsymbol{\Sigma}_{\mathbf{x_i}\mathbf{x_i}}\mathbf{R}^\top \tag{8}$$

The EKF process is then the following.

- Initialize the state with Identity or a least squares estimate with a large covariance matrix (100 to 1000 times the least-squares covariance, for example), in order to minimize the influence of the initial state on both the estimation and its uncertainty.
- For each couple of matched points $(\hat{\mathbf{x}}_i, \hat{\mathbf{y}}_i)$:
  - Compute $\hat{\mathbf{h}}_i = \hat{\mathbf{y}}_i - \hat{\mathbf{g}}_{i-1} \star \hat{\mathbf{x}}_i$
  - Compute $\mathbf{M}_i = \frac{\partial \mathbf{h}}{\partial \mathbf{g}} = -\frac{\partial(\mathbf{g}\star\mathbf{x})}{\partial \mathbf{g}}$ estimated at $(\hat{\mathbf{x}}_i, \hat{\mathbf{g}}_{i-1})$
  - Compute $\boldsymbol{\Sigma}_{\mathbf{z}_i\mathbf{z}_i}$ using Eq.(8) with rotation $\mathbf{R}_{i-1}$.
  - Update $(\hat{\mathbf{g}}_{i-1}, \hat{\mathbf{G}}_{i-1})$ to $(\hat{\mathbf{g}}_i, \hat{\mathbf{G}}_i)$ using Eq.(7).

*Least squares* In the isotropic case (more specially when $\boldsymbol{\Sigma}_{\mathbf{x}_i\mathbf{x}_i} = \boldsymbol{\Sigma}_{\mathbf{y}_i\mathbf{y}_i} = \sigma^2\,\mathbf{I_d}$), the covariance on the error vector becomes $\boldsymbol{\Sigma}_{\mathbf{z}_i\mathbf{z}_i} = 2\sigma^2\,\mathbf{I_d}$, and the criterion reduces to a simple least squares

$$C = \frac{1}{2}\sum_i \|\hat{\mathbf{y}}_i - \mathbf{f} \star \hat{\mathbf{x}}_i\|^2 = \frac{1}{2}\sum_i \mathbf{z}_i^\top.\mathbf{z}_i$$

which can be solved exactly by several techniques. One consists in computing the barycenters of the two sets and searching for the rotation using quaternions (see (Horn, 1987)).

The covariance matrix can be determined using equation (3) with some important simplifications: assuming that $\boldsymbol{\chi} = (\mathbf{x}_1, \ldots \mathbf{x_n}, \mathbf{y_1}, \ldots \mathbf{y_n})^\top$ is the vector of all observations and $\mathbf{f}$ the state, the associated implicit function characterizing an optimum is

$$\Phi(\boldsymbol{\chi}, \mathbf{f}) = \frac{\partial C}{\partial \mathbf{f}}^\top = \sum_i \left(\frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}\right)^\top \mathbf{z}_i = 0$$

The values of the errors $\mathbf{z}_i$ being often small around the minimum, we can neglect in the derivatives of $\Phi$ the terms of the form $z.z''$ with respect to the terms of

the form $z'.z'$ (see (Gill *et al.*, 1981, section 4.7)). We obtain thus

$$\mathbf{H} = \frac{\partial \Phi}{\partial \mathbf{f}} \simeq \sum_i \left(\frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}\right)^\top \left(\frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}\right)$$
$$\frac{\partial \Phi}{\partial \boldsymbol{\chi}} \simeq \sum_i \left(\frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}\right)^\top \left(\frac{\partial \mathbf{z}_i}{\partial \boldsymbol{\chi}}\right)$$

The covariance matrix of the state is then given by equation 3:

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{ff}} &= \mathbf{H}^{-1} \left(\sum_i \frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}^\top \frac{\partial \mathbf{z}_i}{\partial} \boldsymbol{\Sigma} \frac{\partial \mathbf{z}_i}{\partial}^\top \frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}\right) \mathbf{H}^{-1} \\ &= \mathbf{H}^{-1} \left(\sum_i \frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}^\top \boldsymbol{\Sigma}_{\mathbf{z}_i\mathbf{z}_i} \frac{\partial \mathbf{z}_i}{\partial \mathbf{f}}\right) \mathbf{H}^{-1}\end{aligned}$$

In our case, the error vectors $\mathbf{z}_i$ are independent and have identical diagonal covariances: $\mathbf{z}_i = (0, 2\sigma^2\,\mathbf{I_d})$ (it is under this assumption that the least-squares estimator is optimal). The central term in the previous equation collapse then into $2\sigma^2\mathbf{H}$ and the covariance of the motion can then be simplified in

$$\boldsymbol{\Sigma}_{\hat{\mathbf{f}}\hat{\mathbf{f}}} = 2\sigma^2\mathbf{H}^{-1}\mathbf{H}\mathbf{H}^{-1} = 2\sigma^2\mathbf{H}^{-1} \qquad (9)$$

with

$$\mathbf{H} = \sum_i \left(\frac{\partial(\hat{\mathbf{f}} \star \mathbf{x}_i)}{\partial \hat{\mathbf{f}}}\right)^\top \left(\frac{\partial(\hat{\mathbf{f}} \star \mathbf{x}_i)}{\partial \hat{\mathbf{f}}}\right)$$

*Estimation from matched frames* We now face the following problem. Assume we have two sets of matched frames $\{\mathbf{f}_{m_i}\}$ and $\{\mathbf{f}_{s_i}\}$ ($m$ for model and $s$ for scene), transformed one into another with a global rigid motion $\mathbf{f}$ such that $\mathbf{f}_{s_i} = \mathbf{f} \circ \mathbf{f}_{m_i}$. We only have access to their measured values $\hat{\mathbf{f}}_{s_i} = \mathbf{f}_{s_i} \circ \mathbf{e}_{s_i}$ and $\hat{\mathbf{f}}_{m_i} = \mathbf{f}_{m_i} \circ \mathbf{e}_{m_i}$ and we want to estimate the rigid motion $\mathbf{f}$ and the uncertainty $\boldsymbol{\Sigma}_{\mathbf{ff}}$ of that estimate. We assume moreover that the compositive noises are independent and centered: $\mathbf{e}_{m_i} = (\,\mathbf{I_d}, \boldsymbol{\Sigma}_{m_i m_i})$ and $\mathbf{e}_{s_i} = (\,\mathbf{I_d}, \boldsymbol{\Sigma}_{s_i s_i})$.

Combining the two measurement equations with the hypothesis and isolating the error term, we get our error vector:

$$\mathbf{z}_i = \hat{\mathbf{f}}_{s_i}^{(-1)} \circ \mathbf{f} \circ \hat{\mathbf{f}}_{m_i} = \mathbf{e}_{s_i}^{(-1)} \circ \mathbf{e}_{m_i} \qquad (10)$$

Here, we are using a particular aspect of our representation for frames: identity corresponds to a null translation and a null rotation vector. This allows us to solve this estimation problem using the EKF.

An initial estimate is easily obtained with the first couple of frames: $\hat{\mathbf{g}}_0 = \hat{\mathbf{f}}_{s_0} \circ \hat{\mathbf{f}}_{m_0}^{(-1)}$. From the observation vector:

$$\hat{\boldsymbol{\chi}} = \left| \begin{matrix} \hat{\mathbf{f}}_m \\ \hat{\mathbf{f}}_s^{(-1)} \end{matrix} \right. \qquad \boldsymbol{\Omega} = \begin{bmatrix} \boldsymbol{\Sigma}_{\mathbf{f}_m\mathbf{f}_m} & 0 \\ 0 & \boldsymbol{\Sigma}_{\mathbf{f}_s^{(-1)}\mathbf{f}_s^{(-1)}} \end{bmatrix}$$

and the estimations $\hat{\mathbf{J}}_1, \hat{\mathbf{J}}_2, \hat{\mathbf{J}}_3$ and $\hat{\mathbf{J}}_4$ at $(\hat{\mathbf{f}}_{m_i}, \hat{\mathbf{f}}_{s_i}, \hat{\mathbf{g}}_{i-1})$ of the following Jacobians:

$$\mathbf{J}_1 = \frac{\partial(\mathbf{g} \circ \mathbf{f}_m)}{\partial \mathbf{g}} \qquad \mathbf{J}_2 = \frac{\partial\left(\mathbf{f}_s^{(-1)} \circ (\mathbf{g} \circ \mathbf{f}_m)\right)}{\partial \mathbf{f}_s^{(-1)}}$$
$$\mathbf{J}_3 = \frac{\partial\left(\mathbf{f}_s^{(-1)} \circ (\mathbf{g} \circ \mathbf{f}_m)\right)}{\partial (\mathbf{g} \circ \mathbf{f}_m)} \qquad \mathbf{J}_4 = \frac{\partial\left((\mathbf{f}_s^{(-1)} \circ \mathbf{g}) \circ \mathbf{f}_m\right)}{\partial \mathbf{f}_m}$$

we can simplify the Jacobians of the error vector:

$$\frac{\partial \mathbf{z}}{\partial \mathbf{g}} = \mathbf{J}_3.\mathbf{J}_1 \qquad \frac{\partial \mathbf{z}}{\partial \boldsymbol{\chi}} = [\mathbf{J}_4\,;\,\mathbf{J}_2]$$

and its covariance matrix is given by

$$\boldsymbol{\Sigma}_{\mathbf{zz}} = \hat{\mathbf{J}}_4 \boldsymbol{\Sigma}_{\mathbf{f}_m\mathbf{f}_m} \hat{\mathbf{J}}_4^\top + \hat{\mathbf{J}}_2 \boldsymbol{\Sigma}_{\mathbf{f}_s^{(-1)}\mathbf{f}_s^{(-1)}} \hat{\mathbf{J}}_2^\top \qquad (11)$$

The EKF estimation can thus be summarized as:

- Initialize the state with

$$(\hat{\mathbf{g}}_0\,,\ \hat{\mathbf{G}}_0) = (\hat{\mathbf{f}}_{s_0} \circ \hat{\mathbf{f}}_{m_0}^{(-1)}\,,\ \boldsymbol{\Sigma}_{(\mathbf{f}_{s_0} \circ \mathbf{f}_{m_0}^{(-1)})(\mathbf{f}_{s_0} \circ \mathbf{f}_{m_0}^{(-1)})})$$

- For each couple of matched frames $(\hat{\mathbf{f}}_{m_i}, \hat{\mathbf{f}}_{s_i})$:
  - Compute $\hat{\mathbf{z}}_i = \hat{\mathbf{f}}_{s_i}^{(-1)} \circ \hat{\mathbf{g}}_{i-1} \circ \hat{\mathbf{f}}_{m_i}$ and the Jacobians $\hat{\mathbf{J}}_1, \hat{\mathbf{J}}_2, \hat{\mathbf{J}}_3, \hat{\mathbf{J}}_4$.
  - Compute $\mathbf{M}_i = \hat{\mathbf{J}}_3.\hat{\mathbf{J}}_1$ and $\boldsymbol{\Sigma}_{\mathbf{z}_i\mathbf{z}_i}$ using Eq.(11).
  - Update $(\hat{\mathbf{g}}_{i-1}, \hat{\mathbf{G}}_{i-1})$ to $(\hat{\mathbf{g}}_i, \hat{\mathbf{G}}_i)$ using Eqs.(7).

*Fusion of rigid motions or frames* Consider now a set $\{\hat{\mathbf{f}}_i\}$ of measurements of the same frame $\mathbf{f}$ such as $\hat{\mathbf{f}}_i = \mathbf{f} \circ \mathbf{e}_i$. The goal is to estimate the frame $(\mathbf{f}, \boldsymbol{\Sigma}_{\mathbf{ff}})$. Assuming an error $\mathbf{e}_i = (\,\mathbf{I_d}, \boldsymbol{\Sigma}_{ii})$ on each measurement, the error vector is

$$\mathbf{z}_i = \hat{\mathbf{f}}_i^{(-1)} \circ \mathbf{f} = \mathbf{e}_i^{(-1)} \qquad (12)$$

An initial estimate is easily obtained with the first frame: $\hat{\mathbf{g}}_0 = \hat{\mathbf{f}}_0$. The measurement $\hat{\boldsymbol{\chi}}$ is now simply $\hat{\boldsymbol{\chi}} = \hat{\mathbf{f}}^{(-1)}$ and its covariance matrix $\boldsymbol{\Omega} = \boldsymbol{\Sigma}_{\mathbf{f}^{(-1)}\mathbf{f}^{(-1)}}$. The EKF process can thus be summarized as:

- Initialize the state with $(\hat{\mathbf{g}}_0\ ,\ \hat{\mathbf{G}}_0) = (\hat{\mathbf{f}}_0, \mathbf{\Sigma}_{\mathbf{f}_0\mathbf{f}_0})$
- For each measurement of frame $\hat{\mathbf{f}}_i$:
  - Compute $\hat{\mathbf{z}}_i = \hat{\mathbf{f}}_i^{(-1)} \circ \hat{\mathbf{g}}_{i-1}$
  - Compute $\mathbf{M}_i = \frac{\partial(\mathbf{f}^{(-1)} \circ \mathbf{g})}{\partial \mathbf{g}}$ and $\hat{\mathbf{J}}_1 = \frac{\partial(\mathbf{f}^{(-1)} \circ \mathbf{g})}{\partial \mathbf{f}^{(-1)}}$ estimated at $(\hat{\mathbf{f}}_i^{(-1)}, \hat{\mathbf{g}}_{i-1})$
  - Compute $\mathbf{\Sigma}_{\mathbf{z}_i\mathbf{z}_i} = \hat{\mathbf{J}}_1 \mathbf{\Sigma}_{\mathbf{f}^{(-1)}\mathbf{f}^{(-1)}} \hat{\mathbf{J}}_1^\top$
  - Update $(\hat{\mathbf{g}}_{i-1}, \hat{\mathbf{G}}_{i-1})$ to $(\hat{\mathbf{g}}_i, \hat{\mathbf{G}}_i)$ using Eqs.(7).

### *3.2. Estimation of the noise process on features*

In the previous section, we assumed that the noise process corrupting the measurements was known, and we estimated the motion. We are now trying to estimate the noise process while the motion is known (or already estimated). In fact, we only estimate its covariance matrix since the mean value should be null with our hypotheses (which is verified experimentally).

*Estimation of the noise on points*

*Anisotropic noise:* Let $\{\mathbf{x}_i\}$ and $\{\mathbf{y}_i\}$ be two sets of matched points with measured values $\hat{\mathbf{x}}_i = \mathbf{x}_i + \delta\mathbf{x}_i$ and $\hat{\mathbf{y}}_i = \mathbf{y}_i + \delta\mathbf{y}_i$, and $\hat{\mathbf{f}}$ be an estimate of the motion $\mathbf{f} = (\mathbf{r}, \mathbf{t})$ linking them together ($\mathbf{y}_i = \mathbf{f} \star \mathbf{x}_i$).

The error vector $\mathbf{z}_i = \hat{\mathbf{y}}_i - \hat{\mathbf{f}} \star \hat{\mathbf{x}}_i$ is, up to the first order, $\mathbf{z}_i = \delta\mathbf{y}_i - \mathbf{r} \star \delta\mathbf{x}_i$ if we assume an exact motion (this assumption will be removed later).

Assuming that each measurement error $\delta\mathbf{x}_i$ and $\delta\mathbf{y}_i$ comes from noise processes of covariances $\mathbf{\Sigma}_{\mathbf{xx}}$ and $\mathbf{\Sigma}_{\mathbf{yy}}$, the covariance on the error vector $\mathbf{z}_i$ should be $\mathbf{\Sigma}_{\mathbf{zz}} = \mathbf{\Sigma}_{\mathbf{yy}} + \mathbf{R}\mathbf{\Sigma}_{\mathbf{xx}}\mathbf{R}^\top$, which can be estimated by

$$\hat{\mathbf{\Sigma}}_{\mathbf{zz}} = \frac{1}{N} \sum_i \mathbf{z}_i\, \mathbf{z}_i^\top \tag{13}$$

If we assume that one set of points is exact, then we can solve for the uncertainty of the other:

$$\mathbf{\Sigma}_{\mathbf{xx}} = 0 \quad \Longrightarrow \quad \hat{\mathbf{\Sigma}}_{\mathbf{yy}} = \hat{\mathbf{\Sigma}}_{\mathbf{zz}}$$

and

$$\mathbf{\Sigma}_{\mathbf{yy}} = 0 \quad \Longrightarrow \quad \hat{\mathbf{\Sigma}}_{\mathbf{xx}} = \mathbf{R}^\top \hat{\mathbf{\Sigma}}_{\mathbf{zz}} \mathbf{R}$$

If we assume that both sets have the same noise $\mathbf{\Sigma}$, we obtain $\hat{\mathbf{\Sigma}}_{\mathbf{zz}} = \mathbf{\Sigma} + \mathbf{R}\mathbf{\Sigma}\mathbf{R}^\top$, which is uniquely solvable for $\mathbf{\Sigma}$ unless $R$ is a rotation of angle $\theta = \pi/2$ or $\theta = \pi$ (see (Koch, 1988) for a solving method). If the rotation between the two sets is small, the covariance on points is approximated at first order by

$$\hat{\mathbf{\Sigma}}_{\mathbf{xx}} = \hat{\mathbf{\Sigma}}_{\mathbf{yy}} = \frac{\hat{\mathbf{\Sigma}}_{\mathbf{zz}}}{2} = \frac{1}{2\,N} \sum_i \mathbf{z}_i\, \mathbf{z}_i^\top$$

*Isotropic noise:* We have in this case $\mathbf{\Sigma}_{\mathbf{xx}} = \mathbf{\Sigma}_{\mathbf{yy}} = \mathbf{\Sigma} = \sigma^2\, \mathbf{I_d}$, and hence the covariance matrix of the error vector should be $\mathbf{\Sigma}_{\mathbf{zz}} = 2\,\sigma^2\, \mathbf{I_d}$. Using the estimation proposed in equation (13) and taking the trace, we obtain:

$$\mathrm{Tr}(\hat{\mathbf{\Sigma}}_{\mathbf{zz}}) = 2\hat{\sigma}^2.\, d = \frac{1}{N} \sum \|\mathbf{z}_i\|^2$$

where $d$ is the dimension of the space ($d = 3$ in our case). Since $\hat{C} = \frac{1}{2} \sum \|\mathbf{z}_i\|^2$ is the value of the least-squares criterion at the minimum, the variance estimation can be summarized as

$$\hat{\sigma}^2 = \frac{\hat{C}}{d\,N}$$

*Estimation of the noise on frame features*

*Standard noise process:* With the same notations as in the previous sections, and neglecting once again the error on the estimate $\hat{\mathbf{f}}$ of the motion $\mathbf{f}$, the error is:

$$\mathbf{e}_i = \hat{\mathbf{f}}_{s_i}^{(-1)} \circ \hat{\mathbf{f}} \circ \hat{\mathbf{f}}_{m_i} = \mathbf{e}_{s_i}^{(-1)} \circ \mathbf{e}_{m_i}$$

Assuming a common underlying process of measurement errors $\mathbf{e}_{s_i} = \mathbf{e}_{m_i} = (\,\mathbf{I_d}, \mathbf{\Sigma})$, we can determine that $\mathbf{e}_{s_i}^{(-1)} = (\,\mathbf{I_d}, \mathbf{\Sigma})$ (since the Jacobian of the inversion is $-\,\mathbf{I_d}$ at the origin), hence $\mathbf{e}_i = (\,\mathbf{I_d}, 2\mathbf{\Sigma})$. It should be noted that the covariances behave here as usual, but only because errors are close to identity. An estimator of the covariance $\mathbf{\Sigma} = \mathbf{\Sigma}_{m_i m_i} = \mathbf{\Sigma}_{s_i s_i}$ of frame features is given by

$$\hat{\mathbf{\Sigma}} = \frac{1}{2\,N} \sum_i \mathbf{e}_i\, \mathbf{e}_i^\top$$

*A simplified model:* In some cases, for example with a small number of matches, the estimation of the above covariance matrix can be unstable. We can use instead a kind of isotropic model of noise with a standard deviation $\sigma_\theta$ for the rotation part and $\sigma_d$ for the translation part. Splitting the error vector into a rotation and a translation component $\mathbf{e}_i^\top = (\mathbf{e}_{\theta_i}^\top,\ \mathbf{e}_{d_i}^\top)$, we can then estimate the variances by

$$\hat{\sigma}_\theta^2 = \frac{\sum \|\mathbf{e}_{\theta_i}\|^2}{6\,N} \quad \text{and} \quad \hat{\sigma}_d^2 = \frac{\sum \|\mathbf{e}_{d_i}\|^2}{6\,N}$$

*Discussion about noise estimation:* Several problems arise in the covariance estimation scheme: assuming a Gaussian noise model with outliers (contaminated Gaussian for instance), it is clear that estimating the covariance matrix on all points, including outliers, will produce a significant elevation of the covariance.

On the other side, estimating on inliers only will induce an under-estimation which corresponds, on a theoretical point of view, to the following partial integration on the inlier domain $\mathcal{D}$

$$\mathbf{\Sigma}' = \int_{\mathcal{D}} (\mathbf{x} - \bar{\mathbf{x}}).(\mathbf{x} - \bar{\mathbf{x}})^\top . \rho(\mathbf{x}).\mathbf{dx}$$

instead of the whole space $\mathbb{R}^d$ for the real covariance matrix $\mathbf{\Sigma}$.

With the Gaussian assumption, and a $\chi^2$ threshold on the Mahalanobis distance selection (see the following section), this equation reduces to $\mathbf{\Sigma}' = \alpha(\chi).\mathbf{\Sigma}$ where the correction factor $\alpha$ is defined by an integral of the $\chi^2$ distribution. In practice, this factor is only slightly inferior to one with standard values of the $\chi^2$ threshold, and we prefer to compute the covariance with matches that have a Mahalanobis distance less than 1.5 or 2 times the $\chi^2$ threshold. This includes only the outliers that are almost inliers and ensures a very small underestimating bias with usual values of the $\chi^2$ threshold. This remains, however, the weak point of our method concerning robustness, at least theoretically (see section 4.2).

Another bias in the covariance estimation comes from the lest-squares technique we choose to determine the motion (which is indeed not exact): if one set of features is an exact model, the residuals are a good estimation of the real error, but when both sets of features are corrupted by noise, the residue minimization makes the residuals slightly inferior to the real error. From (Bard, 1974), the number of observations $N$ in all our covariance estimations should thus be replaced by $N' = N - l/m$, where $m$ is the dimension of the vectorial equations and $l$ the number of parameters we have estimated. In our case, this is a rigid motion estimation, so $l = 6$. If we estimate it from point matches, we have $m = 3$ whereas $m = 6$ from frames. We should then replace $N$ with $N'$ is the previous equations, where

- $N' = N - 2$ for points,
- $N' = N - 1$ for frames.

### 3.3. *Rejecting outliers: Mahalanobis distance and $\chi^2$ test*

Another interesting problem that arises within this framework is compatibility. For a given motion, we want to know, for instance, if a scene feature is compatible with a model feature, i.e. if the scene feature can be considered as the transformation of the model feature modulo measurement errors. Considering sets of matched features, we may also want to sort the matches by saliency. These two problems are important for the estimation since least squares techniques are known to be sensitive to outliers, and the EKF is moreover order dependent: a preliminary phase where matches are sorted by relevancy and outliers rejected often gives substantial improvements on the quality of the estimation.

A related question is to determine when two measurements originate from the same object and differ only because of measurement error, This can be used, for instance, for clustering motions, i.e. to find subgroups of consistent motions. A fusion could then be used within each subgroup to obtain a better estimate of the motion.

Such problems are traditionally tackled via Mahalanobis distances for sorting hypotheses and $\chi^2$ tests for rejecting outliers. Until now, we did not need to assume Gaussian distributions on our random variables. For the $\chi^2$ test, however, this assumption is required. Two main reasons allow us to assume that the distributions of the noise is Gaussian: firstly, the measurement errors are often the sum of independent errors and thus tend toward a Gaussian process by the Central Limit Theorem. Secondly, when the mean and variance of an unknown distribution are the only information available (which is our case), a simple maximum entropy derivation gives the Gaussian distribution as the one that assumes the minimal information (Bard, 1974).

*Two points and a motion* Let $(\mathbf{x}, \mathbf{\Sigma_{xx}})$ and $(\mathbf{y}, \mathbf{\Sigma_{yy}})$ be two measured points and $(\mathbf{f}, \mathbf{\Sigma_{ff}})$ a rigid motion. We want to test the validity of the hypothesis $\mathbf{y} = \mathbf{f} \star \mathbf{x}$.

We saw in section 2.4 how to compute $(\mathbf{f} \star \mathbf{x}, \mathbf{\Sigma_{(f\star x)(f\star x)}})$ from $(\mathbf{x}, \mathbf{\Sigma_{xx}})$ and $(\mathbf{f}, \mathbf{\Sigma_{ff}})$. If the motion was independent of the data $\mathbf{x}$ and $\mathbf{y}$, then the covariance on the error vector $\mathbf{z} = \mathbf{y} - \mathbf{f} \star \mathbf{x}$ should be zero with covariance $\mathbf{\Sigma} = \mathbf{\Sigma_{yy}} + \mathbf{\Sigma_{(f\star x)(f\star x)}}$. In our case, the motion is computed from the data and the independence assumption does not hold. Thus we have to use the covariance of the residue (Förstner, 1987):

$$\mathbf{\Sigma} = \mathbf{\Sigma_{yy}} + \left(\frac{\partial(\mathbf{f} \star \mathbf{x})}{\partial \mathbf{x}}\right) . \mathbf{\Sigma_{xx}} \left(\frac{\partial(\mathbf{f} \star \mathbf{x})}{\partial \mathbf{x}}\right)^\top$$

which does not take into account the uncertainty due to the motion (it is often very small with respect to the features uncertainty). A $\chi^2$ test is then well suited to verify if $\mathbf{z}$ is compatible with its theoretical covariance. The squared Mahalanobis distance $\mu^2$ is

$$\mu^2 = \mathbf{z}^\top \mathbf{\Sigma}^{(-1)} \mathbf{z}$$

The statistical test says that the hypothesis is good if $\mu^2 \leq \varepsilon$, where the threshold $\varepsilon$ is set such that, if the hypothesis is true, we will choose it with the probability $\alpha$. Some values of $\varepsilon$ and $\alpha$ for this 3-D $\chi^2$ test are given found in table 1: for example, if $\mu^2 > 11.34$, there is less than 1% of probability that $\mathbf{y}$ be a measurement of $\mathbf{f} \star \mathbf{x}$. We can thus reject this match.

*Two frames and a motion* Let $(\mathbf{f}_m, \mathbf{\Sigma}_{mm})$ and $(\mathbf{f}_s, \mathbf{\Sigma}_{ss})$ be two measured frames and $(\mathbf{f}, \mathbf{\Sigma})$ a rigid motion. We want to test the hypothesis that $\mathbf{f}_s = \mathbf{f} \circ \mathbf{f}_m$. From section 2.4 we can compute the covariance $\mathbf{\Sigma}$ on the error vector $\mathbf{e} = \mathbf{f}_s^{(-1)} \circ \mathbf{f} \circ \mathbf{f}_m$ (with the same remark as for points concerning the independence of the motion) and use the squared Mahalanobis distance $\mu^2 = \mathbf{e}^\top \mathbf{\Sigma}^{(-1)} \mathbf{e}$ and a 6-D $\chi^2$ test to decide if the hypothesis is true or not (see table 1).

*Two frames* Let $(\mathbf{f}_1, \mathbf{\Sigma}_{11})$ and $(\mathbf{f}_2, \mathbf{\Sigma}_{22})$ be two measured frames. We want to test the hypothesis that they are measurements of the same frame $\mathbf{f}$. Let $\mathbf{\Sigma}$ be the covariance on the error vector $\mathbf{e} = \mathbf{f}_1^{(-1)} \circ \mathbf{f}_2$. We can once again use the squared Mahalanobis distance $\mu^2 = \mathbf{e}^\top \mathbf{\Sigma}^{(-1)} \mathbf{e}$ and a 6-D $\chi^2$.

### *3.4. Conclusion: uncertainty and rigid motions*

We introduce in this section a new model of noise based on composition and show why it is better adapted than the classical additive noise. Estimation problems within this framework require the computation of a covariance matrix of the estimate: the Extended Kalman Filter provides a unified formalism to handle this. At last, the Mahalanobis distances and $\chi^2$ tests allow the comparison and discrimination of hypotheses based on frames and rigid motions.

This framework can be used in a large number of vision problems. We present in the next section an application to registration; but we could also use this framework for matching algorithms (see (Pennec & Ayache, 1994)). Further applications include the clustering of rigid motions between image structures, or rigid motions in a displacement field, in order to detect and isolate objects. This can also prove to be useful when dealing with the accumulation stage of geometric hashing or Hough transform algorithms.

*Table 1.* Table of the $\chi^2$ distribution for 3 and 6 degrees of freedom.

| $\alpha$ | Dim | 50% | 90 % | 95 % | 99% |
|---|---|---|---|---|---|
| $\varepsilon$ | 3 | 2.37 | 6.25 | 7.81 | 11.34 |
| $\varepsilon$ | 6 | 5.35 | 10.65 | 12.59 | 22.46 |

## 4. A practical and robust algorithm for motion estimation

The output of a feature-based matching algorithm usually consists in two sets of matched features. The basic idea is that if we can compute a reliable estimation of the motion, and the confidence that we have in it, we can give a confidence value on the registration at each point of the image.

The method is the following. We take as input of the evaluation step two sets of matched features and compute the rigid motion between them. Assuming that all the features are corrupted by the same noise process, we compute statistics (i.e. the covariance matrix of the process) on features, and use this information to recompute the rigid motion and its associated covariance. This iterative process can be continued until convergence. This allows us to predict a variance on the position of each registered point of the image.

### *4.1. The iterative process*

The considered features are again points and frames, but initially without any uncertainty information[1]. Without loss of generality, such a framework can be extended to other types of features. In order to use an EKF, we first need to estimate the features covariance matrix (this is detailed in section 3.2). We start with

1. A least squares estimate of $\mathbf{f}$, using point positions only, or directly a least median of squares on frames (see section 4.2).
2. Estimate the noise process on features.
3. Order matches by increasing Mahalanobis distances.

We now have enough information on the features to start the iterative process. Step 1 can be ignored if a reliable motion is provided, as step 3 if matches are already sorted with respect to some other criterion. This ordering of matches is, however, needed to ensure the robustness of the EKF: since the equations are linearized in the EKF, the estimate is order-dependent. If the differences are usually not significant, some special cases can lead to quantitative errors. Step 2 is not used if an estimation of the noise process (even a gross estimation) is given on input (which is what we usually do unless it is the first time we use a new type of data).

Ordering also helps outliers rejection for the motion computation while keeping these matches for other purposes. In protein substructure matching, for instance in (Pennec & Ayache, 1994), we want to compute the motion with reliable amino-acids matches (which can have some precise interactions with other molecules), but we also want to see less constrained matches that point out other structural similarities.

The iterative process itself is as follows.

1. Estimate the motion $(\mathbf{f}, \mathbf{\Sigma_{ff}})$ from the matches using an EKF (in decreasing significance order) until the Mahalanobis distance of the matches becomes larger than a predetermined $\varepsilon$ threshold (this is the $\chi^2$ test to reject outliers).
2. Estimate the noise process with all features (see section 3.2).
3. Sort matches by their Mahalanobis distance. Outliers are the tail of the list.

The process is repeated until convergence, or for a preset maximum number of iterations (typically 5 to 10). Convergence means in particular that the current estimate of the motion is exactly the same as at the previous step.

The estimation of the noise process is the weak point of our scheme for robustness since it is relatively error-prone. This step is dismissed if a reliable estimation of this noise is furnished, but when it comes to a new type of data, we need to estimate it. In practice, we estimate the noise on features (*i.e.* its covariance matrix) only once per three or four iterations in order to stabilize the convergence (in this case around 15 to 20 iterations), and results on real data appear to be quite accurate. A rigorous treatment of this point would require the robust estimation of a covariance matrix.

### *4.2. Variations on the algorithm*

Since a frame is composed of a point and a trihedron, we can also apply point techniques to frames. On this basis, we can then distinguish four ways of estimating both the motion and the noise process: EKF on frames with complete or simplified noise model, EKF on points with a complete noise model (on points), and standard least-squares on points (isotropic noise model).

The appropriate method depends upon the number of matches, the relative quality of the trihedron and the isotropy of the noise. A comparison of the different methods is presented in section 6.1.

In terms of robustness (see (Huber, 1981; Meer *et al.*, 1991) for a review on robust statistics), the algorithm we propose belongs to the class of iterative redescending M-estimators (i.e. robust algorithms based on least squares). However, the breakdown point (the maximum amount of outliers before failure) is not very high. The algorithm can be robustified to a breakdown point of about 0.5 by using a least median of squares between the initialization step (in order to have an estimation of the noise process) and the iterative process, that achieve the goal of efficiency (i.e. reaches a variance on the estimate close to the lowest possible one). This is particularly easy and cheap in complexity for frames. Indeed, a single match determines a unique motion between the model and scene sets: for each of the $N$ frame matches, we classify the other matches by increasing Mahalanobis distances and take the value of the median for the score of this match. The match minimizing the median squared Mahalanobis distance is then used as the initial motion for the iterative process. The complexity of $O(N^2 \log N)$ can be drastically reduced by a Monte Carlo sampling (see (Meer *et al.*, 1991)).

### *4.3. Final Precision evaluation*

From $(\hat{\mathbf{f}}, \mathbf{\Sigma_{ff}})$, we can compute for each point $\mathbf{x}$ of the model image (or object) its transformation $\hat{\mathbf{y}} = \hat{\mathbf{f}} \star \mathbf{x}$ and the uncertainty $\mathbf{\Sigma}_{\hat{\mathbf{y}}\hat{\mathbf{y}}} = \mathbf{\Sigma}_{(\hat{\mathbf{f}}\star\mathbf{x})(\hat{\mathbf{f}}\star\mathbf{x})}$. This information could be used directly as an input for other statistical algorithms but is too rich to give a simple idea of the level of accuracy for the end user. In order to give a more intuitive information, we compute instead the RMS error expectation at this point: if $\mathbf{y} = \mathbf{f} \star \mathbf{x}$ is the real location of the transformed point $\mathbf{x}$, we shall

find it in $\hat{\mathbf{y}} = \hat{\mathbf{f}} \star \mathbf{x}$. The expectation of the squared distance $d^2 = \|\mathbf{y} - \hat{\mathbf{y}}\|^2$ between them is

$$\sigma^2(\hat{\mathbf{y}}) = E\left((\hat{\mathbf{y}} - \mathbf{y})^\top (\hat{\mathbf{y}} - \mathbf{y})\right) = \mathrm{Tr}(\mathbf{\Sigma}_{\hat{\mathbf{y}}\hat{\mathbf{y}}})$$

In the case of 3-D medical images, and in order to characterize the precision with a single value, we can estimate (among other measurements) the averaged RMS for the 8 vertices of the 3D image (we call this value the *typical boundary precision* $\sigma_{corner}$), which is representative of the maximum error, or estimate the averaged RMS on the position of the matches frames (the *typical object precision* $\sigma_{object}$).

### *4.4. Conclusion on registration and accuracy*

From two sets of matched features, and a $\chi^2$ threshold $\varepsilon$, we propose a method to estimate the motion $\mathbf{f}$ and its precision $\mathbf{\Sigma}_{\mathbf{ff}}$, discard outliers, and provide an estimation of the noise process on features. Then we extract from the motion uncertainty a single value characterizing the registration accuracy.

## 5. A statistical validation of registration methods

We have now a way to estimate motions and associated covariances, but what confidence can we have in it ? We propose in this section a method to estimate the accuracy of a registration algorithm when no "ground truth" is available. It also validates the estimation process for feature based registration methods which generate covariance matrices. This is a statistical method which only assumes that the noise on features is centered.

### *5.1. External markers*

We believe that no method can give an exact reference for the motion, but when a method A is one order of magnitude more accurate than a method B, A can serve as a reference for an experimental validation of B. This means that all methods are ultimately statistical.

Debates about the use of external markers are typical in the case of 3D medical image registration problems. Here, validation methods are vital (literally), because the registration can be used to plan a surgical procedure. In fact, external markers have a very strong advantage in matching methods, because we can extract features corresponding to physical and discriminable objects and obtain an "exact" matching. But regarding registration, we have to keep in mind that the measurements of marker positions are corrupted just as the measurement of any other landmark, which implies the elaboration of a theoretical model of the markers, and the application of a statistical method to evaluate the motion and its uncertainty. Other authors have concluded experimentally that, for the case of high resolution medical images, marker-based techniques are probably much less accurate than image-based techniques (see (van den Elsen, 1993)).

Hence, marker-based techniques are not more deterministic than other registration methods, and the results of the present paper could also be applied to them (generally, markers are modeled as points or frames).

### *5.2. Analytical methods*

Ideally, we might think of modeling analytically the whole process, from the physical object positions to the final estimated motion. For real cases, this implies modeling object deformations (nothing is really rigid), evaluating the distortions due to image acquisition and reconstruction process, evaluating the errors in the extraction of feature points, before considering the errors made during registration. This is generally impossible to do in practice, when all this also depends on the shape of the object, and the tuning of the acquisition device.

### *5.3. An* a posteriori *estimation of the errors*

The method that we present now applies when there is no ground truth for the motion, and when the analytical determination of the errors is impossible (the majority of cases). We consider the registration method as a black box which takes two representations of the same object as input, and gives an estimated rigid motion as output.

Assuming independence of couples of matched features, we can subsample the matches in order to get several independent estimates of the unknown motion, and compare them. In particular, we can split the set of matches in two sets approximately equal in size and obtain two estimates of $f$:

$$\hat{\mathbf{f}}^1 = \mathbf{f} \circ \mathbf{e}^1 \qquad \text{and} \qquad \hat{\mathbf{f}}^2 = \mathbf{f} \circ \mathbf{e}^2$$

The two motions should be very close, and we can study their "difference" $\mathbf{e} = (\hat{\mathbf{f}}^2)^{(-1)} \circ \hat{\mathbf{f}}^1 = (\mathbf{e}^2)^{(-1)} \circ$

$\mathbf{e}^1$ which does not depend on the exact motion $\mathbf{f}$ and should be close to identity.

If matches are randomly subsampled to a fixed number $M$ of matches, in order to conserve independence and a similar distribution of matches in space, then the estimation errors $\mathbf{e}^1$ and $\mathbf{e}^2$ follow the same law of covariance $\mathbf{\Sigma}(M)$. If moreover both the feature extraction and the registration method are unbiased[2] (which means that $E(\mathbf{e}^1) = E(\mathbf{e}^2) = \mathbf{I_d}$), then we can derive that the covariance of $(\mathbf{e}^2)^{(-1)}$ is also $\mathbf{\Sigma}$ and the measured error $\mathbf{e}$ should then have a covariance $\mathbf{\Sigma_{ee}}(M) = 2\mathbf{\Sigma}(M)$.

Repeating the experiment with $n$ homogeneous sets of matches, we can estimate the covariance matrix of the estimation of $\mathbf{f}$ from $M$ feature matches by

$$\mathbf{\Sigma}(M) = \frac{1}{2\,n} \sum_{i=1}^{n} \mathbf{e}_i\, \mathbf{e}_i^\top$$

### 5.4. A posteriori *validation of the uncertainty*

Our estimation scheme produces not only a motion estimation $\hat{\mathbf{f}}$ but also a covariance estimation on it $\mathbf{\Sigma_{\hat{f}\hat{f}}}$. The goal of this section is to verify *a posteriori* that the distribution of $\hat{\mathbf{f}}$ has for mean value $\mathbf{f}$ with the given covariance. We consider first the case of synthetic data where the exact motion $\mathbf{f}$ is known. Since every experiment is based on a different exact motion and produce a different covariance matrix on the estimate, we have to "normalize" our results in order to obtain several estimates of the same distribution.

To normalize the mean motion, we compute for each registration the error vector $\mathbf{e} = \mathbf{f}^{(-1)} \circ \hat{\mathbf{f}}$ (for simpler notations, the index of the registration experiment is omitted). Its distribution should have a zero mean, but its covariance matrix is given, since $\mathbf{f}$ is exact, by $\mathbf{\Sigma_{ee}} = \mathbf{J}.\mathbf{\Sigma_{\hat{f}\hat{f}}}.\mathbf{J}^\top$ with $\mathbf{J} = \partial\mathbf{e}/\partial\hat{\mathbf{f}}$. A new change of variables is needed to normalize the distributions with respect to this covariance: under the Gaussian hypothesis, the Mahalanobis distance with identity $\mu^2 = \mathbf{e}^\top\, \mathbf{\Sigma}_{\mathbf{ee}}^{(-1)}\mathbf{e}$ should be $\chi^2_6$ distributed.

We can now repeat this experiment on $M$ pairs of images to obtain $M$ independent values $\mu_i^2$ and verify if it is really $\chi^2_6$ distributed. The Kolmogorov-Smirnov test (Press *et al.*, 1991) is well adapted to do that (referred from now on to the K-S test), but since it only gives a binary answer, we also use the fact that the mean value of a $\chi^2_6$ distribution is 6 and its variance is 12: We call *validation index* the estimated mean value of $\mu_i^2$:

$$I = \bar{\mu}^2 = \frac{1}{M} \sum \mu_i^2$$

and its variance is computed accordingly with

$$\sigma_I^2 = \frac{1}{M-1} \sum (\mu_i^2 - \bar{\mu}^2)^2$$

This index can be interpreted as an indications on how the estimation method under-estimates ($I > 6$) or over-estimates ($I < 6$) the error on the estimated motion. It is a kind of relative error on the error estimation.

This method can be generalized to validate our estimation scheme with real data: randomly splitting the set of matches in two (approximately equal) sets, we compute two estimates $(\hat{\mathbf{f}}^1, \mathbf{\Sigma_{f^1 f^1}})$ and $(\hat{\mathbf{f}}^2, \mathbf{\Sigma_{f^2 f^2}})$ which are two independent measures of the exact motion $\mathbf{f}$. Their "difference" $\mathbf{e} = (\hat{\mathbf{f}}^2)^{(-1)} \circ \hat{\mathbf{f}}^1$ can now be used just as above. Some results with synthetic and real data are given in sections 6.1 and 6.2.

### 5.5. *Conclusion on statistical validation*

We show in this section that all validation methods are ultimately statistical and present a quite general method to estimate the quality of a registration process, independently of our previous method. This leads us to the validation of our uncertainty prediction with both synthetic and real data.

## 6. Experiments

The first part of this section deals with the comparison of the four different methods on synthetic data, and shows how to choose the best method among isotropic or anisotropic noise and points or frames features. We will see through these experiments that all the methods are perfectly validated in their own domain. The second and third part focuses on real data, with 3D medical image registration and protein substructure matching.

### 6.1. *Synthetic data experiments*

For our synthetic experiments, we have used lists of associated frames or points with a Gaussian error distribution on the positions and on the orientations and a uniform random placement in 256x256x256 voxels

images (one voxel equal one millimeter). The exact motion is chosen randomly but the translation is limited to a third of the image size in order to obtain an overlap between the two images. The covariance matrix of the simulated noise process is given by the experiments on 3D medical images (see the covariance matrix of table 2). We use a $\chi^2$ value of 16 for frames and 8 for points, which corresponds to a 96% confidence interval, and limit the number of iterations to a maximum of 15.

We also run several other sets of experiments using a real model randomly moved and perturbed, and a uniform model of noise with the same covariance. All those experiments showed very similar results.

*Anisotropic versus simplified noise model* We have presented in section 3.2 two models of noise for points (iso and anisotropic) and their equivalent for frames. The question we investigate here is which one should we use, and does the anisotropic model provide an improved accuracy of the registration? In these experiments, frames are generated with a compositive model of noise and points with an additive one.

We claim that the relevant parameter to decide which method is the most suited is the number of matches. Indeed, the estimation of the 21 parameters (resp. 6 for points) of the covariance matrix is less stable than the estimation of the 2 (resp. 1) parameters of the simplified model, in particular with a small number of samples. However, for a sufficient number of matches, we can estimate the whole covariance matrix with a good confidence, even if the original noise is isotropic.

In figures 3 and 4, we plot on the top the validation index (ideally 6) and its standard deviation. The dotted lines represent the theoretical mean value and standard deviation of a $\chi^2_6$ distribution. We also indicate experiments where the K-S test at 5% rejects the validation. On the bottom of the figures, we plot the typical precisions as a function of the number of matches for isotropic methods and the gain in precision obtained using anisotropic methods. Each sample point is the averaged result of 100 trials.

As expected, the validation index shows that isotropic methods need less matches to have a good uncertainty prediction than anisotropic methods ones: about 10 points are needed for the isotropic method on points to be reliable, versus 15 to 20 matches for the anisotropic one. For frames, these figures are as expected a little bigger: respectively 15 to 20 and 30 to 40 matches are needed. On the other hand, anisotropic methods lead to a more accurate motion estimation: in this case, it is roughly 1.25 times more accurate for frames and between 1.15 and 1.4 times more accurate for points. However, we note that these values are just examples and do vary with the anisotropy of the original noise. From now on, we use the simplified model of noise for less than 40 matches with frames (20 with points) and the anisotropic one otherwise.

The difficult and possibly inaccurate point in our our scheme is therefore the covariance matrix estimation which is, as we have already noted, not really robust. In order to verify the robustness and accuracy of the remainder of the algorithm, we run another series of experiments where we fix the covariance on features (i.e. we consider that it is already known and we do

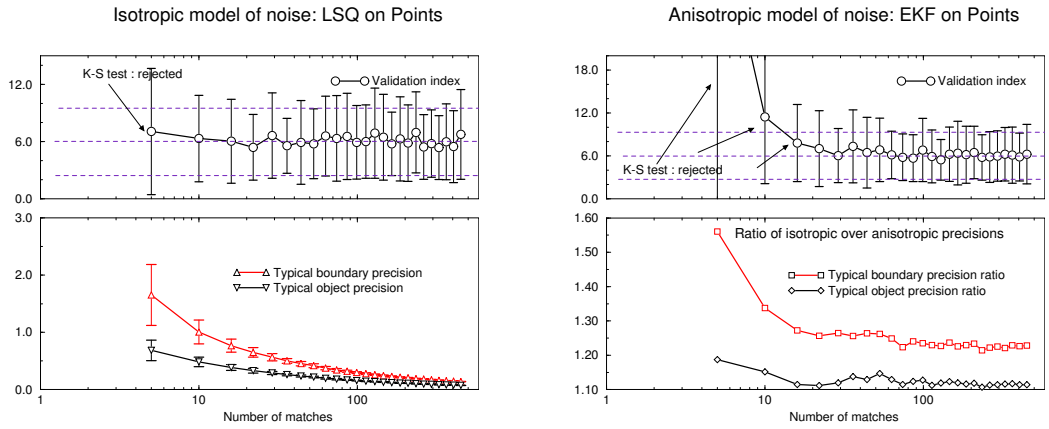


*Fig. 3.* Validation index with respect to the number of matches for anisotropic (top right) and simplified (top left) noise model on point features. The typical precisions are presented for the isotropic model (bottom left), along with the ratio between isotropic and anisotropic precisions (bottom right).

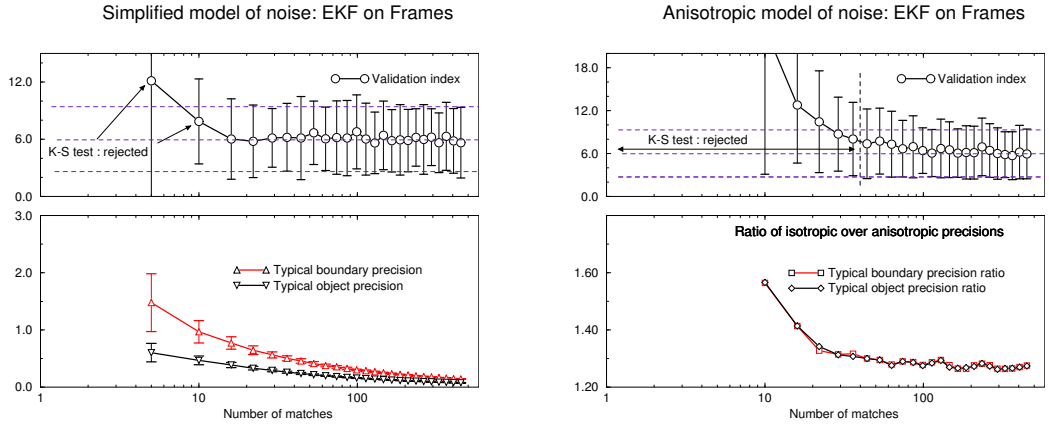


*Fig. 4.* Validation index with respect to the number of matches for anisotropic (top right) and simplified (top left) noise model on frame features. The typical precisions are presented for the isotropic model (bottom left), along with the ratio between isotropic and anisotropic precisions (bottom right).

not have to estimate it) and introduce about 10% of outliers. In all cases, even with 5 points or frames matches, the Kolmogorov-Smirnov test (at 5%) successfully validate the rigid motion and its uncertainty estimation.

*Comparison of Frame and Point based methods* An interesting question is what improvement in accuracy is brought by the use of frames instead of points ? Since we add trihedra to points in order to make frames, frame methods should be at least as accurate as point ones. On the other hand, the extraction of trihedra from real images is usually based on higher derivatives than the point position. Hence, points are usually less noisy than trihedra and we do not expect a breakthrough for accuracy in using frames. However, as points become noisier and noisier, there should be sensible improvement with frames.

We choose to run the experiment with a random number of points between 150 and 250, a fixed standard deviation of $\sigma_\theta = 0.02$ rad for the orientation sampling, and a standard deviation increasing from $\sigma_d = 0.2$ mm to 2.8 mm on point position sampling.

Since the typical precisions do vary with the number of matches and the noise process, but in a similar way for both methods, we plot in figure 5 the ratio of typical precisions (standard deviations) for point and frame with respect to the standard deviation on position. Each sample is the averaged result of a set of 50 synthetic "images".

Using frames in spite of points can then lead to an sensible improvement in accuracy, particularly with a small number of matches, or when points becomes noisy with respect to trihedrons. On the other hand, several experiments showed that the frame based method remain at least as accurate as the point based one even when trihedrons become very noise. In general, we should then use the maximal amount of information we have on features and thus the frame based method if we can define trihedrons.

*Table 2.* Estimated covariance matrix for the compositive noise on extremal points (expressed in the local frame).

$$\boldsymbol{\Sigma}_{\mathbf{ee}} = \begin{array}{c} \\ \mathbf{e_r} \\ \\ \mathbf{e_t} \end{array} \overset{\begin{array}{cccccc} \mathbf{e_r}^\top & & & \mathbf{e_t}^\top & & \end{array}}{\left[\begin{array}{cccc|ccc} 0.0024 & 0.0000 & -0.0000 & 0.0002 & -0.0011 & 0.0000 \\ 0.0000 & 0.0030 & -0.0000 & 0.0001 & -0.0000 & 0.0000 \\ -0.0000 & -0.0000 & 0.0373 & -0.0006 & -0.0003 & 0.0001 \\ \hline 0.0002 & 0.0001 & -0.0006 & 0.2276 & 0.0011 & 0.0008 \\ -0.0011 & -0.0000 & -0.0003 & 0.0011 & 0.3157 & -0.0015 \\ 0.0000 & 0.0000 & 0.0001 & 0.0008 & -0.0015 & 0.0838 \end{array}\right]}$$

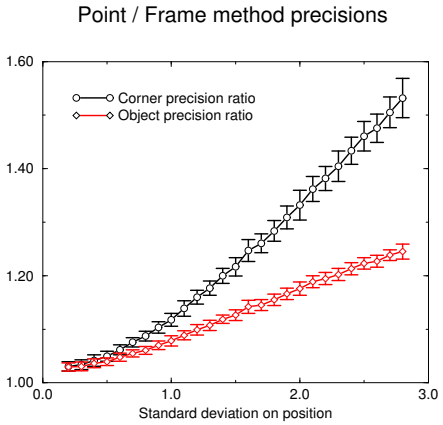


*Fig. 5.* Ratio between Point and Frame based precisions methods with respect to standard deviation on position. The error bar is the standard deviation in 50 trials.

### *6.2. Real data experiments: 3D medical imaging*

We present results from an experiment performed using 3D Magnetic Resonance images (MRI). The images were provided by R. Kikinis from the Brigham and Woman's Hospital, and are part of an extensive study of the evolution of the Multiple Sclerosis (MS) disease. The same patient recieved a complete 3D MR examination several times during one year (typically 24 different 3D acquisitions). The aim is to register precisely in 3D all those images in order to segment the lesions and evaluate very accurately their evolution (MS lesions are white spots in the images in figure 6).

The images are first echo, $256 \times 256 \times 54$ voxels, the voxel size is 1 mm $\times$ 1 mm $\times$ 3 mm. We have already presented an algorithm to perform, fully automatically, the registration of the images in (Thirion, 1994). However, there is a need to estimate accurately the errors in these registrations. The registered images are resampled using a tri-linear interpolation method. The same slices of several different 3D MRI are presented in figure 6, after resampling with the estimated motions. Figure 7 presents the differences between images and shows visually the quality of the registration, and also that the evolutions of the lesions are clearly detected.

*Points and frames in medical images* Our registration algorithm relies on the extraction of feature points in 3D medical images, defined with differential geometry criteria (see figure 8). In our case, these are the *Extremal Points*, as defined in (Thirion & Gourdon, 1993), which are those points of the object surface for which both principal curvatures are extremal. The interesting thing is that not only do we get some invariant measurements associated with those points (the principal curvatures), which are used to reduce the complexity of the matching process, but we get also the principal

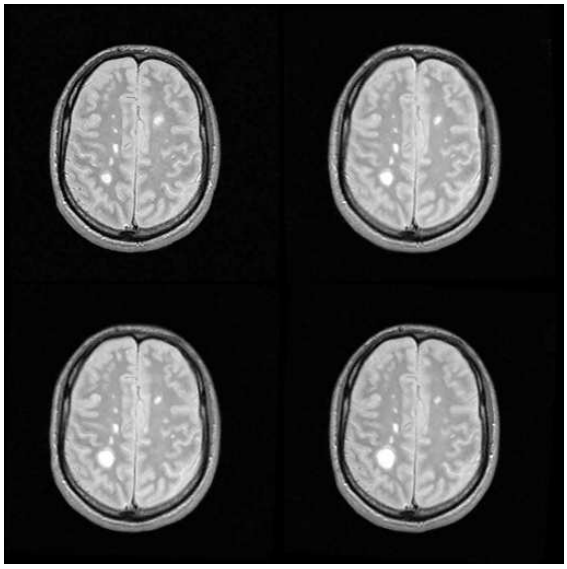

*Fig. 6.* The same slices from four registered 3D MR images of the Multiple Sclerosis study. Note the evolution of two MS lesions (white spots). One is growing in the anterior left hemisphere, one is shrinking in the right posterior hemisphere. There is two weeks between each acquisition.

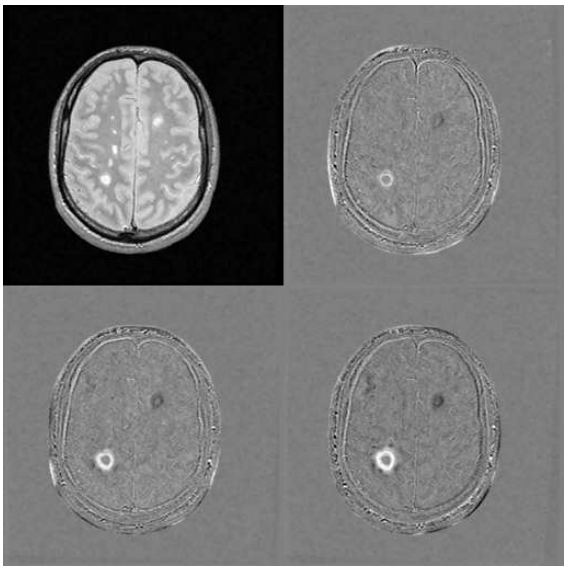

*Fig. 7.* Differences between images after registration, with respect to the first one. The intensity is multiplied by 5, and shifted such that no difference is a gray value. The growing lesion appear as a white disk, and the shrinking lesion as black a disk.

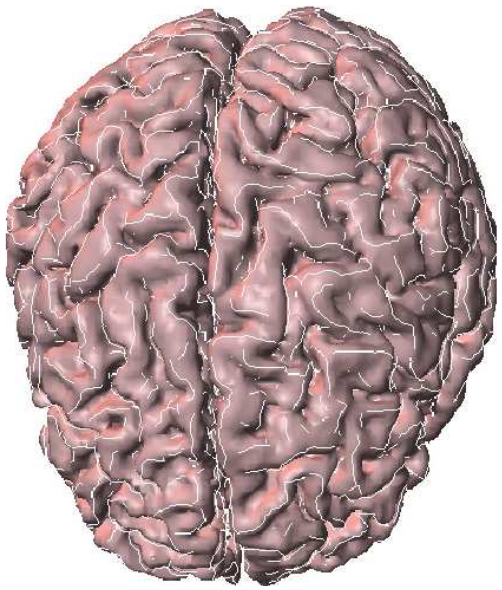

*Fig. 8.* Lines of extremal curvature on the brain circumvolutions. The extremal points, which are used in the registration, are specific points of those lines.

directions, which form, with the normal to the surface and the extremal point itself, an orthonormal basis, that is, a frame (see figure 1 in section 2.2).

Typically, we extract about 2000 extremal points from a 3.5 million voxels image. Our matching algorithm produces about 600 pairs of associated extremal points between two images, with a residual mean square error (RMS) of about $1mm$. With a density fewer than $0.1\%$ of extremal points in the image, the probability of false matches is very low.

*Results* With 24 different images, we could have generated 576 couples of images. In order to achieve independence, we have restricted our experiments to 58 randomly chosen registrations. For each registration, we randomly split the list of matches in two approximately equal lists and compute the validation index $I$ from the two rigid motion estimates. We fuse then the two motion estimates in order to compute the final motion and its uncertainty.

We find a typical boundary precision around $\sigma_{corners} = 0.114$ mm and a typical object precision far below the voxel size: $\sigma_{object} = 0.060$. The validation index is $I = \bar{\mu} = 6.19$ with a variance of $\sigma_I^2 = 17.05$ (remember that the theoretical values are 6 and 12) and the K-S test validate our results for real data with a significance level of 0.16. These values were found using frames and showed an improvement of 10% in accuracy with respect to points.

*Analysis of the estimated noise on features* These experiments showed an interesting difference between the point and frame noise model. Indeed, point based methods gave a quite isotropic noise of standard deviation $\sigma = 0.5$, whereas the frame based method gave the covariance matrix on frame features displayed in Table 2 .

As said in the synthetic experiments section, this covariance is roughly diagonal, with standard deviations $\sigma_{r_x} = 0.05$, $\sigma_{r_y} = 0.055$, $\sigma_{r_z} = 0.20$ for the rotation vector part (in radian), and $\sigma_{d_x} = 0.5$, $\sigma_{d_y} = 0.55$, $\sigma_{d_z} = 0.25$ for the position part (in mm), which gives a mean standard deviation on position of $\sigma = 0.46$, comparable with the one computed on points only.

In order to interpret these values, we have to remember that, with the compositive noise model, the error vector is expressed in the local frame. In our case, the frame is defined by $\{\mathbf{t}_1, \mathbf{t}_2, n\}$ where $\mathbf{t}_1$ and $\mathbf{t}_2$ are the two principal directions of the surface at the extremal point and $\mathbf{n}$ the normal. As far as the rotation vector is concerned, the values of the standard deviations show that the expected error rotation around the $\mathbf{n}$ ($= \mathbf{z}$) axis is four times the expected error rotation around the $\mathbf{t}_1$ or $\mathbf{t}_2$ axes. This means that the normal $\mathbf{n}$ is about four times more stable than the principal directions. Regarding the position, we can see that the coordinate along the normal to the surface is 3 to 4 times more stable than coordinates in the tangent plane, which is in accordance with what was expected for extremal points.

The compositive noise model exhibits here an anisotropy which is not detectable with a classical noise: the absence of orientation information leads to a quasi isotropic noise for points, and an additive noise on frames would fail to detect this regularity since the error would not be expressed in local frames. This effect therefore constitutes an a posteriori justification of our compositive noise model with real data.

*Discussion* Our matching algorithm tries to estimate the motion of a single rigid substructure, with the largest number of common extremal points, which is the brain in those images. However, with that level of accuracy, the assumption of a global rigid motion does not hold any more, and we can distinguish several structures undergoing slightly different motions. The skull, for example, can move (quite rigidly) with respect to the brain, and the skin surface is subject to

large deformations (this is very clear with animated sequences spanning the one year study). The points corresponding to those structures have been rejected as outliers, but, with the help of the framework developed in this paper, we can think to classify and discriminate substructures. This is an improvement of our registration method that we are currently implementing.

Even for the brain itself, there are local deformations, for example due to the studied MS lesions, which are larger than the typical boundary error we measured, but because we are interested in an average motion for the whole structure of interest (here the brain), it makes sense to search for such an accurate motion. In fact, our registration is so accurate that it allows us to visualize, for the first time, the dynamic effect of the lesions on the surrounding brain tissues.

### *6.3. 3D substructure matching of proteins*

Most biological actions of proteins depend on some typical parts of their three-dimensional structure, called 3D *motifs*. To automatically discover corresponding 3D motifs between proteins, we proposed in (Pennec & Ayache, 1994) a new 3D substructure matching algorithm based on the geometric hashing technique. The key feature of the method is the introduction of a 3D reference frame attached to each amino acid (see figure 1 in section 2.2). This allows to compute for every couple of amino acids 6 invariants, that are used in the first step of the algorithm: a geometric hashing aimed to recognize possible correspondences between individual model and scene frames. The introduction of frame features instead of points drastically reduce the complexity of both the preprocessing and recognition stages, typically from $O(n^4)$ to $O(n^2)$. A clustering step based on motions performs then the aggregation of compatible matches. Individual motions within a cluster are fused using the scheme of section 2 in order to obtain a unique and accurate global motion. The last step is an alignment test realized as a kind of "iterative closest frame" (by analogy with ICP (Besl & McKay, 1992)), where our robust motion estimation scheme is embedded.

The noise on frame features is fixed on input and determines the type of common substructures the algorithm extracts. A small noise produces small but very accurate sets of matches, whereas a large noise favors more global substructure matches. For all our experiments, we use the atom coordinates of proteins provided by Brookhaven National Laboratory's Protein Data Bank (Bernstein *et al.*, 1977; Abola *et al.*, 1987). Visualization is done using the RasMol program of R. Sayle (Sayle & Bissel, 1992).

*Detection of the Helix-Turn-Helix motif* Structural motifs can be defined as the super-secondary structure. They are the simple combination of a few secondary structure elements. Some of them are associated with particular functions or are simple parts of larger structural and functional assemblies. For instance, the Helix-Turn-Helix motif is responsible for the binding of DNA within many procaryotic proteins. Some of them bind tightly to the DNA at a promoter of a gene, preventing RNA polymerase from fixing and hence blocking the initiation of the transcription. They are repressors. Conversely, activators bind next to the promoter and help polymerase to bind.

We choose to compare two proteins known to bind DNA: the tryptophan repressor of E. Coli (PDB code 2WRP (Lawson *et al.*, 1988), 105 amino-acids) and phage 434 CRO (PDB code 2CRO (Mondragon *et al.*, 1989), 65 amino-acids), whose Helix-Turn-Helix sequence were determined in (Brennan & Matthews, 1989; Harrison & Aggarwal, 1990).

Looking for a 3D binding motif, we used a quite small model of isotropic noise ($\sigma_\theta = 0.1$ rad $= 5$ deg and $\sigma_d = 0.35$ Å ). The algorithm ends up with only the correct matches from (15 MET - 66 MET) to (36 ALA - 87 ASN). The last match (37 GLY - 88 SER) is indeed very arguable considering the distance after registration and especially the difference in orientation. The typical object precision due to the registration (on the 22 matched amino-acids) is given to $\sigma_{obj} = 0.29$ Å. We show in figure 9 the two proteins and the registration found in figure 10. The only two other common substructures found score 13 and 8 matches and correspond to alpha helices, which are very stable secondary structure elements.

*Discussion* In order to test the stability of our algorithm, we also did the experiment with a noise two times larger ($\sigma_\theta = 10$ deg and $\sigma_d = 0.7$ Å ). We just find four additional matches preceding the beginning of the HTH motif ( from (7 LYS - 61 LEU) to (11 ILE - 64 GLY)). The two other clusters now score 14 and 9 matches. This shows that the detection of the HTH motif is very stable, and we argue that this is mainly due to the selectivity brought by the use of frames.

Indeed, the orientation of an amino acid is crucial to determine the position of collateral chains and most

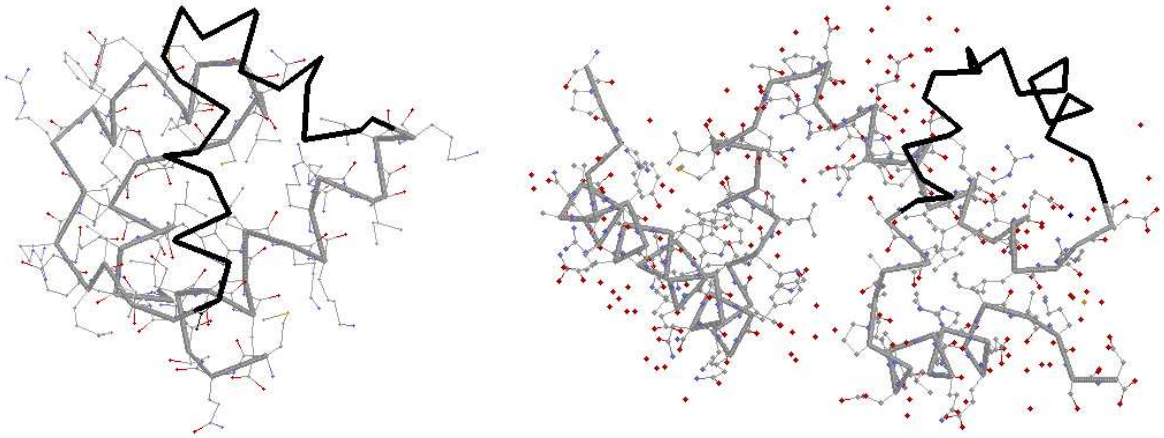

*Fig. 9.* The CRO protein (2CRO) of phage 434 on the left and the tryptophan repressor of E. Coli (2WRP) on the right. The matched part (the HTH motif) is displayed in black.

interactions of the protein happen within these side atoms. The position of these atoms is then not only determined by the position of the backbone but also its orientation and using just points to represent amino acids generally leads to a significant amount of additional matches with non compatible orientations. This implies a drastic reduction of selectivity for the matching process. In this case, frames bring rather more selectivity than more precision.

## 7. Conclusion

We developed a new formalism to handle uncertainty for rigid motion evaluation, which consider residual errors as residual geometric motions, and not as an additive noise applied to the measurements coordinates or to the parameters of the unknowns. This leads to new formulae for the equations of measurement which are used within the Extended Kalman Filter framework to evaluate both the motion and its covariance matrix, in the case of 3-D points or 3-D frames matches. We provide a quite general scheme to estimate approximately the accuracy of a registration method, and a more precise way to validate the predictions of our registration scheme. Several experiments on both synthetic and real data validate our framework and justify the compositive noise model. A practical result is to show that, in the case of 3D medical images, a precision of the registration far below the voxel size can be achieved. In the case of 3D protein substructure matching, frame features drastically improve the selectivity and reduce the complexity of the process.

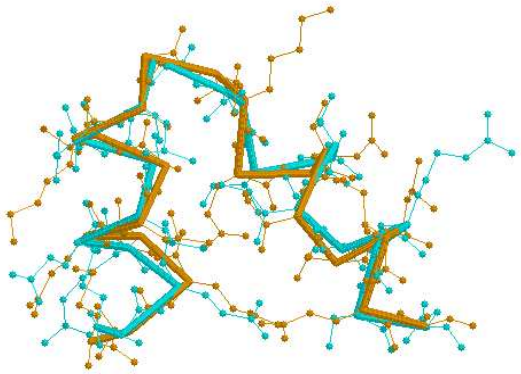

*Fig. 10.* Registration found between the HTH motifs of 2CRO and 2WRP. We can see that not only the backbone is very well matched, but also collateral chains are pretty well conserved.

This work also opens new theoretical questions about the noise on geometric features under a given type of motion, and offer numerous possibilities of extensions: other types of features, more general transformation groups, and also other high level algorithms based on this framework, such as clustering or matching.

## Appendix

### A.1. Rotations

#### *A.1.1. Geometric parameters: axis and angle*

Let $\mathbf{R}$ be a 3D rotation matrix ($\mathbf{R}\mathbf{R}^\top = \mathbf{I_d}$ and $\det(\mathbf{R}) = 1$). It is characterized by its axis $\mathbf{n}$ (unit vector) and its angle $\theta$. The relationship between these

two representations are given by Rodrigues formula (see for instance (Kanatani, 1993)).

$$\begin{aligned}\mathbf{R} &= \mathbf{I_d} + \sin\theta\,\mathbf{S_n} + (1-\cos\theta)\mathbf{S_n}^2 \\ &= \cos\theta\,\mathbf{I_d} + \sin\theta\,\mathbf{S_n} + (1-\cos\theta)\mathbf{n}.\mathbf{n}^\top\end{aligned} \tag{A1}$$

The matrix $\mathbf{S_n}$ is the skew matrix corresponding to (left) cross product: for all vector $\mathbf{v}$ we have $\mathbf{S_n}\mathbf{v} = \mathbf{n}\times\mathbf{v}$. If the coordinates of $\mathbf{n}$ are $(n_x, n_y, n_z)$, the matrix $\mathbf{S_n}$ is

$$\mathbf{S_n} = \begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix}$$

and we have the relation $\mathbf{S_n}^2 = \mathbf{n}.\mathbf{n}^\top - \mathbf{I_d}$, used to derive the second part of equation A1. Note that $\mathbf{S_n}$ uniquely determines the vector $\mathbf{n}$. Conversely, let $\mathrm{Tr}(\mathbf{R})$ be the trace of $\mathbf{R}$; the parameters are:

$$\begin{cases} \theta = \arccos\left(\dfrac{\mathrm{Tr}(\mathbf{R})-1}{2}\right) \\ \mathbf{S_n} = \dfrac{\mathbf{R}-\mathbf{R}^\top}{2\,\sin\theta} \end{cases} \tag{A2}$$

The last equation is valid only when $\theta \in ]0;\pi[$. Indeed, for $\theta = 0$ (i.e. identity) the rotation axis $\mathbf{n}$ is not determined, and $\sin(\theta) = 0$ for reflections ($\theta = \pi$).

**$\mathbf{R}$ *close to identity: $\theta$ is small*** Since the axis $\mathbf{n}$ is not defined for identity, there is a singularity and a numerical instability around it. However, we can compute the rotation vector with a Taylor expansion:

$$\begin{aligned}\mathbf{S_r} &= \theta\,\mathbf{S_n} = \tfrac{\theta}{2\sin\theta}(\mathbf{R}-\mathbf{R}^\top) \\ &= \tfrac{1}{2}\left(1+\tfrac{\theta^2}{6}\right)(\mathbf{R}-\mathbf{R}^\top) + O\left(\theta^4\right)\end{aligned}$$

**$\mathbf{R}$ *close to a reflection: $\pi-\theta$ is small*** The axis is this time well defined, but we have to use another equation. From Rodrigues formula, we get $\mathbf{R}+\mathbf{R}^\top - 2\,\mathbf{I_d} = 2(1-\cos\theta)\mathbf{S_n}^2$, and since $\mathbf{S_n}^2 = \mathbf{n}.\mathbf{n}^\top - \mathbf{I_d}$, we have

$$\mathbf{n}.\mathbf{n}^\top = \mathbf{I_d} + \frac{1}{2(1-\cos\theta)}\left(\mathbf{R}+\mathbf{R}^\top - 2\,\mathbf{I_d}\right)$$

Let $\varrho = (1-\cos\theta)^{-1}$; taking diagonal terms in the last equation gives $n_i^2 = 1 + \varrho.(R_{i,i}-1)$ and thus

$$n_i = \varepsilon_i\sqrt{1+\varrho.(R_{i,i}-1)}$$

The off diagonal terms are used to determine the signs $\varepsilon_i$: considering that the sign of $n_1$ is $\varepsilon \in \{-1;+1\}$, we can compute that

$$\mathrm{sign}(n_k) = \varepsilon.\mathrm{sign}(R_{1,k}+R_{k,1})$$

If we have an exact reflection, the sign $\varepsilon$ does not matter since rotating clockwise or counter-clockwise gives the same result, but for a quasi-reflection, this sign is important. In this case, the vector $\mathbf{w} = 2\sin\theta\,\mathbf{n}$ is very small but not identically null: it can be computed without numerical instabilities with $\mathbf{S_w} = \mathbf{R}-\mathbf{R}^\top$. Since $\theta < \pi$, the largest component $w_k$ in absolute value of this vector must have the same sign as the corresponding component $n_k$ of vector $\mathbf{n}$.

### *A.1.2. Rotation vector*

The representation of a rotation by its axis and angle values or by a unit quaternion are very useful for a lot of problems, but they are not minimal: there are quadratic constraints needed to enforce unitary vectors. This is a problem for most minimization techniques, and especially the extended Kalman filter (EKF).

A minimal representation is obtained with the *rotation vector* $\mathbf{r} = \theta\,\mathbf{n}$. Some problems of uniqueness are encountered around $\theta = \pi$ and four charts are theoretically needed to define an atlas (a set of differentiable maps covering the complete manifold). The rotation vector can be computed from the rotation matrix with the equations of section A.1.1. Conversely, Rodrigues formula allows us to compute the rotation matrix from the rotation vector:

$$\mathbf{R} = \mathbf{I_d} + \frac{\sin\theta}{\theta}\mathbf{S_r} + \frac{(1-\cos\theta)}{\theta^2}\mathbf{S_r}^2$$

with $\theta = \|\mathbf{r}\|$. In order to avoid numerical instabilities, we have to use a Taylor expansion for a small $\theta$:

$$\begin{aligned}\tfrac{\sin\theta}{\theta} &= 1 - \tfrac{1}{6}\theta^2 + O\left(\theta^4\right) \\ \tfrac{(1-\cos\theta)}{\theta^2} &= \tfrac{1}{2} - \tfrac{1}{24}\theta^2 + O\left(\theta^4\right)\end{aligned}$$

A more developed presentation of the rotation vector is given in (Ayache, 1991).

*Operations on rotation vectors* Let $\mathbf{r}$ be a generic rotation vector. The associated rotation matrix is denoted

by $\mathbf{R}$ and the parameters by $\theta = \|\mathbf{r}\|$ and $\mathbf{n} = \mathbf{r}/\|\mathbf{r}\|$). We are interested in this section in applying $\mathbf{r}$ to the vector $\mathbf{x}$: $\mathbf{y} = \mathbf{r} \star \mathbf{x}$, inverse it: $\mathbf{r}^{(-1)}$, and compose rotation vectors $\mathbf{r}_1$ and $\mathbf{r}_2$: $\mathbf{r} = \mathbf{r}_2 \circ \mathbf{r}_1$. The two first operations are easy to perform:

$$\mathbf{r} \star \mathbf{x} = \mathbf{R}.\mathbf{x} \qquad \text{and} \qquad \mathbf{r}^{(-1)} = -\mathbf{r}$$

The composition is more complex. Of course we can compute the associated rotation matrices, multiply them, ($\mathbf{R} = \mathbf{R}_2.\mathbf{R}_1$) and come back from $\mathbf{R}$ to the rotation vector $\mathbf{r}$, but it would be difficult to differentiate. We choose to use the unit quaternion representation as an intermediate step (see section (A.1.3) below). Let $\mathbf{r}_1$ and $\mathbf{r}_2$ be two rotation vectors. We compute the associated unit quaternions $\mathbf{q}_1$ and $\mathbf{q}_2$, and multiply them: $\mathbf{q} = \mathbf{q}_2 * \mathbf{q}_1$. Then we come back to the rotation vector $\mathbf{r}(\mathbf{q})$:

$$\mathbf{r}_2 \circ \mathbf{r}_1 = \mathbf{r}(\mathbf{q}(\mathbf{r}_2) * \mathbf{q}(\mathbf{r}_1))$$

### *A.1.3. Quaternions*

Quaternions are the elements of a 4 dimensional algebra on $\mathbb{R}$ which we shall note $\mathcal{Q}$. This is also the first non-commutative division ring (skew field) found by Hamilton in 1843. We can construct this algebra in several ways, but the we are interested in considers a quaternion $\mathbf{q} \in \mathcal{Q}$ as a couple $\mathbf{q} = (a, \mathbf{v})$, where $a \in \mathbb{R}$ is the real part and $\mathbf{v} \in \mathbb{R}^3$ the so-called pure part. The operations defined on quaternions to form the algebra are :

- Addition : $(a_1, \mathbf{v}_1) + (a_2, \mathbf{v}_2) = (a_1 + a_2, \mathbf{v}_1 + \mathbf{v}_2)$
- Internal multiplication : $(a_1, \mathbf{v}_1) * (a_2, \mathbf{v}_2) = (a_1 a_2 - < \mathbf{v}_1 | \mathbf{v}_2 >, \mathbf{v}_1 \times \mathbf{v}_2 + a_1 \mathbf{v}_2 + a_2 \mathbf{v}_1)$ where '$\times$' and '$< . | . >$' are the usual cross and dot products on $\mathbb{R}^3$

Moreover, we define the conjugate quaternion and the norm

- $\overline{(a, \mathbf{v})} = (a, -\mathbf{v})$
- $|\mathbf{q}|^2 = \|\mathbf{q}\|^2_{\mathcal{Q}} = \bar{\mathbf{q}} * \mathbf{q} = a^2 + \|\mathbf{v}\|^2_{\mathbb{R}^3} = \|\mathbf{q}\|^2_{\mathbb{R}^4}$

This allows to write very simply the inverse quaternion $\mathbf{q}^{-1} = \bar{\mathbf{q}}/|\mathbf{q}|^2$ for $\mathbf{q} \neq 0$. We shall note that the norm is compatible with the product: $|\mathbf{q}_1 * \mathbf{q}_2| = |\mathbf{q}_1| . |\mathbf{q}_2|$.

The set of quaternions $(0, \mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^3$ is trivially identified with $\mathbb{R}^3$ itself. Let $\mathbf{x}$ and $\mathbf{y}$ be two vectors (elements of $\mathbb{R}^3$). Their quaternion product is $\mathbf{x} * \mathbf{y} = (- < \mathbf{x} | \mathbf{y} >, \mathbf{x} \times \mathbf{y})$ and $\mathbf{q} * \mathbf{x} * \bar{\mathbf{q}}$ is a vector for any quaternion $\mathbf{q}$.

A more detailed introduction to quaternions and their properties is available in (Casteljau, 1987) and (Altmann, 1986).

*Quaternions and rotations* Let $\mathbf{q}$ be a unit quaternion. Then there exists $\theta \in [0, \pi]$ and $\mathbf{n}$ unit vector on $\mathbb{R}$ such that $\mathbf{q} = \left(\cos\left(\frac{\theta}{2}\right), \sin\left(\frac{\theta}{2}\right) \mathbf{n}\right)$. The map

$$\begin{array}{rccc} \mathbf{R}_{\mathbf{q}} : & \mathcal{Q} & \longrightarrow & \mathcal{Q} \\ & \mathbf{x} & \longmapsto & \mathbf{y} = \mathbf{q} * \mathbf{x} * \bar{\mathbf{q}} \end{array}$$

is an inner automorphism of $\mathcal{Q}$ that conserves pure quaternions (null real part). Its restriction to $\mathbb{R}^3$ is the vectorial rotation of $\mathbb{R}^3$ with angle $\theta$ around the unit vector $\mathbf{n}$. In a symmetric way, we can match to every rotation of $\mathbb{R}^3$ two unit quaternions $\mathbf{q}$ and $-\mathbf{q}$.

Let $\mathbf{R} \sim \mathbf{q}$ denote the association between rotation matrix $\mathbf{R}$ and rotation quaternion $\mathbf{q}$. As direct properties of this representation, we have:

- If $\mathbf{R}_1 \sim \mathbf{q}_1$ and $\mathbf{R}_2 \sim \mathbf{q}_2$ then $\mathbf{R}_1.\mathbf{R}_2 \sim \mathbf{q}_1 * \mathbf{q}_2$.
- If $\mathbf{R} \sim \mathbf{q}$ then $\mathbf{R}^{-1} \sim \bar{\mathbf{q}}$

By definition, the application of $\mathbf{R} \sim \mathbf{q}$ to vector $\mathbf{x}$ is

- $\mathbf{y} = \mathbf{R}\mathbf{x} = \mathbf{q} * \mathbf{x} * \bar{\mathbf{q}}$

*Conversions between unit quaternions and rotation vectors* Let $\mathbf{r} = \theta\, \mathbf{n}$ and $\mathbf{q} = \pm(a, \mathbf{v})$ be related to the same rotation $\mathbf{R}$. We have then $a = \cos(\theta/2)$ and $\mathbf{v} = \sin(\theta/2)\, \mathbf{n}$. The conversion from rotation vector to quaternion is then

$$\mathbf{q}(\mathbf{r}) = \left(\cos(\theta/2)\, ;\ \frac{\sin(\theta/2)}{\theta}\mathbf{r}\right)$$

with $\theta = \|\mathbf{r}\|$. Conversely, the rotation vector can be written (the notation $\overset{\|\mathbf{q}\|=1}{=\!=}$ means equal if $\|\mathbf{q}\| = 1$)

$$\begin{aligned} \mathbf{r}(\mathbf{q}) &= 2\, \mathrm{sign}(a)\ \arcsin\left(\frac{\|\mathbf{v}\|}{\sqrt{a^2 + \|\mathbf{v}\|^2}}\right) \frac{\mathbf{v}}{\|\mathbf{v}\|} \\ &\overset{\|\mathbf{q}\|=1}{=\!=}\ 2\, \mathrm{sign}(a)\ \frac{\arcsin(\|\mathbf{v}\|)}{\|\mathbf{v}\|}\mathbf{v} \end{aligned}$$

More details will be found with Jacobians of these operations.

*A.1.4. Jacobians*

We now have to differentiate the previous operations. These Jacobians (see section 2.4) are needed to propagate the covariance matrices, and in the linearization of the Extended Kalman Filtering equations. We recall first the differentiation of some usual operators for 3D vectors.

- **Norm and Normalization:**

$$\frac{\partial \|\mathbf{x}\|}{\partial \mathbf{x}} = \frac{\mathbf{x}^\top}{\|\mathbf{x}\|} \qquad \frac{\partial}{\partial \mathbf{x}}\left(\frac{\mathbf{x}}{\|\mathbf{x}\|}\right) = \frac{-{\mathbf{S}_\mathbf{x}}^2}{\|\mathbf{x}\|^3}$$

- **Dot product:**

$$\frac{\partial < \mathbf{x}|\mathbf{y} >}{\partial \mathbf{x}} = \mathbf{y}^\top \qquad \frac{\partial < \mathbf{x}|\mathbf{y} >}{\partial \mathbf{y}} = \mathbf{x}^\top$$

- **Cross Product:**

$$\frac{\partial \mathbf{x} \times \mathbf{y}}{\partial \mathbf{x}} = -\mathbf{S}_\mathbf{y} \qquad \frac{\partial \mathbf{x} \times \mathbf{y}}{\partial \mathbf{y}} = \mathbf{S}_\mathbf{x}$$

*Inversion of a rotation vector:* $\mathbf{r}^{(-1)} = -\mathbf{r}$ The differentiation of the inversion is very simple since:

$$\frac{\partial \mathbf{r}^{(-1)}}{\partial \mathbf{r}} = \frac{\partial(-\mathbf{r})}{\partial \mathbf{r}} = -\mathbf{I_d} \tag{A3}$$

*Application of a rotation vector:* $\mathbf{r} \star \mathbf{x} = \mathbf{R}.\mathbf{x}$ The problem is to compute the Jacobian $\frac{\partial(\mathbf{r}\star\mathbf{x})}{\partial \mathbf{r}}$. Let $\alpha, \beta, \gamma$ and $\delta$ be the following functions of $\theta$ with their Taylor expansions for a small $\theta$:

$$\alpha = \sin\theta/\theta = 1 - \tfrac{\theta^2}{6} + O\left(\theta^4\right)$$
$$\beta = (1-\cos\theta)/\theta^2 = \tfrac{1}{2} - \tfrac{\theta^2}{24} + O\left(\theta^4\right)$$
$$\gamma = \alpha'/\theta = (\cos\theta - \alpha)/\theta^2 = \tfrac{1}{3} - \tfrac{\theta^2}{30} + O\left(\theta^4\right)$$
$$\delta = \beta'/\theta = (\alpha - 2\beta)/\theta^2 = -\tfrac{1}{12} + \tfrac{\theta^2}{180} + O\left(\theta^4\right)$$

With Rodrigues formula, we can then write

$$\mathbf{r} \star \mathbf{x} = \mathbf{x} + \alpha.\mathbf{S}_\mathbf{r}.\mathbf{x} + \beta.{\mathbf{S}_\mathbf{r}}^2.\mathbf{x}$$

and taking into account the following derivatives

$$\frac{\partial(\mathbf{S}_\mathbf{r}.\mathbf{x})}{\partial \mathbf{r}} = -\mathbf{S}_\mathbf{x} \qquad \frac{\partial({\mathbf{S}_\mathbf{r}}^2.\mathbf{x})}{\partial \mathbf{r}} = \mathbf{S}_\mathbf{x}.\mathbf{S}_\mathbf{r} - 2\mathbf{S}_\mathbf{r}.\mathbf{S}_\mathbf{x}$$

and $\partial\theta/\partial\mathbf{r} = \mathbf{r}^\top/\theta$, we can differentiate it using the chain rule, which gives after factorizations

$$\begin{aligned}\frac{\partial(\mathbf{r}\star\mathbf{x})}{\partial \mathbf{r}} = & -\mathbf{S}_\mathbf{x}\left(\gamma.\mathbf{r}.\mathbf{r}^\top - \beta.\mathbf{S}_\mathbf{r} + \alpha.\,\mathbf{I_d}\right) \\ & -\mathbf{S}_\mathbf{r}.\mathbf{S}_\mathbf{x}\left(\delta.\mathbf{r}.\mathbf{r}^\top + 2\beta.\,\mathbf{I_d}\right)\end{aligned} \tag{A4}$$

*Link with other works:* Another way to differentiate the rotation application is given by N. Ayache in (Ayache, 1991): let $\eta$ be the function $\eta = (1-\alpha)/\theta^2$ ($\eta = \frac{1}{6} - \frac{\theta^2}{120} + O\left(\theta^4\right)$ for a small $\theta$), $\mathbf{U_r}$ the matrix

$$\mathbf{U_r} = \eta\,{\mathbf{S}_\mathbf{r}}^2 + \beta\,\mathbf{S}_\mathbf{r} + \mathbf{I_d}$$

and $\mathbf{du}$ the infinitesimal vector $\mathbf{du} = \mathbf{U_r}.\mathbf{dr}$. Then he showed that, for any small increment $\mathbf{dr}$ of the rotation vector $\mathbf{r}$, the increment $\mathbf{dR}$ of the rotation matrix $\mathbf{R}$ is given by $\mathbf{dR} = \mathbf{S_{du}}.\mathbf{R}$. To differentiate the application, we can then write

$$\begin{aligned}(\mathbf{R}+\mathbf{dR})\mathbf{x} - \mathbf{R}.\mathbf{x} = \mathbf{dR}.\mathbf{x} &= \mathbf{S_{du}}.\mathbf{R} \\ &= -\mathbf{S}_{(\mathbf{R}.\mathbf{x})}.\mathbf{du} = -\mathbf{S}_{(\mathbf{R}.\mathbf{x})}.\mathbf{U_r}.\mathbf{dr}\end{aligned}$$

and hence

$$\frac{\partial(\mathbf{r}\star\mathbf{x})}{\partial \mathbf{r}} = -\mathbf{S}_{(\mathbf{R}.\mathbf{x})}.\mathbf{U_r} \tag{A5}$$

which is only a factorized form of equation (A4). From a computational point of view, the first form is much faster, in particular if we note that

$$\mathbf{S_r}.\mathbf{S_x} = \mathbf{x}.\mathbf{r}^\top - < \mathbf{x}|\mathbf{r} > \mathbf{I_d} = \mathbf{x}.\mathbf{r}^\top - \mathrm{Tr}(\mathbf{x}.\mathbf{r}^\top)\,\mathbf{I_d}$$

*Composition of two rotation vectors* The most complex part of these computations is the following: how to compute the Jacobians of $\mathbf{r}_2 \circ \mathbf{r}_1$ with respect to the two rotation vectors? From $\mathbf{r} = \mathbf{r}_2 \circ \mathbf{r}_1 = \mathbf{r}(\mathbf{q}(\mathbf{r}_2) * \mathbf{q}(\mathbf{r}_1))$, we can easily derive with the chain rule

$$\frac{\partial \mathbf{r}}{\partial \mathbf{r}_1} = \frac{\partial \mathbf{r}}{\partial \mathbf{q}}.\frac{\partial \mathbf{q}}{\partial \mathbf{q}_1}.\frac{\partial \mathbf{q}_1}{\partial \mathbf{r}_1} \tag{A6}$$

The formula being symmetrical for the derivation with respect to $\mathbf{r_2}$. The problem is now to compute these intermediate Jacobians.

*From rotation vector* $\mathbf{r}$ *to unit quaternion* $\mathbf{q}$. Let $\mathbf{q} = (a, \mathbf{v})$ be one of the unit quaternion associated with

rotation vector $\mathbf{r}$, and $\theta = \|\mathbf{r}\|$. From $a = \cos(\theta/2)$ and $\mathbf{v} = \frac{\sin(\theta/2)}{\theta}\mathbf{r}$, we get

$$\frac{\partial a}{\partial \mathbf{r}} = -\frac{\sin(\theta/2)}{2}\frac{\mathbf{r}^\top}{\theta} = -\frac{\mathbf{v}^\top}{2}$$
$$\frac{\partial \mathbf{v}}{\partial \mathbf{r}} = \frac{cos(\theta/2)}{2\theta}\mathbf{r}.\mathbf{r}^\top - \frac{\sin(\theta/2)}{2\theta}\left(\frac{\mathbf{S_r}}{\theta}\right)^2$$

Since ${\mathbf{S_r}}^2 = \mathbf{r}.\mathbf{r}^\top - \theta^2\,\mathbf{I_d}$, and introducing the following functions of $\theta$:

$$\kappa = \frac{\sin(\theta/2)}{\theta} = \frac{1}{2} - \frac{\theta^2}{48} + O\left(\theta^4\right)$$
$$\lambda = \frac{\sin(\theta/2)}{\theta^3} - \frac{\cos(\theta/2)}{2\theta^2} = \frac{1}{24}\left(1 - \frac{\theta^2}{40}\right) + O\left(\theta^4\right)$$

we can summarize the results as follows

$$\mathbf{q} = \begin{vmatrix} \cos\left(\frac{\theta}{2}\right) \\ \\ \kappa.\mathbf{r} \end{vmatrix} \qquad \frac{\partial \mathbf{q}}{\partial \mathbf{r}} = \begin{bmatrix} -\frac{\mathbf{v}^\top}{2} \\ \\ \kappa\,\mathbf{I_d} - \lambda.\mathbf{r}.\mathbf{r}^\top \end{bmatrix} \quad \text{(A7)}$$

*From unit quaternion $q$ to rotation vector $r$.* Let $\mathbf{q} = (a, \mathbf{v})$ be a quaternion. The unit quaternion $\frac{\mathbf{q}}{\|\mathbf{q}\|}$ represent the rotation associated with rotation vector $\mathbf{r}$. We can get $\mathbf{r}$ from $\mathbf{q}$ with the following equation.

$$\begin{aligned}\mathbf{r}(\mathbf{q}) &= 2\text{sign}(a)\,\arcsin\left(\frac{\|\mathbf{v}\|}{\sqrt{a^2 + \|\mathbf{v}\|^2}}\right)\frac{\mathbf{v}}{\|\mathbf{v}\|} \\ &\underline{\underline{\|q\|=1}}\; 2\,\text{sign}(a)\,\frac{\arcsin(\|\mathbf{v}\|)}{\|\mathbf{v}\|}\mathbf{v}\end{aligned}$$

If $a$ is positive, we can derive that

$$\begin{aligned}\frac{\partial \mathbf{r}}{\partial a} &= \frac{-2\mathbf{v}}{a^2 + \|\mathbf{v}\|^2} \;\underline{\underline{\|\mathbf{q}\|=1}}\; -2\mathbf{v} \\ \frac{\partial \mathbf{r}}{\partial \mathbf{v}} &= \frac{2a}{a^2 + \|\mathbf{v}\|^2}\frac{\mathbf{v}.\mathbf{v}^\top}{\|\mathbf{v}\|^2} \\ &\quad -\frac{2}{\|\mathbf{v}\|}\arcsin\left(\frac{\|\mathbf{v}\|}{a^2 + \|\mathbf{v}\|^2}\right)\frac{{\mathbf{S_v}}^2}{\|\mathbf{v}\|^2} \\ &\underline{\underline{\|\mathbf{q}\|=1}}\; 2\,\frac{\arcsin(\|\mathbf{v}\|)}{\|\mathbf{v}\|}\,\mathbf{I_d} \\ &\quad +2\left(a - \frac{\arcsin(\|\mathbf{v}\|)}{\|\mathbf{v}\|}\right)\frac{\mathbf{v}.\mathbf{v}^\top}{\|\mathbf{v}\|^2}\end{aligned}$$

For $a$ negative, the above equations can be used for $\mathbf{q}' = -\mathbf{q}$ and thus

$$\begin{aligned}\frac{\partial \mathbf{r}}{\partial a} &= \frac{\partial \mathbf{r}}{\partial a'}.\frac{\partial a'}{\partial a} = -2\mathbf{v}'.(-1) = -2\mathbf{v} \\ \frac{\partial \mathbf{r}}{\partial \mathbf{v}} &= \frac{\partial \mathbf{r}}{\partial \mathbf{v}'}.\frac{\partial \mathbf{v}'}{\partial \mathbf{v}} \\ &= -2\,\frac{\arcsin(\|\mathbf{v}\|)}{\|\mathbf{v}\|}\,\mathbf{I_d} \\ &\quad +2\left(-a - \frac{\arcsin(\|\mathbf{v}\|)}{\|\mathbf{v}\|}\right)\frac{\mathbf{v}.\mathbf{v}^\top}{\|\mathbf{v}\|^2}\end{aligned}$$

To summarize the calculi for $\|\mathbf{q}\| = 1$, let $\mu = \|\mathbf{v}\|$ and $\tau$ and $\upsilon$ the following functions of $a$ and $\mu$

$$\begin{aligned}\tau &= 2\,\text{sign}(a)\frac{\arcsin(\mu)}{\mu} \\ &= 2\,\text{sign}(a)\left(1 + \frac{\mu^2}{6}\right) + O\left(\mu^4\right)\end{aligned}$$

$$\begin{aligned}\upsilon &= \frac{2a - \tau}{\mu^2} = 2\,\text{sign}(a)\frac{\mu\sqrt{1-\mu^2} - \arcsin(\mu)}{\mu^3} \\ &= -2\,\text{sign}(a)\left(\frac{2}{3} + \frac{\mu^2}{5}\right) + O\left(\mu^4\right)\end{aligned}$$

where sign is the sign function with $\text{sign}(0) = \pm 1$ indifferently. We have then

$$\begin{aligned}&\mathbf{r} = \tau.\mathbf{v} \\ &\frac{\partial \mathbf{r}}{\partial \mathbf{q}} = \frac{\partial \mathbf{r}}{\partial (a, \mathbf{v})} = \left[-2\mathbf{v}\;;\; \tau\,\mathbf{I_d} + \upsilon.\mathbf{v}.\mathbf{v}^\top\right]\end{aligned} \quad \text{(A8)}$$

*Composition of the two quaternions:* $\mathbf{q} = \mathbf{q}_2 * \mathbf{q}_1$ Let $\mathbf{q}_i = (a_i, \mathbf{v}_i)$. We have then

$$\mathbf{q} = \mathbf{q}_2 * \mathbf{q}_1 = \begin{vmatrix} a_1 a_2 - < \mathbf{v}_1 | \mathbf{v}_2 > \\ \\ \mathbf{v}_2 \times \mathbf{v}_1 + a_2\mathbf{v}_1 + a_1\mathbf{v}_2 \end{vmatrix}$$

The Jacobians can then be easily written

$$\frac{\partial \mathbf{q}}{\partial \mathbf{q}_1} = \frac{\partial(\mathbf{q}_2 * \mathbf{q}_1)}{\partial \mathbf{q}_1} = a_2\,\mathbf{I_d} + \begin{bmatrix} 0 & -\mathbf{v}_2^\top \\ \\ \mathbf{v}_2 & \mathbf{S}_{\mathbf{v}_2} \end{bmatrix} \quad \text{(A9)}$$

and (beware of the minus sign for the right-bottom block of the matrix due to the cross product):

$$\frac{\partial \mathbf{q}}{\partial \mathbf{q}_2} = \frac{\partial(\mathbf{q}_2 * \mathbf{q}_1)}{\partial \mathbf{q}_2} = a_1\,\mathbf{I_d} + \begin{bmatrix} 0 & -\mathbf{v}_1^\top \\ \\ \mathbf{v}_1 & -\mathbf{S}_{\mathbf{v}_1} \end{bmatrix} \quad \text{(A10)}$$

Computations concerning rotations are now completed: we can apply, inverse and compose rotation vectors and compute the Jacobians of these operations.

## Acknowledgements

We would like to thank Nicholas Ayache, Mike Brady and Zhengyou Zang for stimulating discussions about error handling, and also Dr Ron Kikinis from the B&W Hospital, Harward Medical School, Boston, who provided us with the MR images. Part of that research has been supported by the Esprit Basic Research Action VIVA, and also by a contract with Digital Equipment.

## Notes

1. If the covariance matrix (the noise process) is already provided with the feature, we can use it directly.
2. The absence of bias for the registration process can be verified on synthetic data.

## References

Abola, E.E., Bernstein, F.C., Bryant, S.H., Koetzle, T.F., & Weng, J. 1987. Protein Data Bank. *Pages 107–132 of:* Allen, F.H., Bergerhoff, G., & Sievers, R. (eds), *Crystallographic Databases - Information Contents, Software Systems, Scientific Applications*. Data Commission of the Int. Union of Crystallography, Bonn/Cambridge/Chester.

Altmann, S. L. 1986. *Rotations, Quaternions, and Double Groups*. Clarendon Press - Oxford.

Anderson, T.W. 1958. *An Introduction to Multivariate Statistical Analysis*. NY: J. Wiley.

Arun, K.S., Huang, T.S., & Blostein, S.D. 1987. Least-Squares Fitting of Two 3-D Point Sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(5), 698–700.

Ayache, N. 1991. *Artificial Vision for Mobile robots - Stereovision and Multisensor Perception*. MIT-Press.

Ayache, N., & Faugeras, O.D. 1986. HYPER : A new Approach for the Recognition and Positioning of Two-Dimensional Objects. *IEEE Trans. on Patern Analysis and Machine Intelligence*, **8**(1), 44–54.

Bard, Y. 1974. *Nonlinear Parameter Estimation*. Academic Press.

Bernstein, F.C., Koetzle, T.F., Williams, G.J.B., Meyer, E.F., Brice, M.D., Rodgers, J.R., Kennard, O., Shimanouchi, T., & Tasumi, M. 1977. The Protein Data Bank: A Computer-based Archival File for Macromolecular Structures. *J. of Mol. Bio.*, **112**, 535–542.

Besl, Paul, & McKay, Neil. 1992. A Method for Registration of 3−D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **14**(2), 239–256.

Brennan, R.G., & Matthews, B.W. 1989. The Helix-Turn-Helix DNA binding motif. *J. Biol. Chem.*, **264**, 286–290.

Casteljau, P. 1987. *Les quaternions*. Hermes.

Csurka, G., Zeller, C., Zhang, Z., & Faugeras, O. 1995 (June). *Characterizing the Uncertainty of the Fundamental Matrix*. Research Report 2560. INRIA.

Durrant-Whyte, H.F. 1988a. *Integration, Coordination and Control of Multi-Sensor Robot Systems*. Kluwer Academic Publishers.

Durrant-Whyte, H.F. 1988b. Uncertain Geometry in Robotics. *IEEE Journal of Robotics and Automation*, **4**(1), 23–31.

Fang, J., & Huang, T.S. 1984. Some Experiments on Estimating the 3-D Motion Parameters of a Rigid Body from Two Consecutive Image Frames. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6**(5), 545–554.

Feldmar, J., Ayache, N., & Betting, F. 1997. 3D-2D projective registration of free-form curves and surfaces. *Journ. of Computer Vision and Image Understanding*, **65**(3), 403–424.

Förstner, W. 1987. Reliability Analysis of Parameter Estimation in Linear Models with Applications to Mensuration Problems in Computer Vision. *CVGIP*, 273–310.

Gill, P.E., Murray, W., & Wright, M.H. 1981. *Practical optimization*. London and New York: Academic Press.

Grimson, W.E.L. 1992. *Object Recognition by Computer - The role of Geometric Constraints*. MIT Press.

Haralick, R.M., Joo, H., Lee, C.N., Zhuang, X., Vaidya, V.G., & Kim, M.B. 1989. Pose Estimation from Corresponding Point Data. *IEEE transaction on Systems, Man and Cybernetics*, **19**(6), 1426–1446.

Harrison, S.C., & Aggarwal, A.K. 1990. DNA recognition by proteins with the Helix-Turn-Helix motif. *Annu. Rev. Biochem.*, **59**, 933–969.

Horn, B.K.P. 1987. Closed Form Solutions of Absolute Orientation Using Unit Quaternions. *Journal of Optical Society of America*, **A-4**(4), 629–642.

Huber, P. 1981. *Robust Statistics*. New York: John Wiley.

Huttenlocher, D.P., & Ullman, S. 1987. Object Recognition using Alignment. *Pages 72–78 of: Proc. of ICCV*.

Jazwinsky, A.M. 1970. *Stochastic Processes and Filtering Theory*. Academic Press.

Kanatani, K. 1993. *Geometric Computation for Machine Vision*. The Oxford Engineering science series, no. 37. Clarendon Press - Oxford.

Kanatani, K. 1994. Analysis of 3-D Rotation Fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(5), 543–549.

Kendall, M.G., & Moran, P.A.P. 1963. *Geometrical probability*. Griffin's statistical monographs and courses, no. 10. Charlkes Griffin & Co. Ltd.

Koch. 1988. *Parameter Estimation and Hypothesis Testing in Linear Models*. Berlin: Springer.

Lamdan, Y., & Wolfson, H.J. 1988. Geometric Hashing : A General and Efficient Model-Based Recognition Scheme. *Pages 238–289 of: Proc. of Second ICCV*.

Lawson, C.L., Zhang, R.G., Schevitz, R.W., Otwinowski, Z., Joachimiak, A., & Siegler, P.B. 1988. Flexibility of the DNA-Binding Domains of TRP Repressor. *Proteins Struct., Funct., Genet.*, **3**, 18.

Meer, Peter, Mintz, Doron, Kim, Dong Yoon, & Rosenfeld, Azriel. 1991. Robust Regression Methods for Computer Vision: A Review. *International Journal of Computer Vision*, **6**(1), 59–70.

Mondragon, A., Wolberger, C., & Harrison, S.C. 1989. Structure of Phage 434 Cro Protein at 2.35 Angstroms Resolution. *J. of Mol. Bio.*, **205**, 179.

Pennec, X., & Ayache, N. 1994. An $\mathcal{O}(n^2)$ Algorithm for 3D Substructure Matching of Proteins. *In:* Califano, A., Rigoutsos, I., & Wolson, H.J. (eds), *Shape and Pattern Matching in Computational Biology*. Plenum Publishing.

Pennec, X., & Ayache, N. 1996 (June). Randomness and Geometric Features in Computer Vision. *In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'96)*.

Pennec, X., & Thirion, J.P. 1995 (June). Validation of 3D Registration Methods based on Points and Frames. *Pages 557–562 of: Proceedings of the 5th Int. Conf on Comp. Vision (ICCV'95)*.

Press, W.H., Flannery, B.P., Teukolsky, S.A, & Vetterling, W.T. 1991. *Numerical Recipices in C*. Cambridge Univ. Press.

Rigoutsos, I., & Hummel, R. 1993. Distributed Bayesian Object Recognition. *Pages 180–186 of: Proceedings of Int. Conf on Comput. Vis. and Pat. Recog*. IEEE Computer Society Press.

Sayle, R., & Bissel, A. 1992. RasMol: A program for fast realistic rendering of molecular structures with shadows. *In: Proceedings of the 10th Eurographics UK'92 Conference*.

Smith, R., & Cheeseman, P. 1987. On the representation and Estimation of Spatial Uncertainty. *Int. J. Robotics Research*, **5**(4).

Smith, R., Matthew, S., & Cheeseman, P. 1987. A Stochastic Map for Spatial Relationships. *In:* Bolles, R., & Roth, B. (eds), *4th Symposium on Robotics Research*.

Snyder, M.A. 1989. The Precision of 3-D Parameters in Correspondence-Based Techniques: The Case of Uniform Translational Motion in a Rigid Environment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(5), 523–528.

Thirion, J-P. 1994 (June). Extremal Points : definition and application to 3D image registration. *In: IEEE conf. on Computer Vision and Pattern Recognition*.

Thirion, J-P., & Gourdon, A. 1993 (March). *The 3D Marching Lines Algorithm : new results and proofs*. Tech. rept. 1881. INRIA. accepted for publication in CVGIP in august 1994.

Thirion, J-P, & Gourdon, A. 1995. Computing the Differential Characteristics of Isointensity Surfaces. *Computer Vision and Image Understanding*, **61**(2), 190–202.

Umeyama, S. 1991. Least-Squares Estimation of Transformation Parameters Between Two Point Patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(4), 376–380.

van den Elsen, Petra. 1993. Multimodality Matching of Brain Images. *Utrecht University Thesis, the Netherland*.

Wolfson, H.J. 1990. Model-Based Recognition by Geometric Hashing. *Pages 526–536 of:* Faugeras, O. (ed), *Proc. of 1st Europ. Conf. on Comput. Vision (ECCV 90)*. Springer-Verlag. Lecture Notes in Computer Science 427.

Zhang, Z. 1994. Iterative Point Matching for Registration of Free-Form Curves and Surfaces. *International Journal of Computer Vision*, **13**(2), 119–152. Also Research Report No.1658, INRIA Sophia-Antipolis, 1992.

Zhang, Z., & Faugeras, O. 1992. *3D Dynamic Scene Analysis: a stereo based approach*. Springer series in information science, vol. 27. Springer Verlag. Chap. 2: Uncertainty Manipulation and Parameter Estimation, pages 9–27.

Zhuang, X.H., & Huang, Y. 1994. Robust 3-D 3-D Pose Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **16**(8), 818–824.