



# Validation of 3-D Registration Methods based on Points and Frames

Xavier Pennec, Jean-Philippe Thirion

► **To cite this version:**

Xavier Pennec, Jean-Philippe Thirion. Validation of 3-D Registration Methods based on Points and Frames. RR-2470, INRIA. 1995. inria-00074205

**HAL Id: inria-00074205**

**<https://hal.inria.fr/inria-00074205>**

Submitted on 24 May 2006

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

***Validation of 3-D Registration Methods based  
on Points and Frames***

Xavier PENNEC, Jean-Philippe THIRION

**N° 2470**

January 1995

PROGRAMME 4



*Rapport  
de recherche*





# Validation of 3-D Registration Methods based on Points and Frames

Xavier PENNEC, Jean-Philippe THIRION

Programme 4 — Robotique, image et vision  
Projet Epidaure

Rapport de recherche n° 2470 — January 1995 — 42 pages

**Abstract:** In this paper, we propose a new method to estimate a rigid transform from a set of 3-D matched points or matched frames, and we concentrate on the analysis of the uncertainty of the estimated transform. 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 transform and its covariance matrix for points or frames correspondences, and also a statistical validation method to verify the error estimation, which applies even when no “ground truth” is available. The practical contribution is the validation of our transform estimation method in the case of 3-D medical images, which shows that a precision of the registration, far below the size of a voxel, can be achieved.

**Key-words:** Registration, Error handling, Rigid transform, Uncertainty, Precision, Geometry, Medical computer vision

*(Résumé : tsvp)*

# Validation des Méthodes de Recalage 3-D basées sur les Points et les Repères

**Résumé :** Nous proposons dans cet article une nouvelle méthode d'estimation de transformation rigide à partir d'appariements de points ou de repères, et nous nous concentrons plus particulièrement sur l'analyse de l'incertitude de la transformation estimée. Nous introduisons pour cela un modèle de bruit intrinsèque par composition et développons un formalisme unifié pour l'estimation simultanée de la transformation et de son incertitude. Nous proposons également une méthode de validation statistique pour vérifier l'estimation de l'erreur, avec ou sans vérité terrain. La contribution pratique est la validation de notre algorithme d'estimation de transformation rigide dans le cas des images médicales 3D, qui montre qu'un recalage d'une précision bien inférieure à la taille du voxel peut être obtenu.

**Mots-clé :** Incertitude, Gestion de l'Erreur, Transformations Rigides, Recalage Rigide 3D, Precision, Géométrie, Imagerie Médicale

## 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 transform. The aim of matching methods is generally to reduce the complexity of the feature association. See, for example, [AF86] or [HU87] for Alignment, [Gri92] for Interpretation Trees, [LW88, Wol90] or [RH93] for Geometric Hashing, [BM92] or [Zha94] for Iterative Closest Point (ICP) methods.

In the following, we do not discuss matching methods per se, rather the estimation of the transform. The traditional approach is to apply a least squares method, using for example the singular value decomposition (see [AHB87], [Ume91]), or the quaternion representation (see [Hor87]). Some authors, such as [ZH94], concentrate on the robustness of that estimation.

There are fewer studies, however, dealing with the precision of the estimated transform, which is our main concern here. Early experimental works can be found in [FH84], [Sny89], or [HJLZ89]. A theoretical evaluation of the associated errors is introduced in [Kan93, Kan94], mainly for the rotational component.

As those studies deal principally with point-to-point correspondences, one of our contributions is a method which applies also to frame-to-frame correspondences, that is, points with associated trihedrons. Also, a common assumption is that errors in rigid transforms can be modeled with an additive error bias on the parameters of the transform representation, or that translation and rotation errors can be separated. Due to the very peculiar nature of rigid transforms, namely the interaction between the rotation and translation components, we argue that rigid transforms should be considered together and that the assumption of additivity should be replaced by the composition with an error transform.

This idea leads us to the development of a formalism which unifies the handling of the error for the application, composition and evaluation of rigid transforms, in the case of point-to-point or frame-to-frame correspondences.

Section 2 describes this formalism, and shows how to estimate a rigid transform and its covariance matrix from a set of matched points or a set of matched

frames, and also how to fuse several rigid transform estimates. Then, it shows how those estimates can be used to sort matched primitives (points or frames) and reject outliers, that is, matched primitives which are “obviously” incompatible with the estimated transform.

Section 3 describes our transform 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 two processes: transform 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 transform, a covariance matrix, and the sorted set of matched primitives compatible with that transform.

Another contribution is presented in section 4. We describe an a posteriori method to validate statistically our transform estimation algorithm. This statistical validation applies even when no “ground truth” is available.

Finally, we present in section 5 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 for 3D medical images, for which the evaluation of the precision of the registration is vital.

## 2 Uncertainty and rigid transformations

### 2.1 Representation of rigid transformation and frames

Let  $\mathcal{B} = \{O, i, j, k\}$  be the canonical right-handed orthonormal basis of Euclidean space  $\mathbb{R}^3$  and  $\mathcal{F} = \{P, i', j', k'\}$  be another right-handed orthonormal basis (i.e. a frame). The rigid transformation from  $\mathcal{B}$  to  $\mathcal{F}$  can be written in homogeneous coordinates

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

The matrix  $M$  can also be viewed as the homogeneous matrix representing the coordinate change from  $\mathcal{F}$  to  $\mathcal{B}$ .

Let  $R$  be a rotation of angle  $\theta$  around the unit axis  $n$ . Using the rotation vector  $r = \theta n$  (see [Aya91] and appendix A), we can represent the rigid

transformation  $M$  by the six dimensional vector

$$f = \begin{vmatrix} r \\ t \end{vmatrix}$$

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

## 2.2 Operations on frames

Rigid transformations can be composed, inverted and applied to points. The application of a rigid transformation  $f$  to a frame  $f_m$  is simply the composition  $f_s = f \circ f_m$ . The compose (operator  $\circ$ ), inverse ( $^{(-1)}$ ) and apply ( $\star$ ) operations on rotation vectors (see appendix A), can be easily extended to frames in the following way.

- Application of  $f = (r, t)$  to point  $x$ :  $y = f \star x = r \star x + t$
- Composition of  $f_1 = (r_1, t_1)$  by  $f_2 = (r_2, t_2)$ :  $f = f_2 \circ f_1 = \begin{vmatrix} r_2 \circ r_1 \\ r_2 \star t_1 + t_2 \end{vmatrix}$
- Inverse of  $f = (r, t)$ :  $f^{(-1)} = \begin{vmatrix} r^{(-1)} \\ r^{(-1)} \star (-t) \end{vmatrix}$

## 2.3 Model of noise

Assume now that we work with estimated or measured values of frames. Then these values are corrupted by measurement errors. We model this stochastic process by the composition of the true value with an error: let  $f$  be the genuine representation of a frame and  $\hat{f}$  the measured one, then  $\hat{f} = f \circ e$  where  $e$  is a small rigid transformation around the identity. In this process,  $f$  is a deterministic frame, whereas  $e$  and  $\hat{f}$  are probabilistic. Due to the asymmetry of the composition operation, the right error presented above is more suitable



than the left error ( $\hat{f} = e_l \circ f$ ) because the latter cannot be assumed to be close to the identity (see appendix B.1).

This noise model is independent of the representation, and then intrinsic. It allows the comparison of errors between different frames and provides a uniform way of modeling uncertainty due to the measurement process: assuming we have a set of frames  $f_i$ , the measurement process being the same for all frames, we will measure  $\hat{f}_i = f_i \circ e_i$  with each error  $e_i$  following the same law. Last but not least, the error is itself a transformation and can then be easily understood. This is significantly different from the usual additive error model  $\hat{f} = f + \delta f$ , where  $\delta f$  has no meaning since the addition or subtraction of frames is not defined, and moreover completely depends on the representation.

## 2.4 Uncertainty representation

In order to deal efficiently with uncertainty from a computational point of view, we need to keep only a few number of parameters characterizing it. Since we are using a specific representation of frames, we choose to use the first and second order centered moments, i.e. the expectation vector and covariance matrix of the representation considered as a random vector. A similar approach is developed in [DW88a, DW88b, ZF92]. However, we note that it is not used here as a model of the noise process, but only as a convenient way of handling uncertainty.

Let  $\hat{f} = f \circ e$  be the representation of a probabilistic frame: it is a 6D random vector of probability density  $\rho$ . Its expectation is defined by  $\bar{f} = E(\hat{f}) = \int \hat{f} \rho(\hat{f}) d\hat{f}$ ; and its covariance matrix is given by

$$W_f = E \left( (\hat{f} - \bar{f})(\hat{f} - \bar{f})^\top \right)$$

The implicit assumptions here are that (i) the distribution (in the representation space) is symmetric with respect to the mean value, (ii) the first and second order moments exist, and (iii) higher moments are negligible with respect to them.

The probabilistic frame  $f$  will then be treated as the pair  $(\bar{f}, W_f)$ . A common hypothesis is that the noise is *without bias*, i.e.  $E(e) = Id$ , which means that  $\bar{f} = f$ . A probabilistic frame is in that case the pair  $(f, W_f)$  and a deterministic frame will have a null covariance matrix. We already know how to

operate on deterministic frames (section 2.2), so we can concentrate on how the covariance matrices propagate through the three main operations.

For that purpose, we will use the classical first order approximation of Jacobians. Let  $x$  be a vector with covariance  $W_x$  and  $h$  a multi-dimensional function acting on  $x$ . The covariance of the vector  $z = h(x)$  is, to first order

$$W_z = J_h W_x J_h^\top \quad \text{with} \quad J_h = \frac{\partial h(x)}{\partial x}$$

The propagation of covariance matrices through the operations of application, composition and inversion of frames are derived in appendix B.3.

## 2.5 Estimation of a rigid transformation

In this study, we face two main problems of estimation. The first is the estimation of a rigid transformation  $(f, W_f)$  from two sets of matched geometric primitives, here points or frames. This is the registration problem. The second is the merging of several measures  $(f_i, W_{f_i})$  of the same transformation in order to get a more precise estimation. This is a data fusion problem.

Many papers on pose estimation focus on closed-formed solutions in the case of point-to-point correspondence (see the references in the introduction and especially [Kan93]), but very few deal with the uncertainty estimation of the pose found. Moreover, we also want to estimate the rigid transform from frame-to-frame correspondences, and no classical method easily extends to such a problem.

However, a unified framework for estimation with uncertainty has been developed using Extended Kalman Filtering (EKF) in [Aya91]. We extend this framework to the estimation from frames correspondences in the following sections. A supplementary hypothesis about features for the EKF is that their distribution, in the parameter space, is approximately Gaussian.

### 2.5.1 Estimation from matched points

Assume we have two sets of matched points  $x_i$  and  $y_i$ , one transformed into the other with a rigid transformation  $f$  such that  $y_i = f \star x_i$ . During the measurement process, these points are corrupted by additive noise: we only measure

$\hat{x}_i = x_i + \delta x_i$  and  $\hat{y}_i = y_i + \delta y_i$  and we want to estimate the transformation  $f$  and the covariance  $W_f$  of this estimation. We assume that the errors  $\delta x_i$  and  $\delta y_i$  have covariance  $W_{x_i}$  and  $W_{y_i}$ .

Most minimization techniques, and in particular Extended Kalman Filtering (EKF), rely on the definition of an error vector  $z_i$  with covariance  $W_i$  that should ideally be null for each measure  $i$ , and try then to minimize the weighted least squares criterion

$$C = \sum_i z_i^\top W_i^{(-1)} z_i$$

The error in this case is the classical difference in position  $z_i = \hat{y}_i - f \star \hat{x}_i$ , and we can use an EKF to estimate  $(f, W_f)$ .

In the isotropic case (say when  $W_{x_i} = W_{y_i} = \sigma^2 Id$ ), the criterion reduces to a simple least squares

$$C = \sum_i \|\hat{y}_i - f \star \hat{x}_i\|^2$$

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 [Hor87, Aya91]). However, there is no covariance associated with the transformation found.

### 2.5.2 Estimation from matched frames

We now face the following problem. Assume we have two sets of matched frames  $\{f_{m_i}\}$  and  $\{f_{s_i}\}$  ( $m$  for model and  $s$  for scene), transformed one into another with a global rigid transformation  $f$  such as  $f_{s_i} = f \circ f_{m_i}$ . We only have access to their measured values  $\hat{f}_{s_i} = f_{s_i} \circ e_{s_i}$  and  $\hat{f}_{m_i} = f_{m_i} \circ e_{m_i}$  and we want to estimate the rigid transformation  $f$  and a covariance matrix  $W_f$  based of that estimate. We assume moreover that we have some knowledge about the measurement process:  $e_{m_i} = (Id, W_{m_i})$  and  $e_{s_i} = (Id, W_{s_i})$ .

Combining the two measure equations with the hypothesis and isolating error, we get

$$\hat{f}_{s_i}^{(-1)} \circ f \circ \hat{f}_{m_i} = e_{s_i}^{(-1)} \circ e_{m_i} \quad (1)$$

The left side of this equation is our error vector  $e_i = (Id, W_i)$  to be minimized in the criterion  $C = \sum_i e_i^\top W_i^{(-1)} e_i$ .

We are using here 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. The computation needed to linearize the measure equation  $e_i = \hat{f}_{s_i}^{(-1)} \circ f \circ \hat{f}_{m_i}$  are given in appendix C.

### 2.5.3 Fusion of rigid transformations

Consider now a set  $\{\hat{f}_i\}$  of measures of the same frame  $f$  such as  $\hat{f}_i = f \circ e_i$ . The goal is to estimate the frame  $(f, W_f)$ . We assume an error  $e_i = (Id, W_i)$  on each measure. The measure equation is

$$\hat{f}_i^{(-1)} \circ f = e_i^{(-1)} \quad (2)$$

and the corresponding weighted least squares criterion can also be minimized using the EKF (appendix C).

## 2.6 Rejecting outliers: Mahalanobis distance and $\chi^2$ test

Another interesting problem that arises within this framework is compatibility. For a given transformation, 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 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 the quality of the estimation.

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

Such problems are traditionally tackled via Mahalanobis distances for sorting hypotheses and  $\chi^2$  tests for rejecting outliers.

### 2.6.1 Two points and a transformation

Let  $(x, W_x)$  and  $(y, W_y)$  be two measured points and  $(f, W_f)$  a rigid transformation. We want to test the validity of the hypothesis  $y = f \star x$ .

We saw in section 2.4 how to compute  $(f \star x, W_{f \star x})$ , then the covariance on the error vector  $z = y - f \star x$  should be zero with covariance  $W = W_y + W_{f \star x}$ . A  $\chi^2$  test is well suited to verify if  $z$  is compatible with its theoretical covariance. The squared Mahalanobis distance  $\mu^2$  is

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

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 can be found in table 1.

For example, if  $\mu^2 > 11.34$ , there is less than 1% of probability that  $y$  be a measure of  $f \star x$ . We can thus reject this match.

### 2.6.2 Two frames and a transformation

Let  $(f_m, W_m)$  and  $(f_s, W_s)$  be two measured frames and  $(f, W)$  a rigid transformation. We want to test the hypothesis that  $f_s = f \circ f_m$ . From section 2.4 we can compute the covariance  $W$  on the error vector  $e = f_s^{(-1)} \circ f \circ f_m$  defined above and use the squared Mahalanobis distance  $\mu^2 = e^\top W^{(-1)} e$  and a 6-D  $\chi^2$  test to decide if the hypothesis is true or not (see table 1).

### 2.6.3 Two frames

Let  $(f_1, W_1)$  and  $(f_2, W_2)$  be two measured frames. We want to test the hypothesis that they are measures of the same frame  $f$ . Let  $W$  be the covariance on the error vector  $e = f_1^{(-1)} \circ f_2$ . We can once again use the squared Mahalanobis distance  $\mu^2 = e^\top W^{(-1)} e$  and a 6-D  $\chi^2$ .

$\alpha$	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

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

## 2.7 Conclusion on uncertainty and rigid transformations

We have developed in this section a theoretical framework for modeling uncertainty and for handling frames and rigid transformations. We propose a new model of noise based on composition and show why it is better than the classical additive noise. The use of a covariance matrix on the parameter vector is then a powerful tool for handling uncertainty in computations. Estimation problems within this framework require the computation of a covariance matrix of the estimate: the Extended Kalman Filter provides a uniform 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 transformations.

This framework can be used in a large number of vision problems. For example, we present in the next section an application to registration; but we could also use this framework for matching algorithms (see [PA94]). Further applications include the clustering of rigid transformations between image structures, or rigid motions in a displacement field, in order to detect and isolate objects (see also [SB]).

## 3 A practical and Robust algorithm for transform estimation and accuracy evaluation

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 transform, 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 transformation between them. Assuming all the features are corrupted by the same stochastic process, we compute statistics (i.e. the covariance matrix of the process) on features, and use this information to recompute the rigid transformation and its associated covariance. This bootstrap process can be continued until convergence. This will then allow us to predict a variance on the position of each registered point of the image.

### 3.1 The bootstrapping process

The considered features are again points and frames, but initially without any quality information. 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  $f$ , using point positions only.
- 2/ Estimate a global covariance matrix on features.
- 3/ Order matches by increasing Mahalanobis distances.

We now have enough information on features to start the bootstrapping process. Step 1 can be ignored if a reliable transformation is provided, whereas step 3 can be 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 can be order-dependent. If the differences are usually not significant, some special cases can lead to quantitative errors.

Ordering also helps outliers rejection for the transformation computation while keeping these matches for other purposes. In protein substructure matching, for instance in [PA94], we want to compute the transformation 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 bootstrap process itself is as follows.

- 1/ Estimate the transformation  $(f, W_f)$  from the matches using an EKF (in decreasing significance order) until the Mahalanobis distance of the matches becomes larger than a predetermined  $\varepsilon$  value (this is the  $\chi^2$  test to reject outliers).
- 2/ Estimate the covariance matrix of all features<sup>1</sup>.
- 3/ Sort matches by their Mahalanobis distance. Outliers are at the end of the list.

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

## 3.2 Estimation of the covariance matrix of features

### 3.2.1 The case of point features

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

The error vector  $z_i = \hat{y}_i - \hat{f} \star \hat{x}_i$  is, to first order,  $z_i = \delta y_i - r \star \delta x_i$  if we consider the error in the estimate of  $f$  to be small compared with the error on points (which should be the case if we have enough matched features to estimate it).

Assuming that each measurement error  $\delta x_i$  and  $\delta y_i$  comes from the same (isotropic) process, with covariance  $W$ , the covariance on the error vector  $z_i$  should be  $2W$ . We can then estimate  $W$  by

$$W_{x_i} = W_{y_i} = W = \frac{1}{2} E(z_i z_i^\top) = \frac{1}{2N} \sum_i z_i z_i^\top$$

It should be noted that neglecting the error on the transformation leads to an overestimation of  $W$ , which is an acceptable condition.

---

<sup>1</sup>Computing the covariance matrix on inliers leads to an under-estimation because only small errors are taken into account



### 3.2.2 The case of frame features

With the same notations as in the previous sections, and neglecting once again the error on the estimate  $\hat{f}$  of the transformation  $f$ , the error is:

$$e_i = \hat{f}_{s_i}^{(-1)} \circ \hat{f} \circ \hat{f}_{m_i} = e_{s_i}^{(-1)} \circ e_{m_i}$$

Assuming a common underlying process of measurement errors  $e_{s_i} = e_{m_i} = (Id, W)$ , we can determine that  $e_{s_i}^{(-1)} = (Id, W)$ , hence  $e_i = (Id, 2W)$ . It should be noted that the covariances behave as here usual, but only because errors are around identity. An estimator of the covariance  $W$  of frame features is given by

$$W_{m_i} = W_{s_i} = W = \frac{1}{2} E(e_i e_i^\top) = \frac{1}{2N} \sum_i e_i e_i^\top$$

### 3.3 Final Precision evaluation

From  $(\hat{f}, W_f)$ , we can compute for each point  $x$  of the model image (or object) its transform  $\hat{y} = \hat{f} \star x$  and the uncertainty  $W_y = W_{\hat{f} \star x}$ . In order to give a more intuitive information, we compute instead the RMS error expectation at this point: if  $y = f \star x$  is the real location of the transformed point  $x$ , we shall find it in  $\hat{y} = \hat{f} \star x$ . The expectation of the squared distance  $d^2 = \|y - \hat{y}\|^2$  between them is

$$\sigma_d^2(y) = E \left( (\hat{y} - y)(\hat{y} - y)^\top \right) = \text{trace} \left( E \left( (\hat{y} - y)^\top (\hat{y} - y) \right) \right) = \text{trace}(W_y)$$

In the case of 3-D medical images, and in order to characterize the precision with a single value, we can estimate the averaged RMS for the 8 vertices of the image. We will call this value the *typical boundary error*.

### 3.4 Conclusion on registration and accuracy

From two sets of matched features, and one parameter  $\varepsilon$ , we propose a method to estimate the transformation  $f$  and its precision  $W_f$ , discard outliers, and provide an estimation of the precision of inlier features. We extract then from the transformation precision a single value characterizing the registration accuracy.

## 4 A statistical validation of registration methods

We now propose a method to estimate the precision 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 features have no bias.

### 4.1 Ground truth

We believe that no method can give an objective reference for the transform. Simply, some methods are more reliable than others: when a method A is one order of magnitude more precise than a method B, A can serve as a reference for an experimental validation of B. But all methods are ultimately statistical, even methods using *external markers*.

### 4.2 External markers

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. Other authors have concluded experimentally that, for the case of high resolution medical images, marker-based techniques are probably much less precise than image-based techniques (see [VdE93]).

In fact, external markers have a very strong, *psychological* advantage over other validation methods, because we can physically touch them: they are more tractable than an abstract covariance matrix. We have to keep in mind, however, that the measures of marker positions are corrupted, which implies the elaboration of a theoretical model of the markers, and the application of a statistical method to evaluate the transform and its uncertainty.

Hence, external marker techniques are no less statistical than other methods, and the results of the present paper apply directly to them also (generally, markers are points or frames).

### 4.3 Analytical methods

Ideally, we might think of modeling analytically the whole process, from the physical object positions to the final estimated transform. 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.

### 4.4 An a posteriori statistical estimation of the errors

The method that we present now applies when there is no ground truth for the transform, and when the analytical determination of the errors is impossible (the majority of cases). We consider the registration method as a black box (figure 1), which takes two representations of the same object as input, and gives an estimated rigid transform as output.

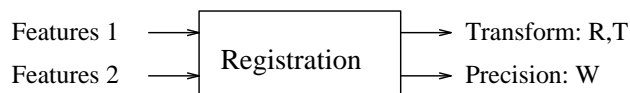


Figure 1: Registration considered as a black box

Assuming independence of pairs of matched features, we can subsample the matches in order to get several independent estimates of the researched transformation, and compare them. In particular, we can split the set of matches in two sets and obtain two estimates of  $f$ :

$$f^1 = f \circ e^1 \quad f^2 = f \circ e^2$$

The two transformations should be very close, and we can study their “distance”  $e = (f^2)^{(-1)} \circ f^1 = (e^2)^{(-1)} \circ e^1$  which does not depend of the exact transform  $f$  and should be close to identity.

If matches are randomly subsampled to a fixed number  $M$  of matches, in order to conserve independence and an similar distribution of matches in space, then the estimation errors  $e^1$  and  $e^2$  follow the same law of covariance  $W(M)$ . If moreover both the feature extraction and the registration method have no bias<sup>2</sup> (this means that  $E(e^1) = E(e^2) = Id$ ), then we can derive that the measured error  $e$  should have a covariance  $W_e(M) = 2W(M)$ .

Repeating the experiment with several homogeneous sets of matches, we can estimate the covariance matrix of the estimation of  $f$  from  $M$  feature matches by

$$W(M) = \frac{1}{2n} \sum_{i=1}^n e_i e_i^\top$$

## 4.5 Validation of the covariance estimation

This method is also used to validate our estimation scheme. We compute the two estimates  $f^1$  and  $f^2$ , along with their covariance matrices, and then the error  $e$  and its *predicted* uncertainty  $W_{e_{pred}}$ . The best way to see if the measured values of  $e$  are compatible with their predicted uncertainty is to use the Mahalanobis distance of  $(e, W_{e_{pred}})$  with identity.

$$\mu^2 = e^\top W_{e_{pred}}^{(-1)} e$$

Requirements about the number of features used to compute  $f^1$  and  $f^2$  can then be removed. If the covariance estimation is perfect, then the expected value of  $\mu^2$  is 6, the dimension of the transformation space. We use then several sets of matches and estimate the expectation by the average value:

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

and characterize the adequation of the *predicted* uncertainty of the estimated transform with the *a posteriori* measured values by the *validation index*

$$I_2 = \sqrt{\frac{\bar{\mu}^2}{6}}$$

---

<sup>2</sup>The absence of noise for the registration process can be verified on synthetic data.

This index gives indications on how the estimation method under-estimates ( $I_2 > 1$ ) or over-estimates ( $I_2 < 1$ ) the error on the estimated transform. It is a kind of relative error on the error estimation.

In the case of synthetic data with a known transformation  $f$ , a direct validation index  $I_1$  can also be computed from the error

$$e = f^{(-1)} \circ f'$$

since by definition the transform estimation is  $f' = f \circ e$ .

These two validation indexes are used in the Experiments section to validate our estimation scheme, both with synthetic and with real data.

## 4.6 Conclusion on statistical validation

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

# 5 Experiments

We present now the validation of our transform estimation algorithm, first for simulated data, and then for real 3D medical image registration.

## 5.1 Synthetic data experiments

We have generated a large number of lists of associated frames, with a Gaussian error distribution on the positions and on the orientations, and without any false matches. Each registration involves 500 points, and we have averaged the results of a 100 registration experiments. To be as close as possible to the real case, the frames were randomly chosen in the same volume as in the real case (see section 5.2) and the covariance matrix used to perturbate features was measured from the real experiments (the variances are approximately  $\sigma_d = 0.41 \text{ mm}$  on position and  $\sigma_\theta = 0.08 \text{ rad}$  on orientation). We use in both real and synthetic experiments a  $\chi^2$  value of 12 to remove outliers.

	Corner RMS	$I_1$	$I_2$
Points	0.120 mm	1.175	1.18
Frames	0.107 mm	0.966	1.025

Table 2: Statistical validation, synthetic case

We present in table 2 the average typical boundary error (average RMS on the corners of the image over all the experiments) and the validation indexes  $I_1$  and  $I_2$  for the point and frame algorithms. Remember that these indexes are relative errors on the error prediction. This shows a very good fit of the algorithm prediction for frames and a slight under-estimation for points.

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

### 5.1.1 Comparison between points and frame

Since frames are composed of points and trihedrons, we can compare easily the precision of the point to point registration with respect to the frame to frame one. We choose to run the experiment with 75 points, a fixed variance of  $\sigma_\theta = 0.02rad$  on the orientation, and a variance increasing from  $\sigma_d = 0.4mm$  to  $2.4mm$  on point position. The resulting precision (or typical boundary error) is drawn in figure 2. Using frames in spite of points can then lead to a important improvement in precision. However, this is more sensible when the number of matches decreases, and/or with an increasing ratio  $\sigma_d/\sigma_\theta$ .

## 5.2 Real data experiments

We present results from an experiment performed using 3D Magnetic Resonance images (MRI). The images come 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 get a complete 3D MR examination several times during one year (typically 24 different 3D acquisitions). The aim is to register very precisely, and in 3D, all those images, in order to segment the

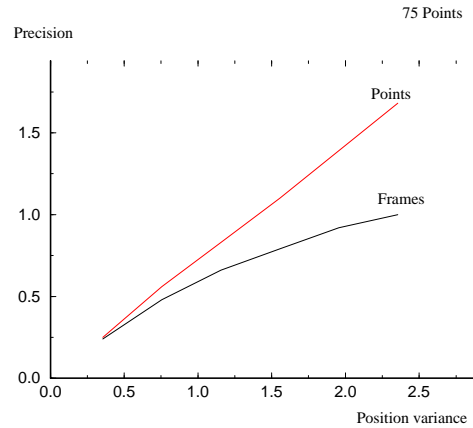


Figure 2: Comparison of the estimation precision using points and frames.

lesions and evaluate very accurately their evolution (MS lesions are white spots in the images in figure 4).

The images are first echo,  $256 \times 256 \times 54$  voxels, the voxel size is  $1mm \times 1mm \times 3mm$ . We have already presented an algorithm to perform, fully automatically, the registration of the images in [Thi94]. However, there is a need to estimate precisely the errors in those registrations. The registered images are resampled using a tri-linear interpolation method. The same slices of several different 3D MRI are presented in figure 4, after resampling with the estimated transforms. Figure 5 presents the differences between images, to show visually the quality of the registration, and also that the evolutions of the lesions are clearly detected.

### 5.3 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 3). In our case, these are the *Extremal Points*, as defined in [Thi93], 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 measures associated with those points (the principal curvatures), which are used to reduce the complexity of the matching process, but also that we get the principal directions, which form, with the normal to the surface and the extremal point itself, an orthonormal basis, that is, a frame.

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  $1\text{mm}$ . With a density less than 0.1% of extremal points in the image, the probability of false matches is very low.

## 5.4 Results

With 24 different images, we could have generated 576 pairs of images. Since each registration, including the extraction of extremal points, takes about 4 minutes CPU for an DEC alpha workstation, we have to date restricted our experiments to 12 registrations. We find a typical boundary RMS error of  $\sigma = 0.13\text{ mm}$  and a typical predicted covariance on corners of

$$\bar{W}_y = \begin{bmatrix} 0.00496 & -0.00002 & 0.00011 \\ -0.00002 & 0.00473 & -0.00009 \\ 0.00011 & -0.00009 & 0.00707 \end{bmatrix}$$

The validation index is  $I_2 = 1.094$ , which validates our algorithm for real data.

The other interesting result is that the average error  $\sigma$  for the 3-D image boundary ( $0.13\text{ mm}$ ) is far below the resolution, which is ( $1\text{ mm} \times 1\text{ mm} \times 3\text{ mm}$ ). The typical predicted covariance on corners shows moreover that the uncertainty of the registration in the  $X$  and  $Y$  dimensions is similar (256 voxels of 1 mm in each dimension reduced to 128 of 2 mm for feature extraction), but about 1.5 times inferior to the uncertainty in the  $Z$  axis, where we only have 56 voxels of 3 mm. This proves that covariance matrices are needed to correctly handle uncertainty when images are not isotropic.

## 5.5 Discussion

With that level of accuracy, we can no more consider the elements in the image as undergoing the same transform, nor that this transform is rigid.



Our matching algorithm tries to estimate the motion of a single rigid sub-structure, with the largest number of common extremal points, which is the brain in those images. The skull, for example, undergoes a different transform, with far fewer points matched, 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 our evaluation algorithm, we could also have classified and discriminated several different structures, undergoing different rigid transforms. 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 far larger than the typical boundary error  $\sigma$  we have measured, but because we are interested in an average transform for the whole structure of interest (here the brain), it makes sense to search for a transform that precise. In fact, our registration is so precise that it allows us to visualize, for the first time, the dynamic effect of the lesions on the surrounding brain tissues.

At that point, however, we do not think that the registration is to be improved, but rather adapted to the case of locally deformable registration.

## 6 Conclusion

We have developed a new formalism to handle uncertainty for rigid transform evaluation, which consider simultaneously rotation and translation, and which consider residual errors as transforms, and not as an additive noise. This leads to new formulae for the equations of measures which are used within the Extended Kalman Filter framework to evaluate both the transform 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 precision of a registration method, and a more precise way to validate the predictions of our registration scheme. 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

## 7 Acknowledgments

We want to thank Nicholas Ayache and Mike Brady for stimulating discussions about error handling, and also Dr Ron Kikinis from the B&W Hospital, Harvard 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.

## References

- [AF86] N. Ayache and O.D. Faugeras. Hyper : A new approach for the recognition and positioning of two-dimensionnal objects. *IEEE Trans. on Patern Analysis and Machine Intelligence*, 8(1):44–54, 1986.
- [AHB87] K.S. Arun, T.S. Huang, and S.D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):698–700, September 1987.
- [Aya91] N. Ayache. *Artificial Vision for Mobile robots - Stereo-vision and Multisensor Perception*. MIT-Press, 1991.
- [BM92] Paul Besl and Neil McKay. A method for registration of 3–D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, February 1992.
- [Cas87] P. Casteljaou. *Les quaternions*. Hermes, 1987.
- [DW88a] H.F. Durrant-Whyte. *Integration, Coordination and Control of Multi-Sensor Robot System*. Kluwer Academic Publishers, 1988.
- [DW88b] H.F. Durrant-Whyte. Uncertain geometry in robotics. *IEEE Journal of Robotics and Automation*, 4(1):23–31, February 1988.
- [FH84] J. Fang and T.S. Huang. 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, September 1984.

- [Gri92] W.E.L. Grimson. *Object Recognition by Computer - The role of Geometric Constraints*. MIT Press, 1992.
- [HJLZ89] R.M. Haralick, H. Joo, C.N. Lee, X. Zhuang, V.G. Vaidya, and M.B. Kim. Pose estimation from corresponding point data. *IEEE transaction on Systems, Man and Cybernetics*, 19(6):1426–1446, November, December 1989.
- [Hor87] B.K.P. Horn. Closed form solutions of absolute orientation using unit quaternions. *Journal of Optical Society of America*, A-4(4):629–642, April 1987.
- [HU87] D.P. Huttenlocher and S. Ullman. Object recognition using alignment. In *Proc. of ICCV*, pages 72–78, 1987.
- [Jaz70] A.M. Jazwinsky70. *Stochastic Processes and Filtering Theory*. Academic Press, 1970.
- [Kan93] K. Kanatani. *Geometric Computation for Machine Vision. The Oxford Engineering science series*, Clarendon Press - Oxford, 1993.
- [Kan94] K. Kanatani. Analysis of 3-d rotation fitting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):543–549, May 1994.
- [LB87] M. Le Borgne. *Quaternions et Controle sur l'Espace des rotations*. Research Report 751, INRIA, November 1987.
- [LW88] Y. Lamdan and H.J. Wolfson. Geometric hashing : A general and efficient model-based recognition scheme. In *Proc. of Second ICCV*, pages 238–289, 1988.
- [PA94] X. Pennec and N. Ayache. An  $o(n^2)$  algorithm for 3d substructure matching of proteins. In *Proceedings of the first International Workshop on Shape and Pattern Matching in Computational Biology*, JUNE 1994.

- [RH93] I. Rigoutsos and R. Hummel. Distributed bayesian object recognition. In *Proceedings of Int. Conf on Comput. Vis. and Pat. Recog*, pages 180–186, IEEE Computer Society Press, June 1993.
- [SB] L.S. Shapiro and M. Brady. Motion and structure clustering using maximum affinity spanning trees. Submitted to ICCV95.
- [Sny89] M.A. Snyder. 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, May 1989.
- [Thi93] J-P Thirion. *New Feature Points based on Geometric Invariants for 3D Image Registration*. Technical Report 1901, INRIA, April 1993. accepted for publication in IJCV.
- [Thi94] J-P Thirion. Extremal points : definition and application to 3d image registration. In *IEEE conf. on Computer Vision and Pattern Recognition*, Seattle, June 1994.
- [Ume91] S. Umeyama. Least-squares estimation of transformation parameters between two point patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(4):376–380, April 1991.
- [VdE93] Petra Van den Elsen. Multimodality matching of brain images. *Utrecht University Thesis, the Netherland*, (ISBN 90-71546-02-0), 1993.
- [Wol90] H.J. Wolfson. Model-based recognition by geometric hashing. In O. Faugeras, editor, *Proc. of 1st Europ. Conf. on Comput. Vision (ECCV 90)*, pages 526–536, Springer-Verlag, April 1990. Lecture Note in Computer Science 427.
- [ZF92] Zhang Z. and O. Faugeras. *3D Dynamic Scene Analysis: a stereo based approach*, chapter 2: Uncertainty Manipulation and Parameter Estimation, pages 9–27. Volume 27 of *Springer series in information science*, Springer Verlag, 1992.

- [ZH94] X.H. Zhuang and Y. Huang. Robust 3-d 3-d pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(8):818–824, August 1994.
- [Zha94] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *the International Journal of Computer Vision*, 13(2):119–152, 1994. Also Research Report No.1658, INRIA Sophia-Antipolis, 1992.

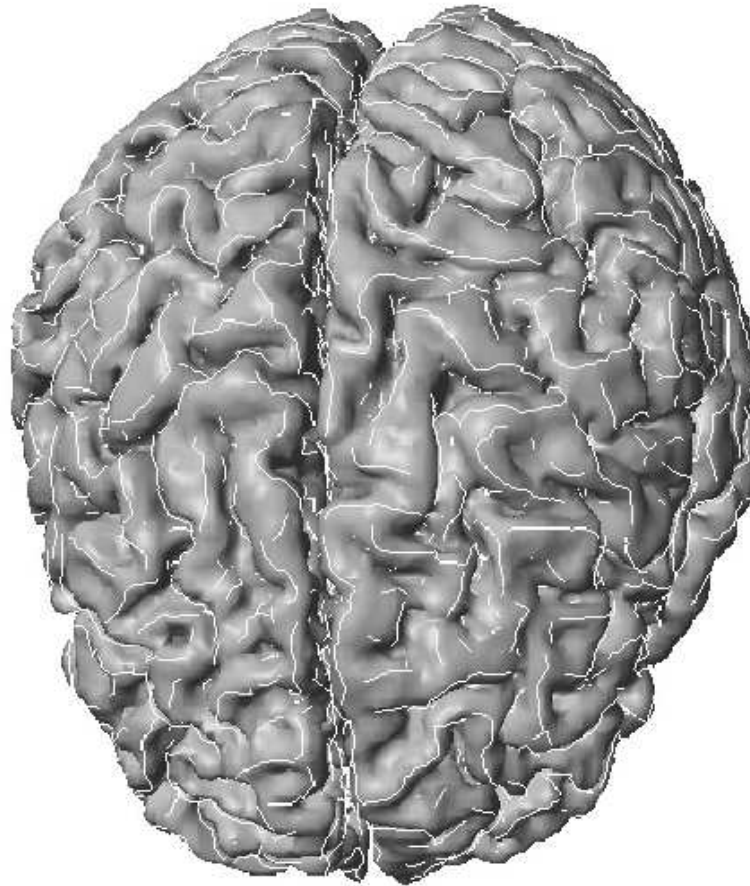


Figure 3: Lines of extremal curvature on the brain circumvolutions. The extremal points, which are used in the registration, are specific points of those lines

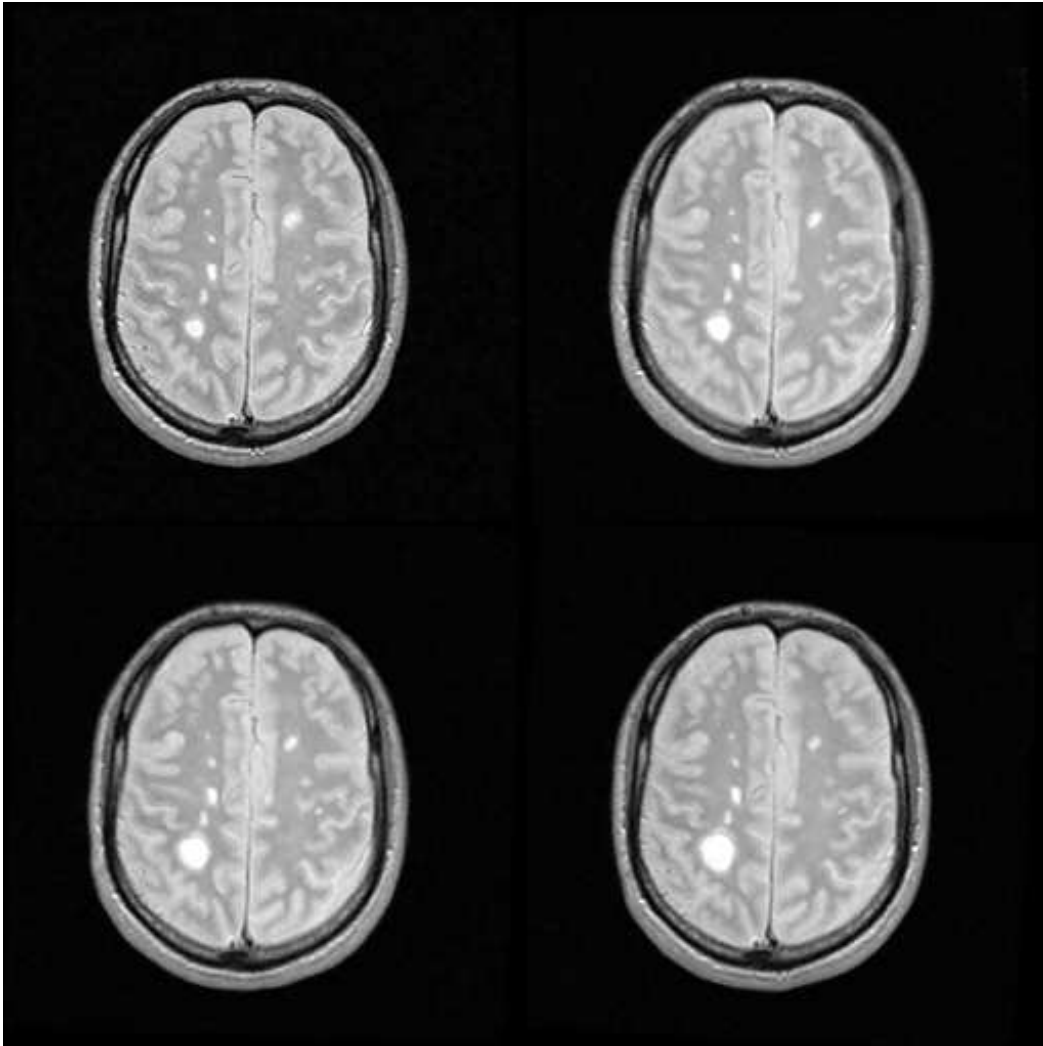


Figure 4: The same slices from four registered 3D MR images of the Multiple Sclerosis study. Note the evolution of two MS lesions (white spots). One is growing in the anterior left hemisphere, one is shrinking in the right posterior hemisphere. There is two weeks between each acquisition.

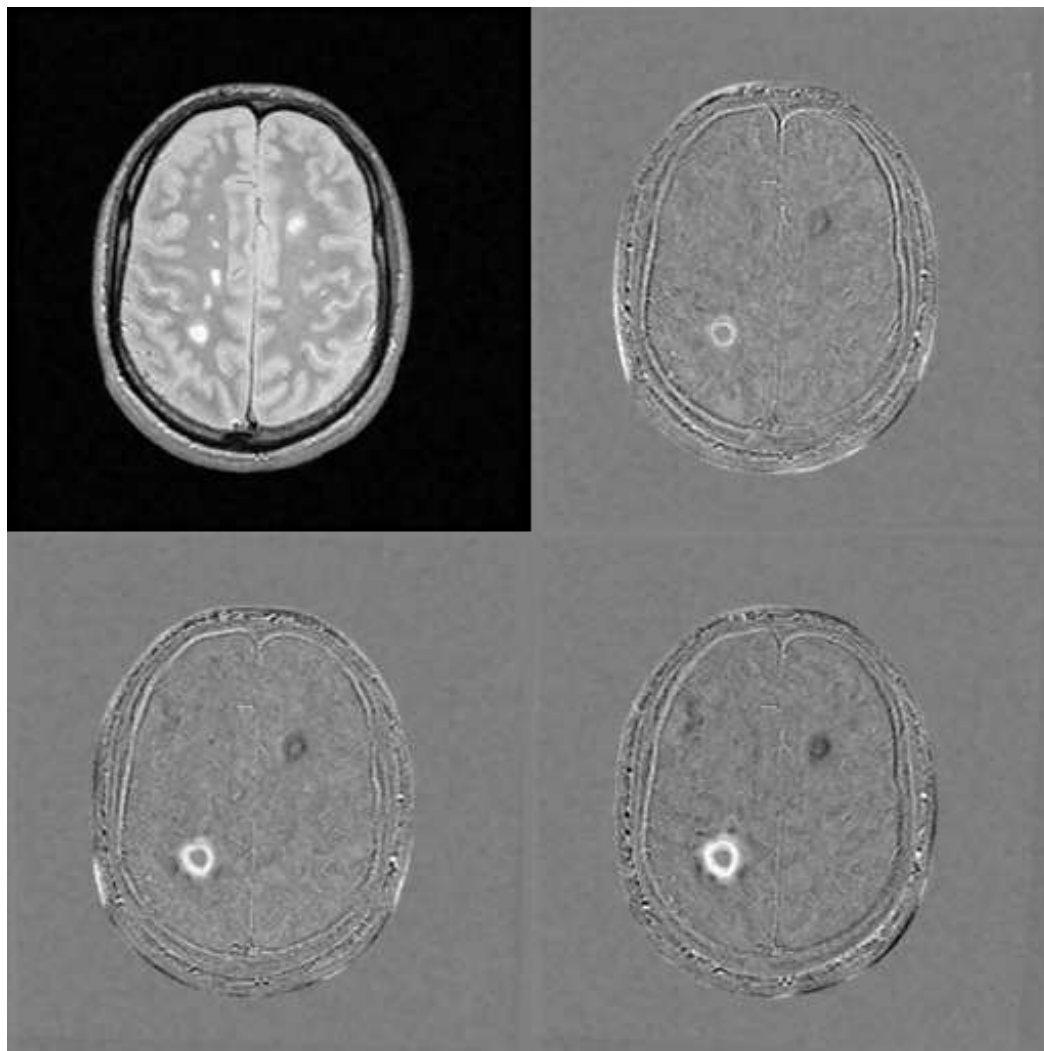


Figure 5: Differences between images after registration, with respect to the first one. The intensity is multiplied by 5, and shifted such that no difference is a gray value. The growing lesion appear as a white disk, and the shrinking lesion as black a disk.



## A Rotations

### A.1 Geometric parameters

Let  $R$  be a rotation matrix ( $RR^\top = I_d$  and  $\det(R) = 1$ ). It is characterized by its axis  $n$  (unit vector) and its angle  $\theta$ . The relationship between these two representations are given by Rodrigues formula.

$$R = I + \sin \theta \tilde{n} + (1 - \cos \theta) \tilde{n}^2 \quad (3)$$

The symbol  $\sim$  denotes the operation which maps to a vector  $n$  the skew matrix  $\tilde{n}$  such as for all vector  $v$  we have  $\tilde{n}v = n \times v$ . If the coordinates of  $n$  are  $(n_x, n_y, n_z)$ , the matrix  $\tilde{n}$  is

$$\tilde{n} = \begin{pmatrix} 0 & -n_z & n_y \\ n_z & 0 & -n_x \\ -n_y & n_x & 0 \end{pmatrix}$$

Note that  $\tilde{n}$  uniquely determines the vector  $n$ . Conversely, let  $Tr(R)$  be the trace of  $R$ :

$$\theta = \arccos \left( \frac{Tr(R) - 1}{2} \right) \quad \text{and} \quad \tilde{n} = \frac{R - R^\top}{2 \sin \theta} \quad (4)$$

This formula is valid only when  $\theta \in ]0; \pi[$ . Indeed,  $n$  is not determined for identity ( $tr(R) = 3$ ) and  $R = I_3 + 2\tilde{n}^2 = 2nn^\top - I_3$  for reflections ( $tr(R) = -1$ ). Assuming  $n = (x, y, z)^\top$ , we have in this case  $2nn^\top = R - I_3 = [2xn, 2yn, 2zn]$ . It is then sufficient to normalize the largest column vector of  $R - I_3$  to obtain the axis  $n$ . It shall be noted that  $n$  and  $-n$  are both valid.

### A.2 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 one we will be interested in considers a quaternion  $q \in \mathcal{Q}$  as a pair  $q = (a, v)$ , where  $a \in \mathbb{R}$  is the real part and  $v \in \mathbb{R}^3$  the so-called pure part. The operations defined on quaternions to form the algebra are :

- Addition :  $(a_1, v_1) + (a_2, v_2) = (a_1 + a_2, v_1 + v_2)$
- Internal multiplication :  $(a_1, v_1) * (a_2, v_2) = (a_1 a_2 - \langle v_1 | v_2 \rangle, v_1 \times v_2 + a_1 v_2 + a_2 v_1)$  where ‘ $\times$ ’ and ‘ $\langle \cdot | \cdot \rangle$ ’ are the usual cross and dot products on  $\mathbb{R}^3$

Moreover, we define the conjugate quaternion and the norm

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

This allows to write very simply the inverse quaternion

$$q^{-1} = \frac{\bar{q}}{|q|^2}$$

for  $q \neq 0$ . We shall note that the norm is compatible with the product:  $|q_1 q_2| = |q_1| \cdot |q_2|$ .

The set of quaternions  $(0, x)$  with  $x \in \mathbb{R}^3$  is trivially identified with  $\mathbb{R}^3$  itself. Let  $x$  and  $y$  be two vectors (elements of  $\mathbb{R}^3$ ). Their quaternion product is  $x * y = (-\langle x | y \rangle, x \times y)$  and their cross product is then  $x \times y = \frac{1}{2}[x, y]$ . Let now  $q$  be a quaternion. Then  $qx\bar{q}$  is a vector.

A more detailed introduction to quaternions and their properties is available in [Cas87] and [LB87].

### A.2.1 Quaternions and rotations

Let  $q$  be a unit quaternion. Then there exists  $\theta \in [0, \pi]$  and  $n$  unit vector on  $\mathbb{R}^3$  such that  $q = \left(\cos\left(\frac{\theta}{2}\right), \sin\left(\frac{\theta}{2}\right) n\right)$ . The map

$$R_q : \mathcal{Q} \longrightarrow \mathcal{Q} \\ x \longmapsto y = q * x * \bar{q}$$

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  $n$ . In a symmetric way, we can match to every rotation of  $\mathbb{R}^3$  two unit quaternions  $q$  and  $-q$ .

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

- If  $R_1 \leftrightarrow q_1$  and  $R_2 \leftrightarrow q_2$  then  $R_1.R_2 \leftrightarrow q_1 * q_2$ .
- If  $R \leftrightarrow q$  then  $R^{-1} \leftrightarrow \bar{q}$

and by definition, the application of to vector  $x$  is

- $y = Rx = q * x * \bar{q}$

### A.3 Rotation vector

The representation of a rotation by its axis and angle values or by a unit quaternion are very useful and easy for a lot of problems, but they are not minimal: there is unit constraints. This is a problem for the EKF.

A minimal representation is however obtained with the *rotation vector*  $r = \theta n$ . Some problems of uniqueness are encountered around  $\theta = \pi$  and four charts are theoretically needed to define an atlas. The rotation vector can be computed from the rotation matrix with the formulas above. A more developed presentation of the rotation vector and the correct way to compute it is given in [Aya91]

### A.4 Conversions between unit quaternions and rotation vectors

Let  $r = \theta n$  and  $q = \pm(\alpha, \beta)$  be related to the same rotation  $R$ . We have then  $\alpha = \cos(\theta/2)$  and  $\beta = \sin(\theta/2) n$ . The formulas are then

$$q(r) = \left( \alpha = \cos\left(\frac{\|r\|}{2}\right), \beta = \sin\left(\frac{\|r\|}{2}\right) \frac{r}{\|r\|} \right)$$

$$r(q) = 2 \operatorname{sgn}(\alpha) \arccos(\alpha) \frac{\beta}{\|\beta\|}$$

with the convention that the normalization  $\frac{x}{\|x\|} = 0$  if  $x = 0$  and where  $\operatorname{sgn}$  is the sign function.

## A.5 Operations on rotation vectors

Let  $r$  be a generic rotation vector. The associated rotation matrix will be denoted by  $R$  and the parameters by  $\theta = \|r\|$  and  $n = r/\|r\|$ . We are interested in this section in apply  $r$  to the vector  $x$ :  $y = r \star x$ , inverse it:  $r^{(-1)}$ , and compose rotation vectors  $r_1$  and  $r_2$ :  $r = r_2 \circ r_1$ . The two first operations are easy to do:

$$r \star x = R.x$$

$$r^{(-1)} = -r$$

The composition is more complex. Of course we can compute the associated rotation matrices, multiply them, ( $R = R_2.R_1$ ) and come back from  $R$  to the rotation vector  $r$ , but it would be difficult to differentiate. We choose to use the unit quaternion representation as intermediate step. Let  $r_1$  and  $r_2$  be two rotation vectors. We compute the associated unit quaternions  $q_1$  and  $q_2$ , and multiply them:  $q = q_2 * q_1$ . Then we came back to the rotation vector  $r(q)$ .

$$r_2 \circ r_1 = r(q(r_2) * q(r_1))$$

## A.6 Jacobians

We now have to differentiate the three previous operations. These Jacobians (see section B.2) are needed to propagate the covariance matrices, and in the linearization of the Extended Kalman Filtering equations.

The differentiation of the application  $\frac{\partial r \star x}{\partial r}$  of a rotation vector to a point is presented in [Aya91]. We do not repeat it here. The differentiation of the inversion is very simple

$$\frac{\partial r^{(-1)}}{\partial r} = \frac{\partial(-r)}{\partial r} = -Id$$

The Jacobians of the composition of  $r_2$  and  $r_1$  with respect to one of the two variable is quite more complex. We recall first the differentiation of some usual operators for 3D vectors.

- **Norm and Normalization:**  $\frac{\partial \|x\|}{\partial x} = \frac{x^\top}{\|x\|}$   $\frac{\partial \left(\frac{x}{\|x\|}\right)}{\partial x} = \frac{-\tilde{x}^2}{\|x\|^3}$

- **Dot product:**  $\frac{\partial \langle a|b \rangle}{\partial a} = b^\top$      $\frac{\partial \langle a|b \rangle}{\partial b} = a^\top$      $\frac{\partial \langle a|b \rangle}{\partial (a,b)} = [b^\top; a^\top]$
- **Cross Product:**  $\frac{\partial a \times b}{\partial a} = -\tilde{b}$      $\frac{\partial a \times b}{\partial b} = \tilde{a}$      $\frac{\partial a \times b}{\partial (a,b)} = [-\tilde{b}; \tilde{a}]$

### A.6.1 From rotation vector $r_i$ to unit quaternion: $q_i$

Let  $r_1$  and  $r_2$  be denoted generically by  $r_i$ ,  $q_i = (\alpha_i, \beta_i)$  being the associated quaternion. We have then

$$\alpha_i = \cos\left(\frac{\|r_i\|}{2}\right) \quad \Longrightarrow \quad \frac{\partial \alpha_i}{\partial r_i} = -\frac{1}{2} \sin\left(\frac{\|r_i\|}{2}\right) \frac{r_i^\top}{\|r_i\|} = -\frac{1}{2} \beta_i^\top$$

$$\beta_i = \sin\left(\frac{\|r_i\|}{2}\right) \frac{r_i}{\|r_i\|} \quad \Longrightarrow \quad \frac{\partial \beta_i}{\partial r_i} = \frac{\alpha_i}{2} \left( \frac{r_i \cdot r_i^\top}{\|r_i\|^2} \right) - \frac{\|\beta_i\|}{\|r_i\|} \left( \frac{\tilde{r}_i}{\|r_i\|} \right)^2$$

If the rotation vector  $r_i$  is identity (null vector), then we cannot normalize by  $\|r_i\|$ . It is easy to compute that in this case  $\alpha_i = 1$ ,  $\beta_i = 0$  and then  $\frac{\partial \alpha_i}{\partial r_i} = 0$  and  $\frac{\partial \beta_i}{\partial r_i} = \frac{1}{2} Id$ .

### A.6.2 Composition of the two quaternions: $q = q_2 * q_1$

Let now  $q = (\alpha, \beta)$  be the composition of  $q_1$  by  $q_2$ :  $q = q_2 * q_1$ . We have then

$$\alpha = \alpha_1 \alpha_2 - \langle \beta_1 | \beta_2 \rangle \quad \text{and} \quad \beta = \beta_2 \times \beta_1 + \alpha_2 \beta_1 + \alpha_1 \beta_2$$

If  $i$  is the index for one of the two rotation vectors, let  $j$  be the other. We find that

$$\frac{\partial \alpha}{\partial r_i} = -\frac{\alpha_j}{2} \beta_i^\top - \beta_j^\top \frac{\partial \beta_i}{\partial r_i}$$

Since the cross product introduces a difference between the two indexes, let  $\varepsilon$  be +1 for  $i = 1$  and -1 for  $i = 2$ . We get

$$\frac{\partial \beta}{\partial r_i} = (\varepsilon \tilde{\beta}_j + \alpha_j Id) \frac{\partial \beta_i}{\partial r_i} - \frac{\beta_j \cdot \beta_i^\top}{2}$$

### A.6.3 From unit quaternion $q$ to rotation vector $r$

$$r = 2 \operatorname{sgn}(\alpha) \arccos(\alpha) \frac{\beta}{\|\beta\|}$$

$$\frac{\partial r}{\partial r_i} = -\frac{2 \operatorname{sgn}(\alpha)}{\|\beta\|} \left( \arccos(\alpha) \left( \frac{\tilde{\beta}}{\|\beta\|} \right)^2 \frac{\partial \beta}{\partial r_i} + \frac{\beta}{\|\beta\|} \frac{\partial \alpha}{\partial r_i} \right)$$

If  $q$  is identity ( $\alpha = 1$  and  $\beta = 0$ ), we cannot normalize by  $\|\beta\|$ , but we can compute that in this case  $r = 0$  and  $\frac{\partial r}{\partial r_i} = 2 \frac{\partial \beta}{\partial r_i}$ .

### A.6.4 Practical computation

The Jacobians are computed using above formulas along with the computation of the composition of the two rotation vectors. For the singularities around identity, we fix a small threshold on the norm of the concerned vector above which we use the identity related Jacobian.

## A.7 Conclusion on rotations

We have now completed the computations related to rotation vectors. We can apply, inverse and compose rotation vectors, and compute the Jacobians of these operations. If we were only concerned with rotations, we could use these formulas to propagate a covariance matrix through the three operations, but we are now going to embed the rotations in rigid transformations.

## B Rigid transformations

### B.1 Left and Right error

Let  $f = (r, t)$  be a frame and  $\hat{f} = (\hat{r}, \hat{t})$  its estimation. We define the right error  $e_r$  and the left error  $e_l$  as follows.

$$\hat{f} = f \circ e_r = e_l \circ f$$

from these formula and the operations on frames (section 2.2), we can find that

$$e_r = \begin{vmatrix} r^{(-1)} \circ \hat{r} \\ r^{(-1)} \star (\hat{t} - t) \end{vmatrix} \quad \text{and} \quad e_l = \begin{vmatrix} \hat{r} \circ r^{(-1)} \\ \hat{t} - (\hat{r} \circ r^{(-1)}) \star t \end{vmatrix}$$

A desirable behavior for the error is that it extend the classical error on the two major components: translation and rotation, at least in norm. Along the two “axes”  $\hat{r} = r$  and  $\hat{t} = t$  the left and right errors become

$$\begin{aligned} \hat{r} = r & : e_d = \begin{vmatrix} Id \\ r^{(-1)} \star (\hat{t} - t) \end{vmatrix} & e_l = \begin{vmatrix} Id \\ \hat{t} - t \end{vmatrix} \\ \hat{t} = t & : e_d = \begin{vmatrix} r^{(-1)} \circ \hat{r} \\ 0 \end{vmatrix} & e_l = \begin{vmatrix} \hat{r} \circ r^{(-1)} \\ (\hat{r} \circ r^{(-1)}) \star t - t \end{vmatrix} \end{aligned}$$

Along the axis  $\hat{r} = r$  (no rotation error), the left error perfectly conserves the translation error whereas, the right error just conserves its norm. On the contrary, along the  $\hat{t} = t$  axis (no translation error), a translation component appears in the left error, which can quite huge if the translation  $t$  is large.

This asymmetry of the left and right error (due to the asymmetry of the composition operation) leads us to preferably choose the right error which can be assumed to be close to identity.

## B.2 Jacobians and covariance matrices

Let  $x = (x_1, \dots, x_m)^\top$  be a vector with covariance  $W_x$  and  $h = (h_1, \dots, h_n)^\top$  a multi-dimensional function acting on  $x$ . The covariance of the vector  $z = h(x)$  is, to first order,

$$W_z = J_h W_x J_h^\top \quad \text{with} \quad J_h = \frac{\partial h(x)}{\partial x} = \begin{bmatrix} \frac{\partial h_1}{\partial x_1} & \dots & \frac{\partial h_1}{\partial x_m} \\ \dots & \dots & \dots \\ \frac{\partial h_n}{\partial x_1} & \dots & \frac{\partial h_n}{\partial x_m} \end{bmatrix}$$

For a function of two independent parameters  $x$  and  $y$ , the covariance matrix on  $z = h(x, y)$  is simply

$$W_z = J_{h_x} W_x J_{h_x}^\top + J_{h_y} W_y J_{h_y}^\top \quad \text{with} \quad J_{h_x} = \frac{\partial h(x, y)}{\partial x} \quad \text{and} \quad J_{h_y} = \frac{\partial h(x, y)}{\partial y}$$

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  $h(x) = (h_1 \circ h_2)(x) = h_1(h_2(x))$ . The Jacobian of  $h$  is then

$$J_h \Big|_x = \frac{\partial h}{\partial x} = J_{h_1} \Big|_{h_2(x)} J_{h_2} \Big|_x$$

### B.3 Propagation of covariance matrices

- **Application of a frame  $(f, W_f)$  to point  $(x, W_x)$ :**

Let  $J_\star$  be the Jacobian of  $y = f \star x$  with respect to  $f = (r, t)$  and  $R$  the rotation matrix associated to the rotation vector  $r$ .

$$J_\star = \frac{\partial(r \star x + t)}{\partial(r, t)} = \left[ \frac{\partial(r \star x)}{\partial r}; \frac{\partial t}{\partial t} \right] = \left[ \frac{\partial(r \star x)}{\partial r}; I_3 \right]$$

$$J_x = \frac{\partial(r \star x + t)}{\partial x} = R$$

The derivation of a rotation  $r \star x = Rx$  with respect to the rotation vector  $r$  is given in [Aya91]. We obtain

$$W_y = J_\star W_f J_\star^\top + R W_x R^\top \quad \text{with} \quad J_\star = \left[ \frac{\partial(r \star x)}{\partial r}; I_3 \right]$$

- **Composition of frame  $(f_1, W_1)$  by  $(f_2, W_2)$ :**

Let  $J_1$  be the Jacobian of  $f = f_2 \circ f_1$  w.r.t  $f_1 = (r_1, t_1)$  and  $J_2$  w.r.t.  $f_2 = (r_2, t_2)$ . Then

$$J_1 = \frac{\partial(r_2 \circ r_1, r_2 \star t_1 + t_2)}{\partial(r_1, t_1)} = \left[ \begin{array}{cc} \frac{\partial(r_2 \circ r_1)}{\partial r_1} & 0 \\ 0 & R_2 \end{array} \right]$$

$$J_2 = \frac{\partial(r_2 \circ r_1, r_2 \star t_1 + t_2)}{\partial(r_2, t_2)} = \left[ \begin{array}{cc} \frac{\partial(r_2 \circ r_1)}{\partial r_2} & 0 \\ \frac{\partial(r_2 \star t_1)}{\partial r_2} & I_3 \end{array} \right]$$

The Jacobian of the composition of two rotation vectors is given in appendix A. With the two above expressions, the covariance on  $f$  is simply given by  $W_f = J_1 W_1 J_1^\top + J_2 W_2 J_2^\top$ .



- **Inversion of frame** ( $f, W_f$ ):

Let  $J_I$  be the opposite of the Jacobian of  $f^{(-1)} = (r^{(-1)}, r^{(-1)} \star (-t))$  w.r.t.  $f = (r, t)$ . Taking into account the fact that  $r^{(-1)} = -r$ , we have:

$$J_I = \begin{bmatrix} I_3 & 0 \\ \frac{\partial (r^{(-1)} \star (-t))}{\partial r^{(-1)}} & R^{(-1)} \end{bmatrix}$$

and the covariance of  $f^{(-1)}$  is then  $W_{f^{(-1)}} = J_I W_f J_I^T$ .

## B.4 Conclusion on rigid transformations

We show first that left and right composition are not equivalent for rigid transformations, and that a right error is better adapted because it can be assumed to close to identity. We computed the Jacobians of the three main operations: application to a point, composition and inversion, and saw how to propagate the covariance matrices through these operations. These Jacobians will also be useful in the next section, for rigid transformation estimation using Extended Kalman Filtering.

# C Kalman Filtering

## C.1 Extended Kalman Filtering

Assume we have a set of data  $\{X_i\}$  and a researched state variable  $a$  such as, for each data  $X_i$ , we have the relation

$$h_i(X_i, a) = 0$$

We will call it the measure equation. In our case, the state will be the researched rigid transformation and the data pairs of matched points (section C.2.1) or frames (section C.2.2) or simple measures of this rigid transformation (section C.2.3). Since we work with noisy data, we only know the measured values of data  $\hat{X}_i = X_i + l_i$ . The additive noise  $l_i$  is assumed to be without bias and with a known covariance  $\Lambda_i$ :

$$E(l_i) = 0 \quad \text{and} \quad \Lambda_i = E(l_i l_i^T)$$

The measure equations are generally not linear, but assuming we know a good estimate  $\hat{a}$  of the state  $a$ , 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 (see [Aya91]). Since Kalman Filtering is a recursive filter, we will assume that we have at each step  $i$  an estimate  $\hat{a}_{i-1}$  of the state vector. We can then linearize the measure equation  $h_i$  around  $(\hat{X}_i, \hat{a}_{i-1})$  with Taylor expansion of order one:

$$h_i(X_i, a) = 0 \simeq h_i(\hat{X}_i, \hat{a}_{i-1}) + \frac{\partial h_i}{\partial a}(a - \hat{a}_{i-1}) + \frac{\partial h_i}{\partial X}(X_i - \hat{X}_i)$$

The partial derivatives (Jacobians) are estimated at point  $(\hat{X}_i, \hat{a}_{i-1})$ . This equation can be re-written in the linear form

$$M_i a = Y_i + \omega_i$$

where  $M_i$  is the matrix

$$M_i = \left. \frac{\partial h_i}{\partial a} \right|_{(\hat{X}_i, \hat{a}_{i-1})}$$

$Y_i$  is the linearized measure

$$Y_i = M_i \hat{a}_{i-1} - h_i(\hat{X}_i, \hat{a}_{i-1})$$

and  $\omega_i$  is a noise without bias of computable covariance  $W_i$

$$\omega_i = \left. \frac{\partial h_i}{\partial X} \right|_{(\hat{X}_i, \hat{a}_{i-1})} (\hat{X}_i - X_i) \quad \Longrightarrow \quad W_i = \left( \left. \frac{\partial h_i}{\partial X} \right|_{(\hat{X}_i, \hat{a}_{i-1})} \right)^\top \Lambda_i \left( \left. \frac{\partial h_i}{\partial X} \right|_{(\hat{X}_i, \hat{a}_{i-1})} \right)$$

Given an initial estimate  $\hat{a}_0$  of the state and a covariance matrix  $S_0$  on it, the global criteria to be minimized is now written

$$C = (\hat{a}_0 - a)^\top S_0^{-1} (\hat{a}_0 - a) + \sum_i (Y_i - M_i a)^\top W_i^{-1} (Y_i - M_i a)$$

The recursive solution for the minimization of this criteria is called the Kalman Filter ([Jaz70, Aya91]). At each step, the input is an estimate  $(\hat{a}_{i-1}, S_{i-1})$  of the

state and the linearized measure  $(Y_i, W_i)$ , and the output is the updated estimation of the state  $(\hat{a}_i, S_i)$ . We just recall here the equations ( $\hat{h}_i = h_i(\hat{X}_i, \hat{a}_{i-1})$ ):

$$\begin{aligned} K_i &= S_{i-1}M_i^\top (W_i + M_iS_{i-1}M_i^\top)^{-1} \\ \hat{a}_i &= \hat{a}_{i-1} + K_i(Y_i - M_i\hat{a}_{i-1}) = \hat{a}_{i-1} - K_i\hat{h}_i \\ S_i &= (I - K_iM_i)S_{i-1} \end{aligned} \quad (5)$$

## C.2 Estimation of rigid transformations

The state vector is now a rigid transformation:  $a = (r, t)$ . In each case, we will point out the data, recall the measure equation, give an initial estimate  $(a_0, S_0)$ , and show how to compute the linearized measures. In order to simplify the notations and since all the measure equations are identical inside a particular problem, we will sometimes forgot the index  $i$  of the measure.

### C.2.1 Estimation from point matches

The measure equation is  $h = y - a \star x$ . Let  $\hat{X}$  be the data of the measure with covariance  $\Lambda$ :

$$\hat{X} = \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} \quad \Lambda = \begin{bmatrix} W_x & 0 \\ 0 & W_y \end{bmatrix}$$

It is easy to see that if  $a = (r, t)$  with  $r$  corresponding to the rotation matrix  $R$ , we have  $\frac{\partial h}{\partial X} = [-R; Id]$  and hence  $W = W_y + RW_xR^\top$ . The EKF process is then the following.

- Initialize the state with Identity or a least squares estimate with a big covariance matrix.
- For each pair of matched points  $(\hat{x}_i, \hat{y}_i)$ :
  - Compute  $\hat{h}_i = \hat{y}_i - \hat{a}_{i-1} \star \hat{x}_i$
  - Compute  $M_i = \frac{\partial h}{\partial a} = -\frac{\partial(a \star x)}{\partial a}$  estimated at  $(\hat{f}_i^{(-1)}, \hat{a}_{i-1})$   
and  $W_i = W_{y_i} + R_{i-1}W_{x_i}R_{i-1}^\top$
  - Update  $(\hat{a}_{i-1}, S_{i-1})$  to  $(\hat{a}_i, S_i)$  using equations (5).

### C.2.2 Estimation from frame matches

The measure equation is  $h = f_s^{(-1)} \circ a \circ f_m = 0$ . An initial estimate is easily obtained with the first couple of frames:  $\hat{a}_0 = \hat{f}_{s_0} \circ \hat{f}_{m_0}^{(-1)}$ . Let  $\hat{X}$  be the data of the measure:

$$\hat{X} = \begin{bmatrix} \hat{f}_m \\ \hat{f}_s^{(-1)} \end{bmatrix} \quad \Lambda = \begin{bmatrix} W_{f_m} & 0 \\ 0 & W_{f_s^{(-1)}} \end{bmatrix}$$

and  $J_1, J_2, J_3$  and  $J_4$  the following Jacobians (evaluated using measured values with the formulas of previous sections).

$$J_1 = \frac{\partial(a \circ f_m)}{\partial a} \quad J_2 = \frac{\partial(f_s^{(-1)} \circ (a \circ f_m))}{\partial f_s^{(-1)}}$$

$$J_3 = \frac{\partial(f_s^{(-1)} \circ (a \circ f_m))}{\partial(a \circ f_m)} \quad J_4 = \frac{\partial((f_s^{(-1)} \circ a) \circ f_m)}{\partial f_m}$$

We can then write

$$\frac{\partial h}{\partial a} = J_3 J_1 \quad \frac{\partial h}{\partial X} = [J_4 ; J_2]$$

The computation of  $W$  can then be simplified to  $W = J_4 W_{f_m} J_4^\top + J_2 W_{f_s^{(-1)}} J_2^\top$  and the process can then be written:

- Initialize the state with  $(\hat{a}_0, S_0) = (\hat{f}_{s_0} \circ \hat{f}_{m_0}^{(-1)}, W_{(f_{s_0} \circ f_{m_0}^{(-1)})})$
- For each pair of matched frames  $(\hat{f}_{m_i}, \hat{f}_{s_i})$ :
  - Compute  $\hat{h}_i = \hat{f}_{s_i}^{(-1)} \circ \hat{a}_{i-1} \circ \hat{f}_{m_i}$  and the Jacobians  $J_1, J_2, J_3, J_4$ .
  - Compute  $M_i = J_3 J_1$  and  $W_i = J_4 W_{f_{m_i}} J_4^\top + J_2 W_{f_{s_i}^{(-1)}} J_2^\top$
  - Update  $(\hat{a}_{i-1}, S_{i-1})$  to  $(\hat{a}_i, S_i)$  using equations (5).

### C.2.3 Merging frames

The measure equation is  $h = f^{(-1)} \circ a = 0$ . An initial estimate is easily obtained with the first frame:  $\hat{a}_0 = \hat{f}_0$ . The measure  $\hat{X}$  is now simply  $\hat{X} = \hat{f}^{(-1)}$  and its covariance matrix  $\Lambda = W_{f^{(-1)}}$ . The EKF process is then simply:

- Initialize the state with  $(\hat{a}_0, S_0) = (\hat{f}_0, W_{f_0})$
- For each measure of frame  $\hat{f}_i$ :
  - Compute  $\hat{h}_i = \hat{f}_i^{(-1)} \circ \hat{a}_{i-1}$
  - Compute  $M_i = \frac{\partial(f^{(-1)} \circ a)}{\partial a}$  and  $J_1 = \frac{\partial(f^{(-1)} \circ a)}{\partial f^{(-1)}}$  estimated at  $(\hat{f}_i^{(-1)}, \hat{a}_{i-1})$
  - Compute  $W_i = J_1 W_{f^{(-1)}} J_1^\top$
  - Update  $(\hat{a}_{i-1}, S_{i-1})$  to  $(\hat{a}_i, S_i)$  using equations (5).

### C.3 Conclusion of Kalman Filtering

We show in this section how to use Extended Kalman Filtering to estimate a rigid transformation from matched points, matched frames and a set of measures of a unique rigid transformation. The difficult points are the definition of the measure equation, which rely on a model of the noise process, and the computations of the different needed Jacobians. Once this is done, EKF provide a very efficient way of estimation. Moreover, different measure equations due to different type of features can be used in a single estimation.

Covariance matrices on features and EKF for estimation can then be considered as a unified framework for uncertainty handling on geometric computations.



---

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,  
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY  
Unité de recherche INRIA Rennes, Irista, Campus universitaire de Beaulieu, 35042 RENNES Cedex  
Unité de recherche INRIA Rhône-Alpes, 46 avenue Félix Viallet, 38031 GRENOBLE Cedex 1  
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex  
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

---

Éditeur  
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)  
ISSN 0249-6399