Handbook of Medical Image Processing and Analysis.
I. Bankman Editor. New Edition, Academic Press. 2008.

34
Landmark-Based Registration Using Features Identified Through Differential Geometry

Xavier Pennec
Nicholas Ayache
Jean-Philippe Thirion
INRIA -Asclepios
Project-team

Registration of 3D medical images consists in computing the “best” transformation between two acquisitions, or equivalently, determines the point to point correspondence between the images. Registration algorithms are usually based either on features extracted from the image (feature-based approaches) or on the optimization of a similarity measure of the images intensities (intensity-based or iconic approaches). Another classification criterion is the type of transformation sought (e.g. rigid or non-rigid).

In this chapter, we concentrate on feature-based approaches for rigid registration, similar approaches for non-rigid registration being reported in another set of publication [41, 42]. We show how to reduce the dimension of the registration problem by first extracting a surface from the 3D image, then landmark curves on this surface and possibly landmark points on these curves. This concept proved its efficiency through many applications in medical image analysis as we will see in the sequel. This work has been for a long time a central investigation topic of the Epidaure team [3] and we can only reflect here on a small part of the research done in this area.

We present in the first section the notions of crest lines and extremal points and how these differential geometry features can be extracted from 3D images. In Section 2, we focus on the different rigid registration algorithms that we used to register such features. The last section analyzes the possible errors in this registration scheme and demonstrates that a very accurate registration could be achieved.

1 Features Extraction: Extremal Points and Lines

To extract reliable curves on a surface, most approaches try to generalize the notion of “edges” to smooth surfaces to find the most salient features on the surface: ridges. Prior to the late 1980s and early 1990s, the interest in ridges was mostly theoretical, in areas of mathematics related to catastrophe theory [38, 27, 24, 39, 2]. Crest lines are then defined as the cuspidal edges of a caustic surface, and the link between caustics and curvatures on a surface was established.

Practical applications were then discovered by researchers in computer vision, graphics, and medical imaging together with the specification of algorithms to extract ridges. In the following, we are going to focus on the *crest and extremal line* as introduced in [28, 29] and developed in [44, 45]. Basically, these curves are (subsets of) the loci of the surface where one of the principal curvatures reaches a locally optimum in the associated principal direction. In these works, the crest lines are extracted using third order derivatives of the image intensities. An alternative approach was presented in [17] with the use of a B-spline approximation of the surface.

A different notion of ridges is given in [46, 13]: they are defined as the salient structures of the intensity surface defined by $I = f(x, y, z)$. Here, the ridges are surfaces and are more like results of the medial axis transform than the intuitive notion of salient lines on a surface. Cutting *et al* [9] also developed a method using a template of ridges to assist in their extraction from image data. This method was extended by Dean *et al* [10]. A good review of the many definitions of ridges can be found in [5].

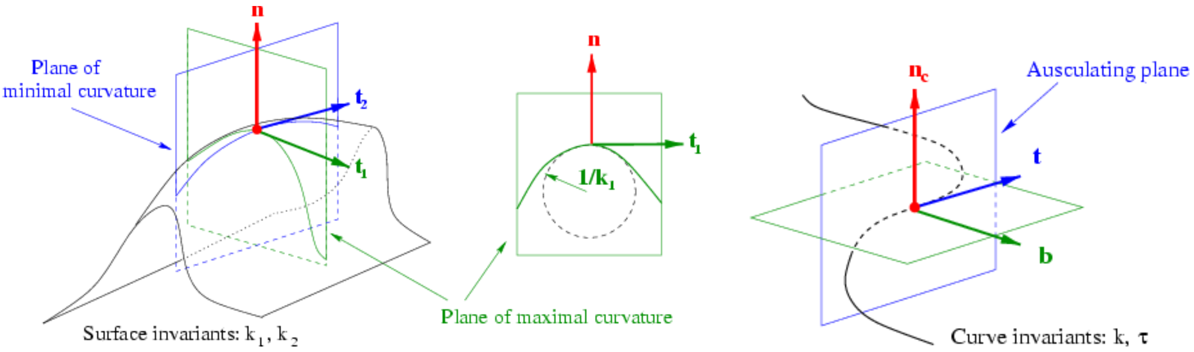

Figure 1: Differential geometry of 3D curves and surfaces. Left: principal directions and curvatures of a surface. Right: Frénet trihedron of a 3D curve and first differential invariants: curvature and torsion.

More recently, there has been a raising interest for ridges in computer graphics because it synthetises some important visual and geometric surface information and could even be used to greatly enhance surface remeshing [31, 1]. In particular, new algorithms have been designed to extract ridges on meshes [32], sometimes with topological guaranties (especially around umbilics) [7, 40, 8].

1.1 Definition and Properties

1.1.1 Differential Geometry of 3D Surfaces

Let us first recall briefly some results of differential geometry about surface curvatures (a good introduction to these notions can be found in [12] or in [24]). In this paper, we call a *smooth surface* a 3D surface which is continuously differentiable up to the third order. At any point P of such a 3D surface, we can define one curvature per direction $\mathbf{t}$ in the tangent plane of the surface. This *directional curvature* $k_{\mathbf{t}}$ is the curvature of the 3D curve defined by the intersection of the plane $(P, \mathbf{t}, \mathbf{n})$ with the surface, where $\mathbf{n}$ is normal to the surface.

Except for the points where this curvature $k_{\mathbf{t}}$ is the same for all the directions $\mathbf{t}$, which are called *umbilic points*, the total set of curvatures can be described with only two privileged directions, $\mathbf{t}_1$ and $\mathbf{t}_2$, and two associated curvature values, $k_1 = k_{\mathbf{t}_1}$ and $k_2 = k_{\mathbf{t}_2}$, which are called respectively the principal directions and the associated principal curvatures of the surface at point P, as shown in Figure 1. These two principal curvatures are the extrema of the directional curvatures at point P, and (except for umbilic points) one of these two is maximal in absolute value, let us say k_1: we call this the *largest curvature*, in order not to be mistaken with the maximal curvature. We simply call the *second (principal) curvature* the other principal curvature k_2.

1.1.2 Extremal Lines

The crest lines are intuitively the loci of the surface where the "curvature" is locally maximal. More precisely, we define them as the loci of the surface where the largest curvature, k_1, is locally maximal (in absolute value) in the associated principal direction $\mathbf{t}_1$. In [29], it is shown that these points can be defined as the zero-crossing of an extremality function e, which is the directional derivative of k_1 in the direction $\mathbf{t}_1$.

We have proposed another method to compute them in [45, 44], for the case of iso-intensity surfaces. Our method is based on the use of the implicit functions theorem. Basically, we have shown that the crest lines can be extracted as the intersection of two implicit surfaces $f = I$ and $e = 0$, where f represents the intensity value of the image, I an iso-intensity threshold, and $e = \nabla k_1 \cdot \mathbf{t}_1$ is the extremality function (see Figure 2, left). We have proposed an algorithm, called the Marching Lines, to automatically extract these crest lines. This algorithm can also be used to overcome some orientation problems (mainly due to the fact that the principal directions are only directions and not oriented vectors), by locally orienting the principal directions along the extracted lines.

In fact, for each point of the surface, two different extremality coefficients can be defined, corresponding to the two principal curvatures:

$$e_1 = \nabla k_1 \cdot \mathbf{t}_1 \qquad \text{and} \qquad e_2 = \nabla k_2 \cdot \mathbf{t}_2 \qquad (1)$$

We found experimentally that the maxima (in absolute values) are more stable landmarks than the minima: crests or rifts (maxima) are stable, whereas the loci in a valley where the ground floor is the flattest (minima) are very sensitive to small perturbations in the data.

We call *extremal lines* all the lines defined as the zero-crossings of either e_1 or e_2. There is therefore four major different types of extremal lines, depending of whether the corresponding curvature is the largest or the second one and whether it is a local maximum or minimum. Furthermore, the signs of the largest and second curvatures help to distinguish between four additional sub-types of extremal lines, leading to a classification into 16 types. The crest lines are two of them: positive largest curvature maxima ($k_1 > 0$ and $\nabla e_1 \cdot \mathbf{t}_1 < 0$)

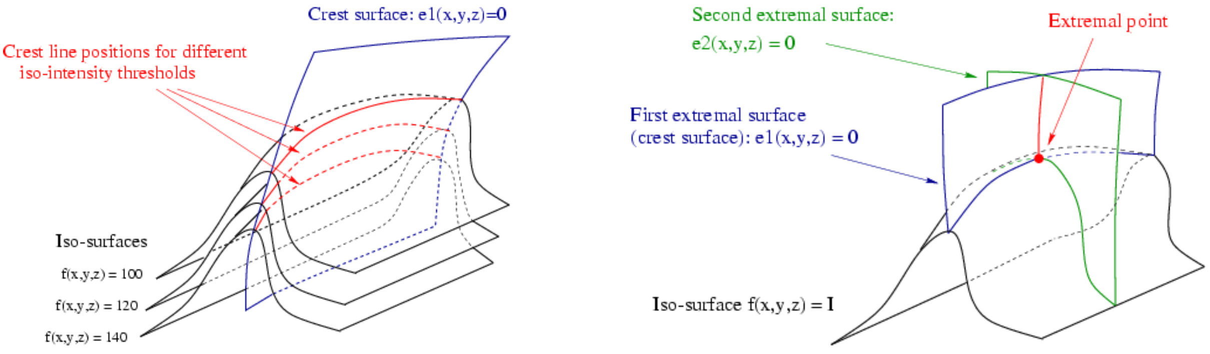
Crest surface: e1(x,y,z)=0
Crest line positions for different iso-intensity thresholds
Iso-surfaces
f(x,y,z) = 100
f(x,y,z) = 120
f(x,y,z) = 140
Second extremal surface: e2(x,y,z) = 0
Extremal point
First extremal surface (crest surface): e1(x,y,z) = 0
Iso-surface f(x,y,z) = I

Figure 2: Left: Crest lines as the intersection of two implicit surfaces. Right: Definition of the extremal points as the intersection of three implicit surfaces.

and negative largest curvature minima ($k_1 < 0$ and $\nabla e_1 \cdot \mathbf{t}_1 > 0$).

1.1.3 Extremal Points

We now define the extremal points as the intersection of the three implicit surfaces: $f = I$, $e_1 = 0$ and $e_2 = 0$. The notions of extremal lines and extremal points are closely related to the notion of corner points, in 2D images, as defined in [23], [30], and [11]. A study of the evolution in 2D of corner points with respect to the scale can be found in [16]. A similar study on the scale-space behavior of the extremal lines and the extremal points was presented in [15].

Extremalities e_1 and e_2 are geometric invariants of the implicit surface $f = I$: they are preserved with rigid transforms (rotations and translations of the object). Therefore, the relative positions of the extremal points are also invariant with respect to a rigid transformation, i.e. for two different acquisitions of the same subject. There are 16 different types of extremal points, depending on the type of extremality: local minimum or maximum of the extremalities e_1 and e_2 and the signs of k_1 and k_2. This classification can be used to reduce the complexity of the matching algorithm.

However, the intuitive interpretation of extremal points is not straightforward. The extremal lines are 3D curves, for which we are able to compute the curvature, but the extremal points are generally not the points of the extremal lines whose curvature is locally maximal. Even if they are not extremal curvature points, the extremal points are very well defined, and there is no reason for their locations along the extremal lines to be less precise that the lines positions themselves, because the precision of the computation of k_1 and k_2 is approximately the same.

1.1.4 Geometric Characteristics

Let us begin with the points on a surface. We have already seen (Figure 1, left) that any such point could be provided with a trihedron $(\mathbf{n}, \mathbf{t_1}, \mathbf{t_2})$ formed by the normal to the surface and the two principal directions. As our points are also on extremal lines, we could provide them with the differential characteristics of 3D curves (Figure 1, right), i.e. the Frénet trihedron $(\mathbf{t}, \mathbf{n_c}, \mathbf{b})$, where $\mathbf{t}$ is the tangent to the curve, $\mathbf{n_c}$ its normal and $\mathbf{b}$ the binormal.

These two trihedrons are not the same as the extremal lines are generally not lines of curvature. However, as the curve is embedded in the surface, the tangent to the curve $\mathbf{t}$ is constrained to be in the tangent plane of the surface spanned by $(\mathbf{t_1}, \mathbf{t_2})$. Thus, there are two independent parameters characterizing the relative configuration of the trihedron: we can measure two angles $\theta = (\widehat{\mathbf{t}, \mathbf{t_1}})$ and $\phi = (\widehat{\mathbf{n_c}, \mathbf{n}})$. These characteristics are invariant with respect to rigid transformations.

Two other invariants come from the surface (principal curvatures k_1 and k_2). One could also think to add the curvature k, the torsion τ of the curve and the geodesic torsion τ_g of the curve with respect to the surface but it appears that k and τ_g are completely determined by the surface invariants: $k \cos\phi = k_1 \cos^2\theta + k_2 \sin^2\theta$ and $\tau_g = (k_2 - k_1)\cos\theta \sin\theta$. Thus, we are left with the torsion of the curve.

However, the computation of the Frénet trihedron $(\mathbf{t}, \mathbf{g}, \mathbf{b})$ and the curve torsion τ has to be done on the extremal curve itself after its extraction. If this can be done directly on the polygonal approximation, a much better method is to compute the characteristics on a local B-spline approximation of the curve [18].

1.2 The Automatic Extraction of the Extremal Points

In practical cases, e_1 and e_2 can be computed for each point of the 3D image with the equations described in [43] directly from the differentials of the intensity function of the image f. We compute these differentials with linear filtering, using the convolution of the discrete image with the differentials of the Gaussian func-

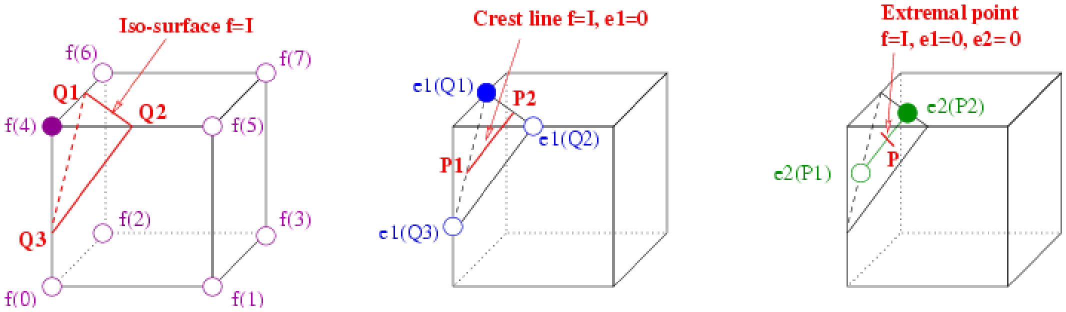

Figure 3: Extraction of the Extremal Points. An empty circle denotes a positive value, whereas a filled circle indicates a negative one.

tion $e^{-\|x\|^2/2\sigma^2}$. The normalization of these filters is not straightforward; we use the responses to simple polynomials, as proposed in [29]. We choose the Gaussian function because it is isotropic, a prerequisite if we are looking for geometric invariants for rigid transformations. Different values of σ can be chosen, depending on the level of noise in the 3D images. Changing σ is somewhat equivalent to changing the scale at which we look for extremal lines and points.

The hypothesis that the iso-surfaces are a good representation of the surface of organs for the case of medical images is a prerequisite: sometimes, the iso-surface can be extracted directly from the 3D image, such as the skin surface in Magnetic Resonance Image (MRI) or the bones in X-ray scanner images. For other soft tissues, such as for the brain surface, a pre-segmentation step is required to isolate the brain from the rest of the data. This can be done with a combination of mathematical morphological operators, filtering, and the search for connected parts or with an automatic “surface edge” extractor, such as the zero-crossing of the image Laplacian. In all cases, the final step of the segmentation is performed using iso-surface techniques.

1.2.1 Computation of the Extremal Points in a 8-Voxel Cell

One solution to get the set of extremal points of the 3D image is to compute e_1 and e_2 for all the voxels of the 3D image and then to consider individually each cubic cell, formed with 8 voxels (8-cell), as shown in Figure 3. There are therefore three values defined for each vertex of the cube: f, e_1, and e_2. The extremal points in that 8-cell are defined as the intersection of the three implicit surfaces $f = I$, $e_1 = 0$, and $e_2 = 0$. The method varies according to the type of interpolation or convolution function used to extend continuously the three values at the vertices of the cubic cell to the entire cell. The tri-linear interpolation is a good first order approximation.

The extraction of a polygonal approximation of the crest lines with some warranties about the topology and the orientation of the reconstructed 3D curves is presented with the marching line algorithm [45]. Its extension to the extraction of the extremal points was performed in [44]. We briefly recall here the method on a very simple example where the iso-surface is a triangle in the cell. This operation can be extended to any configuration of the values of f and e_1 while ensuring that the extracted segments form a continuous and closed 3D curve (except when f or e_1 is not defined, for instance at the borders of the image). The original algorithm also considers orientation problems, which allows us to distinguish between minimum and maximum extremal points.

The first step (Figure 3, left) is to extract the iso-surface within the cell. The iso-surface intersects the edges on the cell with the value I. Computing, by linear interpolation along the edges, the points where $f = I$, we get the three points $\{Q_1, Q_2, Q_3\}$. Since we are using a tri-linear interpolation within the cell, the intersection of the iso-surface with the cell is the triangle $\{Q_1, Q_2, Q_3\}$.

In the second step (Figure 3, middle), we compute the values of e_1 for $\{Q_1, Q_2, Q_3\}$, by linear interpolation along the edges of the cubic cell. If they have the same sign, there is no extremal line of e_1 in this cell. Otherwise we look for the two points along the triangle edges where the interpolated value of e_1 is null: we get the two points $\{P_1, P_2\}$ which form a segment. This is the approximation of the extremal line within the cell.

The last step (Figure 3, right) is to compute the position of the extremal point. Since P_1 and P_2 lie on the surface of the cell, we compute the value of e_2 at these points with a bi-linear interpolation of e_2 in the faces. If the two values have the same sign, there is no extremal point on this cell. Otherwise, as is shown here, we compute its position P by interpolating the zero value along the segment.

1.2.2 Randomized Extraction of Extremal Points

Of course, we could look for extremal points in all the possible cells of the image, excepting regions of null gradient and umbilics. However, it is much more efficient to randomize the search: we start with seed cells,

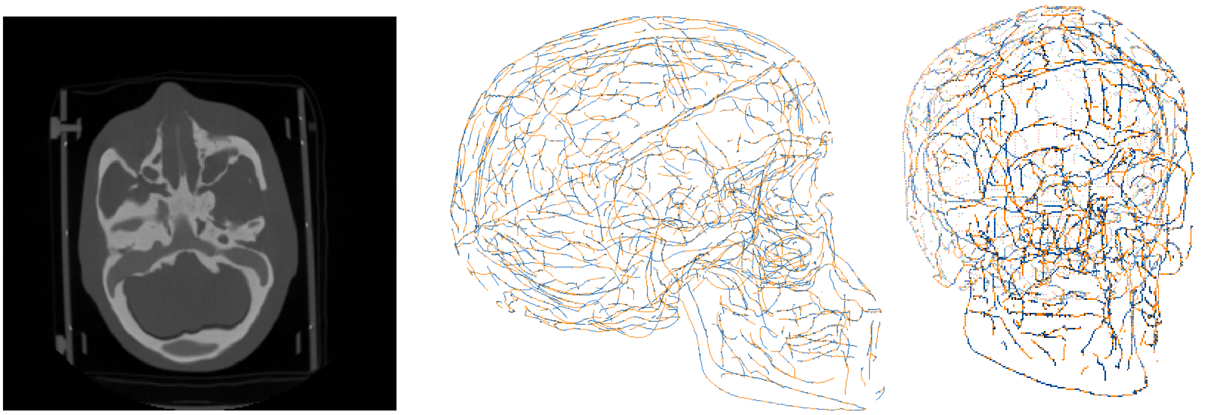

Figure 4: **Left:** An axial slice of a 3D CT image of a dry skull in Plexiglas. **Middle and right:** the crest lines extracted on this image. The iso-intensity was manually chosen to delimit the skull. Original 3D Image courtesy of GE-CGR, Buc, France.

randomly chosen in the 3D image and discard the ones for which the sign of $f - I$ is the same for all the vertices. Then we compute the values of e_1 for the 8 vertices of the cell. Once again, a simple test discards the cells which are not crossed by a k_1 extremal line (the sign of e_1 is the same for the 8 vertices). If there is an extremal line, we extract it from end to end, using the Marching Lines algorithm (we follow the extremal line "marching" from one cell to the next).

At each point of the polygonal approximation of the crest line, we compute the second extremality e_2 by bilinear interpolation. If there is a sign change, we compute the extremal point on the segment of the extremal line that we are currently following.

The randomized implementation of the Marching Lines allows us to extract the main extremal lines (i.e., the longest ones, which experimentally appeared to be the most reliable ones) of the 3D image, with only very few seeds (with respect to the total number of voxels), randomly distributed in the 3D images. The probability of missing an extremal line is approximately proportional to the inverse of its length. This method reduces drastically the number of computations to perform, compared to the extensive implementation: typically, one uses 10% of the number of voxels as seeds. Even if the set of generated extremal points is not complete, it is generally sufficient to perform a reliable 3D registration.

1.3 Example of Extracted Extremal Lines and Points

In Figure 4, we can see an example of the lines extracted automatically (with a manual choice of the iso-intensity threshold) in a CT image of a dry skull. Some of the 550 crest lines may be recognized as anatomical landmarks, such as the orbits or the inferior border of the mandible. The lines are colored by the sign of the e_2 extremality. Thus, extremal points are located at the color changes along the lines. There are around 3000 such extremal points.

In an MR image, the surface of the brain is not very well defined by an iso-intensity of the image. A presegmentation step is usually needed to isolate the brain from the rest of the data. This can be done with a combination of mathematical morphological operators, filtering, and the search for connected parts or with an automatic "surface edge" extractor, such as the zero-crossing of the image Laplacian. In Figure 5, we used a segmentation of the surface of the brain and extracted the crest lines on this surface. Lines in red (with a positive largest curvature) roughly correspond to sulci whereas blue lines (with a negative largest curvature) could be interpreted as gyri.

2 Rigid Registration

Let us now consider two images of the same modality and of the same patient, but in a different position. We extract extremal lines on both images. The problem is to put into correspondence the two sets of lines (the model and the scene), which is often called the matching step, and to compute the best rigid transformation that superimposes the matched lines.

It is important to note that a global registration algorithm, for instance superimposing the barycenters of all points and the inertia axes, will often fail due to the occlusion. Indeed, the images being taken in different positions, the region of interest are frequently different in the two images, leading to crest lines and extremal points present in one image and not in the other. The images noise will also induce the extraction of spurious lines and points in different parts of the two images.

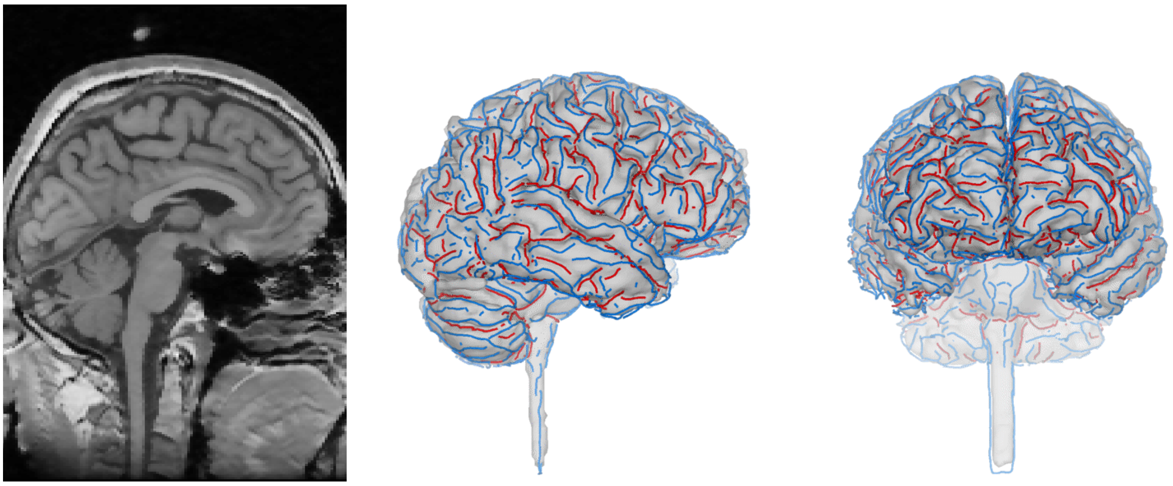

Figure 5: **Left:** A sagittal slice of a 3D MR image. **Middle and right:** 3D views of the extremal lines extracted superimposed on the surface of the brain. Original MR images and segmentation courtesy of Prof. R. Kikinis, Brigham and Women's Hospital, Boston.

2.1 Curve Registration

Several algorithms adapted from computer vision have been proposed and used over time. In [18], Guéziec matches the crest lines using a combination of geometric hashing [25] and Hough transform (see for instance [26]). The basic idea was to index each point of each model crest line in a hash-table using its invariant characteristics. At recognition time, the invariants of each scene crest line point are used to recover, thanks to the hash table, the possible matches with model points (geometric hashing step). For each match (i.e., couple of model and scene points), a rigid transformation is computed by superimposing the Frénet frames and used to index the match in a new accumulator sampling the rigid transformation space (Hough transform step). Densely populated cells in this second accumulator are detected as rigid body transformations that are candidates to match a large number of crest points. For each such cell, a refined least-squares transformation is computed using the matches indexed in this cell.

2.2 Extremal Points Registration using Alignment

With the development of completely automated methods to extract crest lines and the higher resolution of images, the number of crest lines drastically increased, leading to a much higher density of invariants in the hash table. This could lead to an important number of false positives that would overwhelm the correct matches. The maximum complexity would then be reached and the algorithm could even provide a wrong answer. To address this problem, Thirion reduced once again the image information by keeping only a very small number of specific points on the crest lines: the extremal points. Typically, they represent only 16% of the number of crest line points, but we are still left with 2000 to 5000 points in each image.

Thirion used in [43] another computer vision based technique: alignment (or prediction-verification) [4, 20]. The basic idea is to compute the registration of each triplet of model points with each triplet of scene points, superimpose the two sets of points using this transformation and verify this match using an iterative closest point algorithm (see Section 2.4). However, the search for compatible triplets in the model and the scene can be reduced since there are some unary invariants (the principal curvatures k_1 and k_2), secondary invariants (e.g. the distance between the two points, or the relative orientation of the surface normals and the principal curvatures) and even ternary invariants (involving the whole triplet of extremal points). Thirion used 18 such invariants for each triplet, pre-computed and stored them in a hash table to retrieve in constant time the compatible triplets. Thus, the complexity of the algorithm is $O(n^4)$ since there are n^3 triplets, and a verification of $O(n)$ for each triplet match. In practice, this complexity is not reached as we can stop as soon as a given number of points is matched after verification (typically 10 %).

2.3 Substructure Matching with Frame Features

We came back in [19] to geometric hashing, but the idea was to use all the geometric information on the features while taking great care of the uncertainty handling for the algorithm to be robust (see [33] for an analysis of recognition algorithms with uncertain geometric features). In addition to the point's position, we can add the normal vector $\mathbf{n}$ and the two principal directions $\mathbf{t_1}$ and

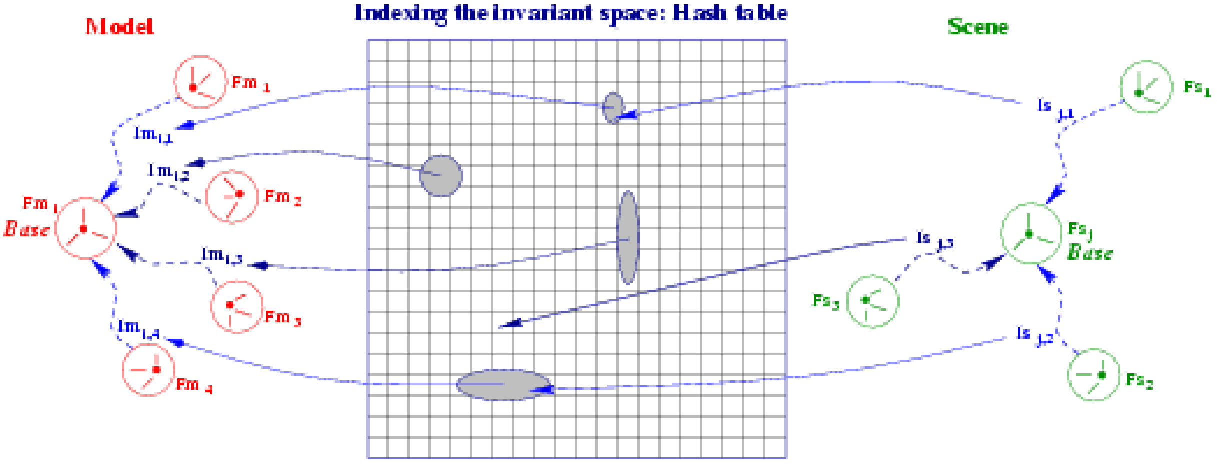

Figure 6: **Preprocessing:** the 6D invariant vector associated with every couple of model frames is computed with its error zone and used as an index for the couple in the hash table. **Recognition:** for each scene frame couple, we compute the 6D invariant vector and retrieve through the hash table every compatible model frame couple. For each such couple, we tally a vote for the matching of the reference frames (here the match (Fm_i, Fs_j) scores 2).

$\mathbf{t_2}$ of the surface to constitute a local coordinate system, or a frame.

In this context, each medical image is modeled by a set of frames and the matching problem is to find the correspondences between two subsets of frames that are in the same configuration in the two “images”, up to a global rigid transformation.

2.3.1 Invariant Representation: Preprocessing Step

To obtain an invariant representation with respect to the global position and orientation of the considered structure, we can express the configuration of all frames relative to one frame (called the basis). For efficiency, this representation is stored in a hash table and, for correctness, we include the uncertainty of each invariant. As only part of the frames are in the same configuration in the two images, the one chosen as the basis may not be present in the other image. The preprocessing step is thus repeated with each frame as the basis.

2.3.2 Recognition Step

Choosing a frame of the second structure (the scene) as the basis, we compute the invariant representation and retrieve, thanks to the hash table, what are the compatible model frame couples. If the basis belongs to a common substructure, then a significant number of frames are in the same configuration with respect to it. We then match the model and scene bases (Fig. 6).

This process is repeated for every extremal point as the basis to find its possible matches in the model and we only keep the matches that are above a given threshold (typically 10 % of the number of extremal points).

2.3.3 Clustering Compatible Matches and Verification

For each individual match, we maintain during the recognition step an estimation of the associated transformation by fusing the transformations between confirming frames. To group matches belonging to the same (rigid) substructure, we run a very simple clustering algorithm on the associated transformation. Indeed, up to measurement errors, frames should undergo a similar transformation within a single substructure. Each cluster is then refined by an iterative closest neighbor technique where we enforce symmetry of the matches and verify their validity with a χ^2 test.

2.3.4 Matching Crest Lines

In order to reduce once again the complexity of the algorithm, we exploited in this method the structure of the extremal points: they belong to crest lines. The principle is to consider each model crest line as a different object. Index all model lines in the same hash-table, we retrieve at recognition time the model lines corresponding to each scene crest line.

However, different crest line matches can correspond to different transformations. Thus, we run once again our clustering algorithm on the transformations to find out the compatible line matches and we obtain a single transformation from the model to the scene image.

2.4 ICP on Frames

When images are close enough, one can use still another algorithm: the Iterative Closest Point [6, 47]. The basic principle is the following. For each scene point, we look for the closest point in the model with the current transformation, compute a new rigid transforma-

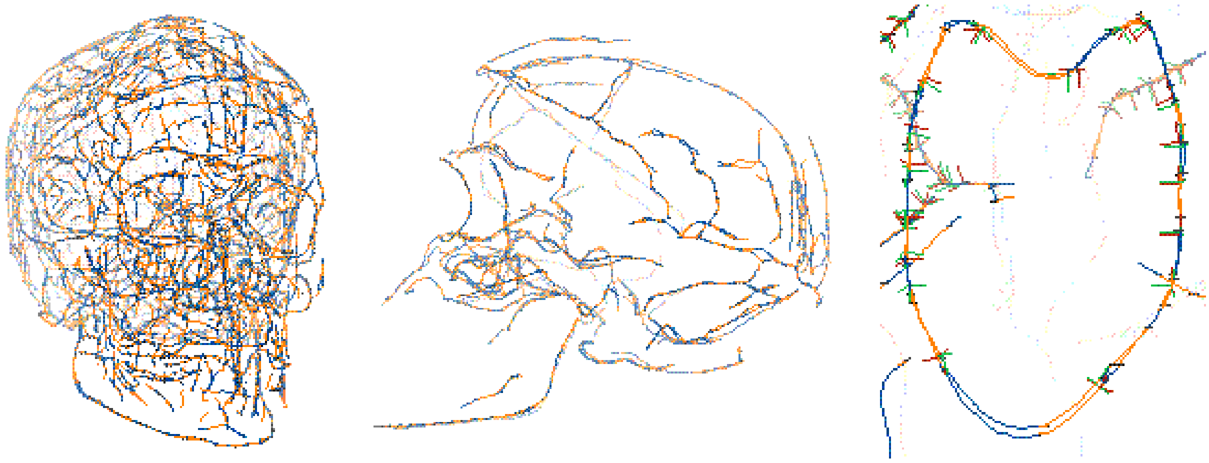

Figure 7: Example of registered crest lines between two CT skull images of the same phantom acquired in two different positions. Extremal points are represented by a color change from yellow to blue on the lines. **Left:** Front view with all crest lines from the two skulls after registration. **Middle:** Left view of the matched crest lines. **Right:** Closeup on the occipital foramen on the right. In this last image, the width of a line is a tenth of a voxel, which shows the very precise registration of these extremal points. One can also see that the trihedron part of the matched frames is very well conserved.

tion with these matches, and iterate the process until convergence.

Of course, since we have more geometric information than just the point position, we use a generalization: the Iterative Closest Feature [33]. The idea is to use a higher dimensional space for the closest point search. In our case, the space is made of the extremal point position, the trihedron $(\mathbf{n}, \mathbf{t_1}, \mathbf{t_2})$, and the unary invariants k_1 and k_2. The important point is to set an appropriate metric on this space in order to combine efficiently the different units of measurement. In our algorithm, this is done using the inverse of the covariance matrix of the features. This matrix can be re-estimated after convergence and the whole process iterated. However, we did not observe a critical influence of the covariance matrix values, as soon as it is approximately respecting the variation range of the different components.

2.5 Examples of Rigid Registrations

2.5.1 Registration of CT Images of the Skull

Figure 7 presents an example of the registration of two CT images of the a dry skull in a Plexiglas box in two different positions. We used the geometric hashing algorithm on frames (Section 2.3). As the transformation between the two images is close enough to the identity, the ICP algorithm also gives very similar results. About 75 crest lines are matched with more than 4 extremal points among the 550 lines in each image leading to a total of 550 matched extremal points (only on the 75 matched lines). Using the techniques described in Section 3.2.2, we have computed that the typical object accuracy (the expected standard RMS error on image super-imposition due to the transformation in the area of the matched features) is 0.04 mm, whereas the typical corner accuracy is 0.1 mm. This is to be compared with the voxel size: 1 x 1 x 1.5 mm.

2.5.2 Registration of MR Images of the Head

Figure 8 is an example of the registration of two MR T1 images of the same patient. In this case, 240 crest lines are matched among approximately 2100 in each image, for a total of 860 matched extremal points among 3600 in each image (about 25 %). We used the zero-crossing of the Laplacian to define the interest surfaces. Thus, there are crest lines all over the head. However, if some of the matched lines are located on the surface of the skin (we can recognize the nose and the eyes), most of them are located on the surface of the brain. The typical object accuracy of the registration is 0.06 mm for a typical corner accuracy of 0.125 mm. Once again, the accuracy is far below the voxel size (0.97 x 0.97 x 1.5 mm).

3 Robustness and Uncertainty Analysis

Once we have registered the images, i.e. found matches and a rigid transformation, the question is: how confident can we be with this result? There are two main types of errors in feature-based registration algorithms. Firstly, the matches could be completely wrong and we simply recognized by chance n features in approximately the same configuration. This is called a *gross*

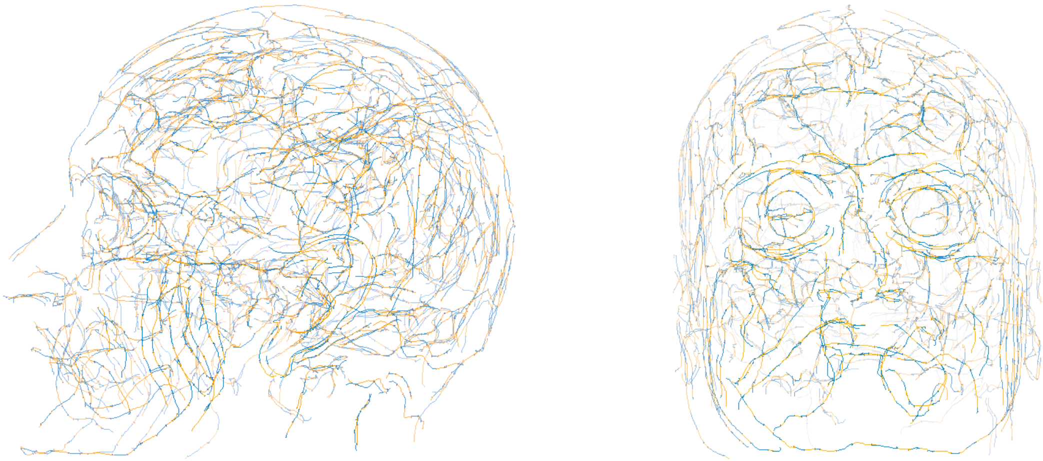

Figure 8: Example of registered crest lines between two MR T1 images of the same patient. Only the 240 matched lines are displayed. Extremal points are represented by a color change from yellow to blue on the lines. **Left:** View of matched crest lines from the left of the head. **Middle:** View from the front.

error in statistics and a *false positive* in recognition. But even if we got the matches right, the features we are using to compute the registration are corrupted by noise and induce a small error, or uncertainty, on the transformation parameters. In this section, we analyze in turn these two types of error.

3.1 Robustness Analysis

Since our features are noisy, we had to allow for a certain error when matching them. In the registration algorithm of Sections 2.3 and 2.4, this is computed from the covariance matrices. The existence of such an error zone allows us to match features that by chance fall in this area. When this probability is sufficiently high, individual false matches can combine themselves (conspiracy) to produce an important matching score.

However, we should note that such a false positive is a correct recognition from a feature point of view (a globally consistent matching) but is incorrect from the image or data point of view. This problem is inherent to the ascendent organization of information: some important information can be dismissed by simplifying the image as a set of features.

In [34], we developed a generic method to evaluate the probability of matching τ features just by chance. The basic framework is the following. Let us assume for the moment that the transformation between the two images is fixed. First, we compute the *selectivity* η, which is the probability that a random feature (uniformly distributed in the image) falls in a given error zone. Then, we compute the probability that at least τ of the m scene features fall in one of the n model error zones. In our analysis, computations are simplified by assuming that all features are randomly distributed.

Now, we will accept a match if there exists one transformation such that at least τ features are put into correspondence. Thus, to obtain the mean number of false positives, we just have to integrate over all possible transformations. Let d be the "diameter" of the images, we get the following estimation:

$$P \simeq \frac{(2\pi d)^3}{3}\left(1 - e^{-(nm\eta)}\sum_{j=0}^{\tau}\frac{(nm\eta)^j}{j!}\right).$$

In the example of Section 3.3, we compute that the selectivity is $\eta_{pt} = 2.10^{-6}$ if we just use the position of the extremal points and $\eta_{fr} = 1.5.10^{-8}$ if we model them using frames. The diameter of the image is $d \simeq$ 400mm and we extracted around 2,500 extremal points in each image. We plot in Figure 9 the number of false positives P with these values.

The actual matches found involve about 500 features and the probability of being a false positive is thus practically zero. However, we must be careful that the "object" we registered is not always the one we wanted, even if this is definitely not a false positive: there can be several different rigid motions in a single image, for instance the skull and the brain in MR images.

3.2 From Feature to Transformation Uncertainty

Here, we assume that the matches are right. However, measurement errors on the features induce an estimation error on the transformation. We developed in [36, 35] a method where we register the features, estimate the noise on the frames and propagate this noise to estimate the uncertainty of the rigid transformation.

3.2.1 Feature Uncertainty

For extremal points (modeled as frames), we proposed a "compositive" model of noise. Basically, this means

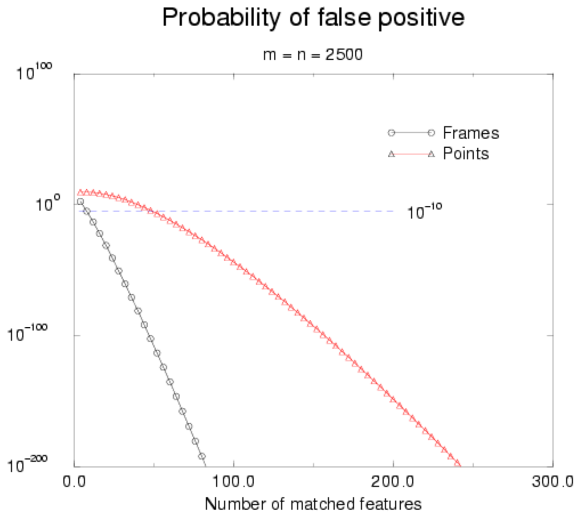

Figure 9: Qualitative estimation of the number of false positives involving at least τ matches in MR images of 2500 features. Comparison between frames and points: we need roughly 5 times more point matches than frame matches to obtain the same probability (10 frames and 56 point matches for a probability of 10^{-10}).

that the noise is assumed to be identical on all extremal points *in the local frame* (i.e., with respect to the surface normal and the principal directions). This has to be compared with the standard additive noise model on points where we assume an identical noise with respect to the image axes. In the case of the MR images of the next Section, this leads to an interesting observation: we draw in Figure 10 a graphical interpretation of the covariance matrix estimated on the extremal points after registration.

We obtain an approximately diagonal covariance matrix with standard deviations $\sigma_{t_1} \simeq \sigma_{t_2} \simeq 2$ deg, $\sigma_n = 6$ deg for the rotation vector (these are typical values of the angle of rotation around the corresponding axis) and $\sigma_{x_{t_1}} \simeq \sigma_{x_{t_2}} \simeq 0.8$ mm, $\sigma_{x_n} = 0.2$ mm for the position. As far as the trihedron part is concerned, this means that the surface normal is more stable than the principal directions, which is expected since the normal is a first order differential characteristic of the image and the principal directions are second order ones.

For the position, the coordinate along the normal is once again more stable than in the tangent plane for the same reasons. The 3D standard deviation of the position is $\sigma = 1.04$, which is in agreement with the additive noise model on points. However, for the additive noise model, the estimated covariance is isotropic. Thus, using an adapted model of noise on frames allows us to extract much more information about the feature noise. This constitutes an *a-posteriori* validation of our "compositive" model of noise on extremal points.

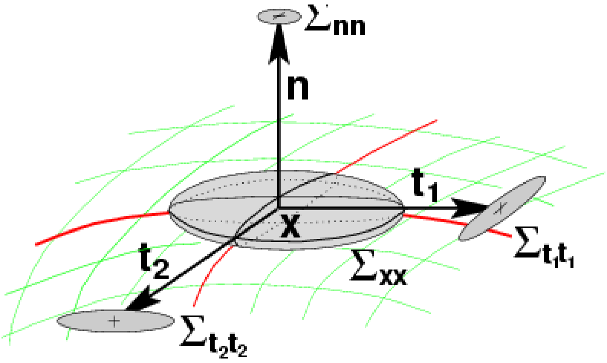

Figure 10: Graphical interpretation of the "compositive" noise model estimated on extremal points. The uncertainty of the origin (point X) is 4 times larger in the tangent plane than along the surface normal. The uncertainty of the normal is isotropic, whereas the principal directions t_1 and t_2 are 3 times more uncertain in the tangent plane.

3.2.2 Transformation Uncertainty

Now the problem is to propagate the feature uncertainty to the rigid transformation. Let f represent a rigid transformation and $\hat{\chi}$ the observed data. The optimal transformation $\hat{f}$ minimizes a given criterion $C(f, \chi)$ (for instance the least-squares or the Mahalanobis distance). Let $\Phi(f, \chi) = \partial C(f, \chi)/\partial f$. The characterization of an optimum is $\Phi(\hat{f}, \hat{\chi}) = 0$. Now, if the data are moving around their observed values, we can relate the new optimal parameters using a Taylor expansion. Let $\hat{H} = \partial\Phi/\partial f$ and $\hat{J}_\Phi = \partial\Phi/\partial\chi$ be the values of the second order derivatives of the criterion at the actual values $(\hat{\chi}, \hat{f})$. We have

$$\begin{aligned} & \Phi(\hat{\chi} + \delta\chi, \hat{f} + \delta f) \simeq \hat{J}_\Phi.\delta\chi + \hat{H}.\delta f = 0 \\ \Leftrightarrow \quad & \Sigma_{\hat{f}\hat{f}} = \mathbf{E}\left(\delta f.\delta f^{\mathrm{T}}\right) = \hat{H}^{(-1)} \hat{J}_\Phi \Sigma_{\hat{\chi}\hat{\chi}} \hat{J}_\Phi^{\mathrm{T}} \hat{H}^{(-1)} \end{aligned}$$

Thus, we can express (an approximation of) the covariance of the resulting transformation using the covariance on features and the criterion derivatives.

However, a covariance matrix on a rigid transformation is quite hard to understand since it mixes angular values for the rotation and metric values for the translation. To characterize the transformation accuracy with a single number, we can compute the uncertainty (expected RMS error) induced on a set of representative points by the registration uncertainty alone (without the uncertainty due to the feature extraction). In our case, two sets are particularly well suited: the position of the matched extremal point represents the localization of the object of interest, whereas the corners of the image symbolize the worst case. In the example below, we find for instance a typical boundary precision around $\sigma_{corn} = 0.11$ mm and a typical object precision far below the voxel size: $\sigma_{obj} = 0.05$ mm for echo-1 regis-

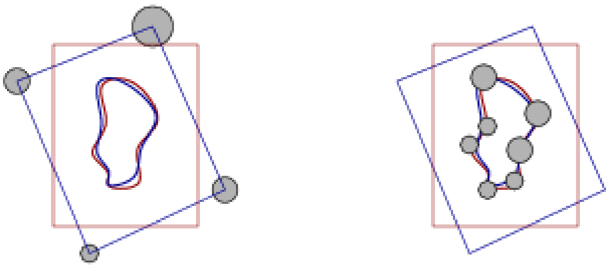

Figure 11: Uncertainty induced on the point positions (image corners, left; some object points, right) by the transformation.

trations. The values are even a little smaller for echo-2 registrations: $\sigma_{corn} = 0.10$ and $\sigma_{obj} = 0.045$ mm.

3.2.3 Validation Index

Last but not least, we need to validate this whole chain of estimations to verify if our uncertainty prediction is accurate. We observed that, under the Gaussian hypothesis, the Mahalanobis distance between the estimated and the exact transformation (or between two independent estimations of the same transformation) should be χ_6^2 distributed (if the covariance matrix on the estimation is exact). To verify this, the idea is to repeat a registration experiment N times and to compute the empirical mean value $I = \bar{\mu}^2 = \frac{1}{N} \sum \mu_i^2$ and the variance σ_I^2 of this Mahalanobis distance. The values for an exact χ_6^2 are respectively 6 and 12. We can also verify using the Kolmogorov-Smirnov test (K-S test) that the empirical distribution corresponds to the exact distribution. The *validation index* I [36] can be interpreted as an indication of how the estimation method under-estimates ($I > 6$) or over-estimates ($I < 6$) the covariance matrix of the estimated transformation. It can also be seen as a sort of relative error on the error estimation.

We run several sets of tests with synthetic data and verify that our uncertainty estimations very perfectly validated for more than 15 extremal point matches. Now, the question we want to answer is: is it still valid for real data?

3.3 Validation with Real Data

The experiment is performed using multiple 2D contiguous Magnetic Resonance images (MRI) which constitute a 3D representation of the head. The images are part of an extensive study of the evolution of the Multiple Sclerosis (MS) disease performed at the Brigham and Woman's Hospital (Harvard Medical School, Boston) by Dr. Guttmann and Prof. Kikinis. Each patient underwent a complete head MR examination several times during one year (typically 24 different 3D acquisitions). The aim is to register precisely all the images acquired at multiple time points in order to segment the lesions and evaluate very accurately their evolution.

Each acquisition provides a first echo image and a second echo image (typically 256 x 256 x 54 voxels of size .9375 x .9375 x 3mm). The two images represent the same slice of T2 weighted signal imaged at different echo times. Thus, they are expected to be in the same coordinate system. This protocol was designed to optimize the contrast in the two channels for easier tissue segmentation. Considering two acquisitions A and B, the registrations of echo-1 images (A_1 to B_1) and echo-2 images (A_2 to B_2) give two relatively independent estimates of the genuine transformation from A to B. The comparison of these two transformations using the Mahalanobis distance gives a real validation index which can be tested for the accuracy of the uncertainty estimation.

In this experiment, the images being close enough, we used the iterative closest feature algorithm. Typically, we match 1000 extremal points out of the about 3000 extracted with a residual mean square error (RMS) of about $1mm$.

3.3.1 Direct Validation Shows Biases

With n different acquisitions, we can run $n * (n-1)/2$ registrations per echo. In a first experiment, we compared directly the registrations between the corresponding echo-1 and echo-2 images. The resulting validation index clearly indicates that the transformations do not agree ($\bar{\mu} = I > 50$ instead of 6). However, our registration method cannot detect systematic biases.

To discover the biases, we ran a series of experiments where we repeated the same registration while varying the algorithm parameters. This confirms that the observed uncertainty is similar in size and shape to the predicted one. Moreover, other experiments show that the inter-echo-1 and the inter-echo-2 registrations are compatible but the two groups significantly differ (Figure 13). Thus we concluded that there was a systematic bias between echo-1 and echo-2 registrations. Additional experiments showed that the bias was different for each registration.

3.3.2 Estimation of the Biases

To estimate the biases, we first observed that the transformation from image A_1 to image B_2 can be written $f_{A_1 B_2} = f_B \circ f_{AB_1} = f_{AB_2} \circ f_A$. If measurements where perfect, the bias f_A could be expressed for any other image Z: $f_A = f_{AZ_2}^{(-1)} \circ f_Z \circ f_{AZ_1}$. Since measurements are noisy, we obtain an estimator of the bias f_A by taking the Fréchet mean value [37]:

$$\bar{f}_A = \arg\min_f \left\{ \sum_{Z \neq A} \operatorname{dist}^2 \left(f \ , \ f_{AZ_2}^{(-1)} \circ f_Z \circ f_{AZ_1} \right) \right\}.$$

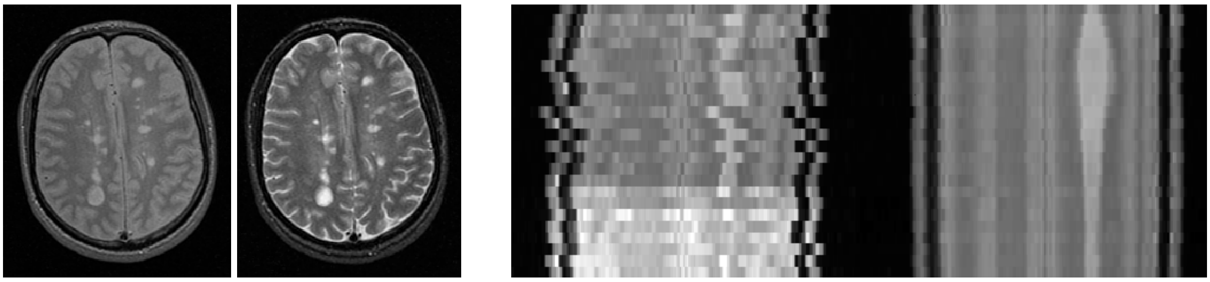

Figure 12: **Left:** Example of MS images. The same slice of one acquisition in echo-1 (left) and echo-2 (right). **Right:** evolution of an image row going through a lesion across 24 time points over a year. Left: without registration; Right: after registration and intensity correction. Original 3D Images courtesy of Dr. Charles Guttmann and Prof. Ron Kikinis from the Brigham and Woman's Hospital (Harvard Medical School, Boston).

	$\bar{I}$	σ_I	K-S test	num. im.	num. reg.
Theoretical values	6	$\sqrt{12} = 3.46$	0.01 – 1	$n \leq 24$	$n * (n-1)/2$
patient 1	6.29	4.58	0.14	15	105
patient 2	5.42	3.49	0.12	18	153
patient 3	6.50	3.68	0.25	14	91
patient 4	6.21	3.67	0.78	21	210

Table 1: Theoretical and observed values of the real validation index with bias for different patients. The number of registrations (which is also the number of values used to compute the statistics on the validation index) is directly linked to the number of images used. Results indicate a very good validation of the registration accuracy prediction: the mean validation index is within 10% of its theoretical value and the K-S test exhibits impressively high values.

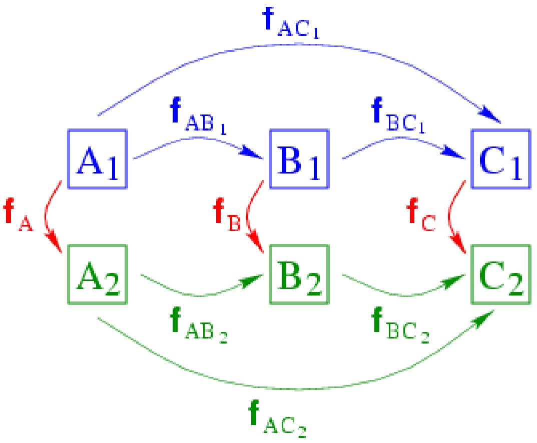

Figure 13: This diagram represents three acquisitions A, B, and C with the three echo-1 images (A_1, B_1, C_1) and the three echo-2 images (A_2, B_2, C_2). The echo-1 and echo-2 registrations are significantly different ($\mu^2(f_{AB_1}, f_{AB_2})$, $\mu^2(f_{AC_1}, f_{AC_2})$, $\mu^2(f_{BC_1}, f_{BC_2})$ ¿ 50) but the intra-echo-1 and intra-echo-2 registrations are compatible ($\mu^2(f_{BC_1} \circ f_{AB_1}, f_{AC_1}) \simeq 6$ and $\mu^2(f_{BC_2} \circ f_{AB_2}, f_{AC_2}) \simeq 6$). This led us to assume a global bias for each acquisition between echo-1 and echo-2 images, represented here by the transformations f_A, f_B, and f_C.

In this formula, each acquisition bias depends upon the others. Thus, we begin with null biases (identity transformations) and iteratively estimate each bias until convergence.

We effectively obtain a different bias for each acquisition that significantly differs from the identity. However, from a more global point of view, all the biases could be modeled as an "additional" noise on the transformation with an identity mean and standard deviations of $\sigma_r = 0.06$ deg on the rotation (not significantly different from 0) and $\sigma_x = 0.09$, $\sigma_y = 0.11$ and $\sigma_z = 0.13$ mm on the translation (significantly different from 0). Very similar values were observed for other patients.

3.3.3 Validation with Bias

Although the biases appear very small, they are sufficient to explain the previous errors in the registration accuracy prediction. Indeed, taking the biases into account, the real validation index between acquisition A and B becomes:

$$I_{AB} = \mu^2 \left(f_B \circ f_{AB_1} \, , \; f_{AB_2} \circ f_A \right).$$

Since the biases are estimated from the registration values, using their uncertainties in this formula would bias the validation index toward low values. Thus we consider them as deterministic. The mean value and standard deviation of this new index across all registrations are now very close to their theoretical value (see table 1).

3.3.4 Origin of the Bias

Most of the extremal points we match are situated on the surface of the brain and the ventricles. These surfaces appear differently in echo-1 and echo-2 images due to the difference in contrast. Other artifacts such as chemical shift or susceptibility effects (see for instance [21]) may also account for the observed bias as they influence the detection of extremal points. Indeed, the two echoes are acquired with different receiver RF bandwidth to improve the signal/noise ratio [22]. Therefore, the chemical shift and the susceptibility effect are different in the two echoes.

We plan to correlate the biases with diverse quantities in the images in order to understand their origin. Ultimately, we would like to predict the biases in each acquisition before registration. This would allow the definite validation of the registration accuracy prediction.

4 Conclusion

We presented in this chapter a method to extract reliable differential geometry features (crest lines and extremal points) from 3D images and several rigid registration algorithms to put into correspondence these features in two different images and to compute the rigid transformation between them. We also presented an analysis of the robustness with the computation of the probability (or mean number) of false positives and an analysis of the accuracy of the transformation.

This method proves to be very powerful for monomodal rigid registration of the same patient imaged at different times, as we show that an accuracy of less than a tenth of voxel can be achieved. In the last experiment, we showed that this uncertainty estimation technique is precise enough to put into evidence systematic biases on the order of 0.1 voxel between features in echo-1 and echo-2 images. Once corrected, multiple experiments on several patients show that our uncertainty prediction is validated on real data.

This registration technique is currently used in many medical applications, such as the registration of a preoperative MR used for planning and MR with a stereotactic frame for neuro-surgery navigation (European Community project Roboscope), or the registration of a series of acquisitions over time of images of Multiple Sclerosis patients to study the disease's evolution (European Community project Biomorph).

Several tracks have been followed to generalize this work to non-rigid registration. Feldmar [14] used the principal curvatures to register surfaces with rigid, affine, and locally affine transformations. Subsol designed an algorithm for non-rigid matching of crest lines. In [41], he used this method to warp 3D MR images of different patients' brains in the same coordinate system, and even to warp an atlas onto a given 3D MR image of a brain in order to segment it. In [42], he used the same method to construct automatically a morphometric atlas of the skull crest lines from several acquisitions of different patients' CT images, with applications in cranio-facial surgery.

Acknowledgments

The authors wish to thank the whole epidaure team and particularly their colleagues (or ex-colleagues) who worked on differential geometry features and registration: S. Benayoun, M. Fidrich, A. Gourdon, A. Guéziec, O. Monga, and G. Subsol. This research would have not been possible without the CT data from GE-CGR (Buc, France), the MR data of Dr. C. Guttmann and Prof. R. Kikinis (Brigham & Women's Hospital and Harvard Medical School, Boston, USA).

References

[1] Pierre Alliez, David Cohen-Steiner, Olivier Devillers, Bruno Levy, and Mathieu Desbrun. Anisotropic polygonal remeshing. *ACM Transactions on Graphics. Special issue for SIGGRAPH conference*, pages 485–493, 2003.

[2] E. V. Anoshkina, A. G. Belyaev, R. Huang, and T. L. Kunii. Ridges and Ravines on a Surface and Related Geometry of Skeletons, Caustics and Wavefronts. In R. A. Earnshaw and J. A. Vince, editors, *Computer Graphics: Developments in Virtual Environments*, chapter 22, pages 311–326. Academic Press, 1995.

[3] N. Ayache. Medical computer vision, virtual reality and robotics – promising research. *Image and Vision Computing*, 13(4):295–313, 1995.

[4] N. Ayache and O.D. Faugeras. Hyper: A new approach for the recognition and positioning of two-dimensional objects. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(1):44–54, 1986.

[5] A.G. Belyaev, E. V. Anoshkina, and T. L. Kunii K. *Topological Modeling for Visualization*, chapter Chap. 18: Ridges, ravines, and singularities. Springer, 1997.

[6] P.J. Besl and N. McKay. A method for registration of 3-d shapes. *PAMI*, 14(2):239–256, 1992.

[7] F. Cazals and M. Pouget. Differential topology and geometry of smooth embedded surfaces: selected topics. *Int. J. of Computational Geometry and Applications*, 15(5), 2005.

[8] Frédéric Cazals and Marc Pouget. Topology driven algorithms for ridge extraction on meshes. Research Report 5526, INRIA, 2005.

[9] C.B. Cutting. Applications of computer graphics to the evaluation and treatment of major craniofacial malformation. In Udupa and Herman, editors, *3-D Imaging in Medicine*. CRC Press, 1989.

[10] D. Dean, P. Buckley, F. Bookstein, J. Kamath, D. Kwon, L. Friedman, and C. Lys. Three dimensional MR-based morphometric comparison of schizophrenic and normal cerebral ventricles. In *Visualization in Biomedical Com-*

puting, volume 1131 of *LNCS*, pages 363–371. Springer Verlag, 1996.

[11] R. Deriche and G. Giraudon. A computational approach for corner and vertex detection. *IJCV*, 10(2):101–124, 1993.

[12] Manfredo P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.

[13] D. Eberly. *Ridges in Image and Data Analysis*. Kluwer Academic Publishers, 1996.

[14] J. Feldmar and N. Ayache. Rigid, affine and locally affine registration of free-form surfaces. *IJCV*, 18(2):99–119, May 1996.

[15] Márta Fidrich. *Multiscale analysis of invariants: Application to volumetric medical image processing*. PhD thesis, University Paris XI, July 1997.

[16] John M. Gauch. *Multiresolution Image Shape Description*. Springer-Verlag, 1992.

[17] A. Guéziec. Surface representation with deformable splines: Using decoupled variables. *IEEE Computational Science and Engineering Magazine*, 2(1):69–80, March 1995.

[18] A. Guéziec and N. Ayache. Smoothing and matching of 3d space curves. *International Journal of Computer Vision*, 12(1):79–104, 1994.

[19] A. Guéziec, X. Pennec, and N. Ayache. Medical image registration using geometric hashing. *IEEE Computational Science & Engineering, special issue on Geometric Hashing*, 4(4):29–41, 1997. Oct-Dec.

[20] D.P. Huttenlocher and S. Ullman. Object recognition using alignment. In *Proc. of ICCV*, pages 72–78, 1987.

[21] P. Jezzard. Physical basis of spatial distortions in magnetic resonance images. In *Handbook of Medical Image Processing*. Academic Press, 1999.

[22] R. Kikinis, C.R.G. Guttmann, D. Metcalf, W.M. III Wells, G.J. Ettinger, H.L. Weiner, and F.A. Jolesz. Quantitative follow-up of patients with multiple sclerosis using MRI: Technical aspects. *JMRI*, 9(4):519–530, 1999.

[23] Les Kitchen and Azriel Rosenfield. Gray-level corner detection. *Pattern Recognition Letters*, 1:95–102, 1982.

[24] J.J. Koenderink. *Solid Shape*. M.I.T. Press, 1990.

[25] 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.

[26] V.F. Leavers. Survey: which Hough transform? *CVGIP: Image Understanding*, 58(2):250–264, september 1993.

[27] S. Markatis. *Some Generic Phenomena in Families of Surfaces in R^3*. PhD thesis, University of Liverpool, 1980.

[28] O. Monga, N. Ayache, and P.T. Sander. From voxel to intrinsic surface features. *Image and Vision Computing*, 10(6):403–417, August 1992.

[29] O. Monga and S. Benayoun. Using Partial Derivatives of 3D Images to Extract Typical Surface Features. *Computer Vision and Image Understanding*, 61(2):171–189, March 1995.

[30] Alison J. Noble. Finding corners. *Image and Vision Computing*, 6:121–128, 1988.

[31] Yutaka Ohtake, Alexander Belyaev, and Ilia Bogaevski. Mesh regularization and adaptive smoothing. *Computer-Aided Design*, 33(11):789–800, 2001.

[32] Yutaka Ohtake, Alexander Belyaev, and Hans-Peter Seidel. Ridge-valley lines on meshes via implicit surface fitting. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 609–612, New York, NY, USA, 2004. ACM Press.

[33] X. Pennec. Toward a generic framework for recognition based on uncertain geometric features. *Videre: Journal of Computer Vision Research*, 1(2):58–87, 1998.

[34] X. Pennec and N. Ayache. Uniform distribution, distance and expectation problems for geometric features processing. *Journal of Mathematical Imaging and Vision*, 9(1):49–67, July 1998.

[35] X. Pennec, C.R.G. Guttmann, and J.P. Thirion. Feature-based registration of medical images: Estimation and validation of the pose accuracy. In *Proc. of First Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI'98)*, number 1496 in LNCS, pages 1107–1114, Cambridge, USA, October 1998. Springer Verlag.

[36] X. Pennec and J.P. Thirion. A framework for uncertainty and validation of 3D registration methods based on points and frames. *Int. Journal of Computer Vision*, 25(3):203–229, 1997.

[37] Xavier Pennec. Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements. *Journal of Mathematical Imaging and Vision*, 25(1):127–154, July 2006. A preliminary appeared as INRIA RR-5093, January 2004.

[38] I. R. Porteous. Ridges and Umbilics of Surfaces. In R. R. Martin, editor, *The mathematics of surfaces II*, pages 447–458. Clarendon Press - Oxford, 1987.

[39] I. R. Porteous. *Geometric Differentiation: For the Intelligence of Curves and Surfaces*. Cambridge University Press, 1994.

[40] Marc Pouget. *Geometry of surfaces: from the estimation of local differential quantities to the robust extraction of global differential features*. Thèse de doctorat en sciences, uni-ver-sité Nice Sophia-Antipolis, Nice, France, 2005.

[41] G. Subsol. Crest lines for curve based warping. In A.W. Toga, editor, *Brain Warping*, chapter 14, pages 241–262. Academic Press, 1998.

[42] G. Subsol, J. Ph. Thirion, and N. Ayache. A general scheme for automatically building 3D morphometric anatomical atlases: application to a skull atlas. *Medical Image Analysis*, 2(1):27–60, 1998.

[43] J-P Thirion. New feature points based on geometric invariants for 3D image registration. *International Journal of Computer Vision*, 18(2):121–137, May 1996.

[44] J-P Thirion and A Gourdon. Computing the differential characteristics of isointensity surfaces. *Computer Vision and Image Understanding*, 61(2):190–202, March 1995.

[45] J-P. Thirion and A. Gourdon. The 3D marching lines algorithm. *Graphical Models and Image Processing*, 58(6):503–509, 1996.

[46] P. A. van den Elsen, J. B. A. Maintz, E. J. D. Pol, and M. A. Viergever. Automatic Registration of CT and MR Brain Images Using Correlation of Geometrical Features. *IEEE Transactions on Medical Images*, 14(2):384–398, 1995.

[47] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *Int. Journ. Comp. Vis.*, 13(2):119–152, 1994. Also Research Report No.1658, INRIA Sophia Antipolis, 1992.