Registration of a curve on a surface using differential properties

Alexis Gourdon, Nicholas Ayache

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

REGISTRATION OF A CURVE ON A SURFACE USING DIFFERENTIAL PROPERTIES

Alexis GOURDON & Nicholas AYACHE

N° 2145

Decembre 1993

PROGRAMME 4

Robotique,
image
et vision

ISSN 0249-6399

INRIA
SOPHIA ANTIPOLIS

REGISTRATION OF A CURVE ON A SURFACE USING DIFFERENTIAL PROPERTIES

Alexis GOURDON & Nicholas AYACHE

Programme 4 — Robotique, image et vision
Projet Epidaure

Rapport de recherche n ° 2145 — Decembre 1993 — 32 pages

Abstract: This article presents a new method to find the best spatial registration between a rigid curve and a rigid surface. We show how to locally exploit the knowledge of differential properties computed on both the curve and the surface to constrain the rigid matching problem. It is in fact possible to write a compatibility equation between a curve point and a surface point, which constrains completely the 6 parameters of the sought rigid displacement. This requires the local computation of third order differential quantities and leads to an algebraic equation of degree 16. A second approach consists in considering pairs of curve and surface points. It is then possible to use only first order differential constraints to compute locally the parameters of the rigid displacement. Each approach leads to a different matching algorithm. Although computationally more expensive, the second approach is more robust, and can be accelerated with a preprocessing of the surface data. The paper presents the mathematical details of both approaches, algorithms, and a preliminary experimental study on both synthetic and real 3D medical data. To our knowledge, it is the first method which takes full advantage of local differential computations to register a curve on a surface.

Key-words: Curve-Surface Registration, 3D rigid matching, Differential Geometry.

(Résumé : tsvp)

Unité de recherche INRIA Sophia-Antipolis
2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex (France)
Téléphone : (33) 93 65 77 77 – Télécopie : (33) 93 65 77 65

RECALAGE D'UNE COURBE SUR UNE SURFACE EN UTILISANT DES PROPRIETES DIFFERENTIELLES

Résumé : Cet article présente une nouvelle approche pour effectuer le recalage rigide d'une courbe sur une surface. Les caractéristiques différentielles calculées sur la courbe et la surface permettent de contraindre les 6 paramètres du recalage rigide. Il est en effet possible d'écrire une équation de compatibilité entre un point d'une courbe et le point homologue de la surface. Cela nécessite le calcul de dérivées jusqu'à l'ordre 3 et conduit à une équation algébrique de degré 16. Une deuxième approche consiste à utiliser des couples de points de la courbe et de la surface. Alors les caractéristiques différentielles du premier ordre suffisent à estimer localement le déplacement rigide. Bien que plus gourmande en temps de calcul, cette approche est plus robuste, et peut être accélerée par un prétraitement de la surface.

Après avoir démontré les contraintes géométriques du problème, nous présenterons les deux types d'algorithmes ainsi que des résultats obtenus sur des données synthétiques et réelles. A notre connaissance, il s'agit de la première méthode permettant de recaler automatiquement une courbe sur une surface en tenant compte de propriétés différentielles.

Mots-clé : Recalage d'une Courbe sur une Surface, Recalage 3D, Géométrie Différentielle.

1 Introduction

This article presents a new method to find the best spatial registration between a rigid curve and a rigid surface (see figure 1).

This is an important problem in the medical field [1] when a volume medical image must be registered either with a single cross-section acquired later with a CT-Scanner or MRI, or with a 3D curve acquired with a laser range finder on the external surface of a patient. This is also an important problem in several industrial and navigation applications where laser range finders are used to locate previously stored 3D models.

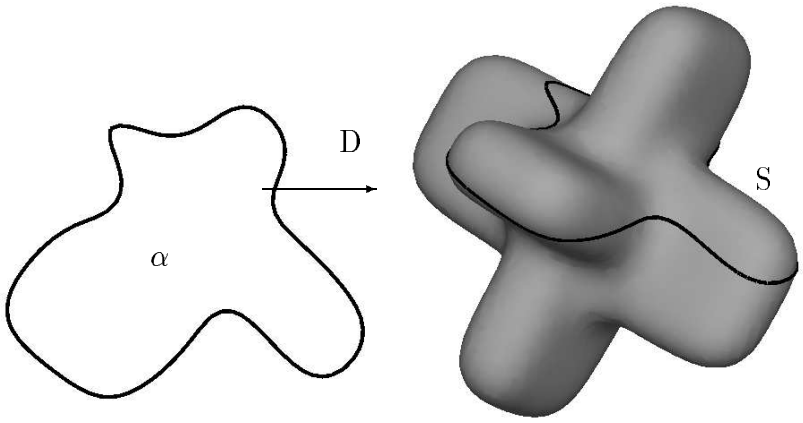

Figure 1: Registration of a curve on a surface

Besl mentionned already this problem in a remarkable paper on 3D registration [3], and Grimson et al presented recently an industrial application of this type but restricted to the recognition of cylinder objects [7].

The scope of our study is more general in that we do not restrict the shape of the observed surfaces or curves. In fact, although results are presented mainly with planar curves, the developped formalism is valid also for general spatial curves and free form surfaces. On the other hand, we assume that it is possible to compute the differential properties of both the curve and the surface, either up to the third order (first approach), or at least up to the first order (second approach). Both are reasonable assumptions with high

resolution medical volume images, where adequate spatial filtering allows for the extraction of anatomical surfaces and curves, with the computation of accurate differential properties [13], [14], [22] [19].

Although none of the following references tackled directly the problem exposed here, most of them have inspired our work. These are the studies on rigid shape matching [6], [8], [9], [21], [17], [2], [10], [11], [12], studies on the algebraic equations resulting from the matching of implicit curves and surfaces [20], and studies on the use of semi-differential invariants for projective matching [15].

Our paper is organized as follows :

- In section 2, we look at the geometrical constraints which apply to a curve point lying on a surface and we prove that we have to compute derivatives up to the third order to constrain completely all six parameters of a rigid transformation. We derive a practical matching algorithm using this approach.
- In section 3, we show that only first order derivatives are necessary when considering pairs of points on both the curve and the surface, and we propose another matching algorithm.
- Finally In section 4, we present experimental results on both synthetic and real data, which confirm the validity of the approach, and propose future extensions to this work.

2 Registration using only one point on curve and surface

2.1 Geometrical Constraints

At each point of a parametric curve, we can define an intrinsic orthonormal frame $(\mathbf{t}, \mathbf{n}, \mathbf{b})$ (the Frenet frame) and metric invariants (the curvature k and torsion τ) [5][16].

We can also build at each point of a parametric surface the two fundamental forms [5][16] (see appendix B) and infer from them an intrinsic orthonormal frame $(\mathbf{e_1}, \mathbf{e_2}, \mathbf{N})$ (the principal frame) and the two principal curvature (k_1, k_2)) [5][16](or see appendices B C).

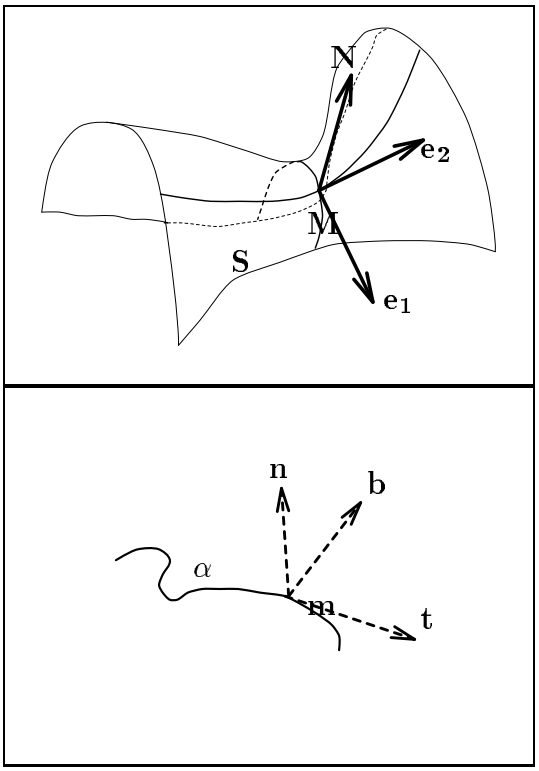

Figure 2: Up: the surface principal frame, down : the curve Frenet frame

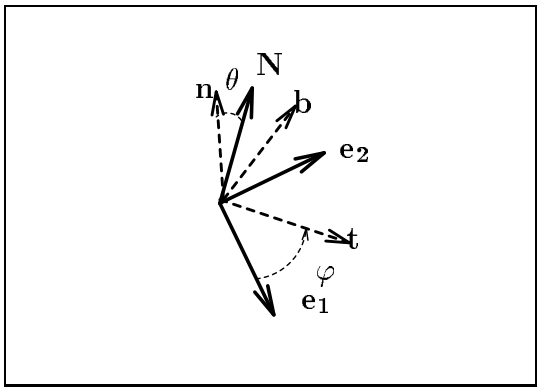

Figure 3: the angles between the principal frame and the Frenet frame

When a curve lies on a surface, its tangent vector is in the tangent plane of the surface. We can then also construct a third intrinsic orthonormal frame $(\mathbf{t}, \mathbf{g}, \mathbf{N})$, called the Darboux frame.

Then (see appendices) we can write the variation of these frames along the curve with respect to the arc length s . The relation between two orthonormal frames $\mathbf{x}$, $\mathbf{y}$ can be described by $\mathbf{x} = U\mathbf{y}$ with U a rotation matrix. Here $\mathbf{x} = (\mathbf{t}, \mathbf{g}, \mathbf{N})^T$ is the Darboux frame, $\mathbf{y} = (\mathbf{t}, \mathbf{n}, \mathbf{b})^T$ or $\mathbf{y} = (\mathbf{e_1}, \mathbf{e_2}, \mathbf{N})^T$ are the Frenet frame or the principal frame. So, if one uses the moving frame method (see appendix A), one can find $d\mathbf{x}/ds = \Omega_x \mathbf{x}$, $d\mathbf{y}/ds = \Omega_y \mathbf{y}$ then $d\mathbf{x}/ds = (dU/ds U^T + U\Omega_y U^T)\mathbf{x}$ which can be written :

$$\Omega_x = \frac{dU}{ds} U^T + U\Omega_y U^T \tag{1}$$

Using the relations found in appendix C expressing the variation of the Darboux frame, the Frenet frame and the principal with respect to the arc-length s frame we obtain the three fondamental relations :

$$\begin{cases} k_n &= k\cos\theta \\ k_n &= k_1\cos^2\varphi + k_2\sin^2\varphi \end{cases} \tag{2}$$

$$\begin{cases} k_g &= k\sin\theta \\ k_g &= \dfrac{1}{k_1 - k_2}\left(\dfrac{\partial k_1}{\partial e_2}\cos\varphi + \dfrac{\partial k_2}{\partial e_1}\sin\varphi\right) + \dfrac{d\varphi}{ds} \end{cases} \tag{3}$$

$$\left\{ \begin{array}{lcl} \tau_g & = & \tau - \dfrac{d\theta}{ds} \\ \tau_g & = & (k_1 - k_2)\cos\varphi \sin\varphi \end{array} \right. \tag{4}$$

(k_n, k_g, τ_g) are respectively the normal curvature, the geodesic curvature and the geodesic torsion.

The knowledge of θ the angle betwen the normal to the curve and the normal to the surface and φ the angle between the curve tangent and the principal direction $\mathbf{e_1}$ (see figure 3) characterizes the rotation between the two intrinsic frames (the Frenet frame for the curve and the principal frame for the surface).

For each point on the curve, we can compute its curvature k and its torsion τ and, if necessary, their derivatives with respect to the arc length s.

Moreover, for each point of the surface the principal curvatures (k_1, k_2) and their derivatives along $\mathbf{e_1}$ and $\mathbf{e_2}$ can be computed. However k_n, k_g, τ_g and the two angles θ, φ are unknown and so are their derivates with respect to s. We have three equations and two unknown quantities $(\theta(s), \varphi(s))$ and the problem is therefore over-constrained.

In our problem, we have a model S of a surface and a curve α that we wish to register on S. For every point m of α and its homologous point M on the surface, every pair (θ, φ) which is solution of equations (2,3,4) gives us a unique registered Frenet frame. It is then straigthforward to find the rotation of the Frenet frame and to infer the rigid transformation which maps m into M (see figures 2,3).

2.2 Simplification of the equations

The major drawback of the system (2,3,4) where the unknown values are $(\theta(s), \varphi(s))$ is the presence of the derivatives of $(\theta(s), \varphi(s))$ with respect to s (except for equation 2). Thus, in order to find (θ, φ) we must eliminate those derivatives. Setting $H = \dfrac{k_1 + k_2}{2}$ (mean curvature) and $J = \dfrac{k_1 - k_2}{2}$ the system (2,3,4) becomes :

$$\left\{ \begin{array}{lcccl} k_n & = & k\cos\theta & = & J\cos 2\varphi + H \\ k_g & = & k\sin\theta & = & \dfrac{1}{2J}(\dfrac{\partial k_1}{\partial e_2}\cos\varphi + \dfrac{\partial k_2}{\partial e_1}\sin\varphi) + \dfrac{d\varphi}{ds} \\ \tau_g & = & \tau - \dfrac{d\theta}{ds} & = & J\sin 2\varphi \end{array} \right.$$

Then if we rewrite $\frac{dk_n}{ds}$ in two different ways we obtain :

$$\frac{1}{k}\frac{dk}{ds}k_n + k_g(3\tau_g - \tau) = \frac{dH}{ds} + \frac{dJ}{ds}\cos 2\varphi + (\frac{\partial k_1}{\partial e_2}\cos\varphi + \frac{\partial k_2}{\partial e_1}\sin\varphi)\sin 2\varphi \quad (5)$$

which can be written :

$$\frac{1}{k}\frac{dk}{ds}k_n + k_g(3\tau_g - \tau) = C(\varphi) \quad (6)$$

with $C(\varphi)$ given by :

$$\begin{aligned} C(\varphi) &= \frac{3}{2}\frac{\partial H}{\partial e_1}\cos\varphi + \frac{3}{2}\frac{\partial H}{\partial e_2}\sin\varphi + \frac{1}{4}(\frac{\partial k_1}{\partial e_1} - 3\frac{\partial k_2}{\partial e_1})\cos 3\varphi + \frac{1}{4}(3\frac{\partial k_1}{\partial e_2} - \frac{\partial k_2}{\partial e_2})\sin 3\varphi \\ &= 3\frac{\partial k_2}{\partial e_1}\cos\varphi + 3\frac{\partial k_1}{\partial e_2}\sin\varphi + (\frac{\partial k_1}{\partial e_1} - 3\frac{\partial k_2}{\partial e_1})\cos^3\varphi + (\frac{\partial k_2}{\partial e_2} - 3\frac{\partial k_1}{\partial e_2})\sin^3\varphi \\ &= \frac{\partial k_1}{\partial e_1}\cos^3\varphi + 3\frac{\partial k_1}{\partial e_2}\cos^2\varphi\sin\varphi + 3\frac{\partial k_2}{\partial e_1}\cos\varphi\sin^2\varphi + \frac{\partial k_2}{\partial e_2}\sin^3\varphi \quad (7) \end{aligned}$$

Therefore we obtain one single equation in φ of the form $f(\varphi) = 0$ with

$$f(\varphi) = (k^2 - k_n^2(\varphi))(3\tau_g(\varphi) - \tau)^2 - (C(\varphi) - \frac{1}{k}\frac{dk}{ds}k_n(\varphi))^2 \quad (8)$$

This equation is an algebraic equation of degree sixteen. It can be easily shown by using the relations $\cos\varphi = \frac{e^{i\varphi} + e^{-i\varphi}}{2}$, $\sin\varphi = \frac{e^{i\varphi} - e^{-i\varphi}}{2i}$ and multiplying by $e^{i8\varphi}$, the equation has terms of the form $e^{ik\varphi}$ with $k \in \{0, 1....16\}$. As the degree is equal to sixteen, we have at most 16 roots for the angle φ. For each solution φ, the first equation of the system (2) gives us $\cos\theta$; $\sin\theta$ is obtained by the equations (3) and (6) .

2.3 Computation of the differential invariants

A volume image can be seen as a continuous fonction $I(x, y, z)$ of the space $\mathbb{R}^3$. For any value a (the iso-value) the iso-surface is the set which separates the regions of the space where $I \geq a$ from the regions where $I < a$. With respect to this definition the iso-surface has good topological properties : it is a continuous, not self-intersecting surface with no holes. Then the use of a

Gaussian convolution filter transforms the digital 3D image into an infinitely differentiable function $I(x, y, z)$ [13].

In fact, we do not need a parametric representation of the surface. We can use the implicit function theorem in order to transform the implicit equation of the iso-surface $I(x, y, z) = a$ (where the gradient of I does not vanish) into a parametric form $(x = u, y = v, z = \phi_z(u, v))$. We can then compute the two fondamental forms, the principal curvatures and the principal directions as expressions of the derivatives of $I(x, y, z)$ [19] [22].

For a planar image, we can also write differential caracteristics of an iso-contour as expressions of the derivatives of the image function.

2.4 Determining the angle φ

Since equation (8) wich gives the angle φ is an algebraic equation of degree sixteen , there is no hope to find explicit roots for the general case. Classical methods can be used for equations $F(x) = 0$ like a bissection, the secant method or the Newton-Raphson method using the function derivatives [18].The main disadvantage of the Newton method if used alone is that the convergence speed can be slow if the derivative is small in a neighborhood of a root. On the contrary with a bissection , after each iteration the bounds containing the root decrease by a factor 2. Nevertheless one can miss some roots if the initial brackets contain more than one solution.

2.5 Matching Algorithm

Now that we can compute the differential invariants and express the geometric constraints for a point on a curve and its homologous point of the surface, let us describe the matching algorithm. First of all, we can notice that since our sets can handle as many points as $p = 200$ points for the curve and $n = 5000$ points for the surface, it follows that as many as one million possible matching pairs may have to be tested. (It is a $O(np)$ algorithm).
Therefore one can limit this huge amount of points by applying one of these two strategies :

- If we take the elliptic points on the surface, then we have $|k| > min(|k_1|, |k_2|)$. Thus if we remove points of the surface with minimal curvature below a certain value, we can also remove points on the curve.

- The alternative is : if a point of the curve has a small curvature, it must correspond to elliptic points such as $min(|k_1|, |k_2|) < |k|$ or hyperbolic points.

Once the number of points has been reduced by one of the above mentionned strategies, our matching algorithm is the following :

- For each remaining pair of points with one point on the curve and the other one on the surface, solve the equation for the angle φ by a bissection for example. Then for each root :
  - Compute θ and determine the rigid transformation.
  - Apply the tranformation to the curve.
  - If a sufficient number of curve points lie at a distance of the surface smaller than a preset threshold, store the 6 parameters of the rigid transformation in a hash-table. (one can also improve the transformation by using the iterative closest point method [4])
- Repeat until at least one bucket of the hash-table contains a sufficient number of stored transfomations (expressed as a percentage of the number of curve points).

This algorithm uses the estimation of the angle φ to predict transformations that register locally the curve on the surface. Then, by applying the transformation to the curve we can easily eliminate irrelevant transformations. However this last process requires the computation, for each point of the curve, of the closest point on the surface. As we have a 3-D image, we can compute for each voxel the closest point with respect to the iso-surface [11] and store it in a 3D distance image. In the general case where the surface is given by a list of points, we can use other methods such as an octree-spline distance map [10]. As soon as the initial guess is reasonably good, we can improve the registration by using an iterative method (the iterative closest point algorithm [4]):

- For each point of the curve at iteration n, compute the closest point on the surface.
- Compute the best fit between the curve at iteration n and their corresponding closest point on the surface with a quaternion-based algorithm.

- Apply this transformation to the curve at iteration n to obtain the curve at iteration $n+1$.
- Terminate the iteration when a given percentage of the curve points falls at a distance below a preset threshod specifying the desired precision of the registration, or when the number of iterations is too large.

This iterative method gives us better fits than the initial transformation and converges towards a local minimum [4]. Unfortunately, to find the initial guess, we have to compute third order derivatives on the curve and on the surface and solve an algebraic equation of high degree. This makes the method applicable to high quality data only, as experiments will show. When the noise level increases, a more expensive but more robust approach is now presented.

3 Registration using a pair of points on curve and surface

3.1 Geometrical Constraints

The geometric constraints between a pair of points (a, b) on the curve α and its homologous pair of points (A, B) on the surface S require only first order differential invariants on the curve and surface.

Since the tangent to a curve lying on a surface is in the tangent plane of the surface and that the scalar product is invariant by a rigid transformation, the constraints between the pairs (a, b) et (A, B) can be written as :

$$\left\{\begin{array}{rcl} \| \mathbf{t_a} \| = \| \mathbf{N_A} \| & = & 1 \\ \| \mathbf{t_b} \| = \| \mathbf{N_B} \| & = & 1 \\ \| \mathbf{ab} \| = \| \mathbf{AB} \| & = & d \\ < D(\mathbf{t_a}) | \mathbf{N_A} > {}^{1} & = & < D(\mathbf{t_b}) | \mathbf{N_B} > = \mathbf{0} \\ < D(\mathbf{t_a}) | \mathbf{AB} > & = & < \mathbf{t_a} | \mathbf{ab} > \\ < D(\mathbf{t_b}) | \mathbf{AB} > & = & < \mathbf{t_b} | \mathbf{ab} > \end{array}\right.$$

D is the unknown displacement, $(\mathbf{t_a}, \mathbf{t_b})$ are the tangents to the curve in (a, b) and $(\mathbf{N_A}, \mathbf{N_B})$ are the normals to the surface in (A, B) homlogous

[1] $< \mathbf{X} | \mathbf{Y} >$ is the scalar product of $\mathbf{X}$ and $\mathbf{Y}$

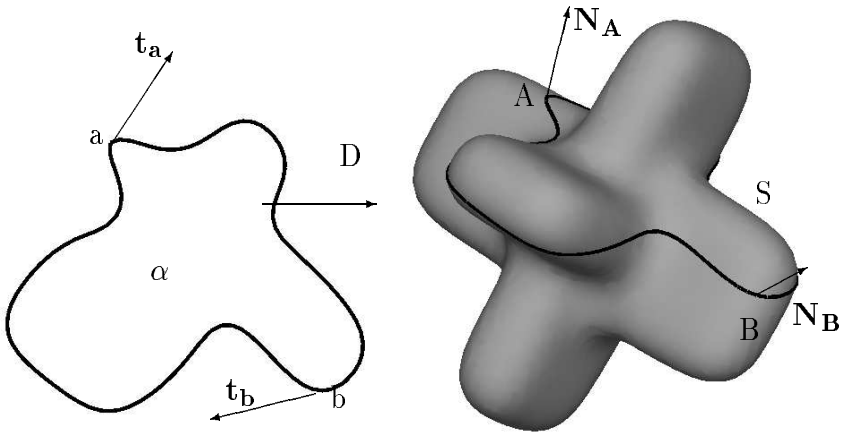

Figure 4: Registration using a pair of points on the curve and surface

points of (a, b) by D (see figure 4). Writing :

$$\begin{cases} < \mathbf{N_A} | \mathbf{N_B} > = \cos \mathbf{\Gamma} \\ < D(\mathbf{t_a}) | D(\mathbf{t_b}) > = < \mathbf{t_a} | \mathbf{t_b} > = \cos \gamma \end{cases}$$

and by expressing $(D(\mathbf{t_a}), D(\mathbf{t_b}))$ in the bases of the tangent planes of the surface in (A, B) [2] we have :

$$\begin{cases} \sin \Gamma \, D(\mathbf{t_a}) = \cos \alpha \, \mathbf{N_A} \wedge \mathbf{N_B} + \sin \alpha \, [\cos \Gamma \mathbf{N_A} - \mathbf{N_B}] \\ \sin \Gamma \, D(\mathbf{t_b}) = \cos \beta \, \mathbf{N_A} \wedge \mathbf{N_B} + \sin \beta \, [\cos \Gamma \mathbf{N_B} - \mathbf{N_A}] \end{cases}$$

(α, β) are the angles between $(D(\mathbf{t_a}), D(\mathbf{t_b}))$ and $\mathbf{N_A} \wedge \mathbf{N_B}$ which is a vector of the intersection of the tangent planes of the surface in (A, B) (see figure 5).

As the scalar product is invariant we can write : $< D(\mathbf{t_a}) | D(\mathbf{t_b}) > = < \mathbf{t_a} | \mathbf{t_b} > = \cos \gamma$. Thus we have one equation between the two unknown angles (α, β) and (γ, Γ) :

$$\cos \alpha \cos \beta - \cos \Gamma \sin \alpha \sin \beta = \cos \gamma \tag{9}$$

[2] when $\mathbf{N_A} \wedge \mathbf{N_B} \neq 0$

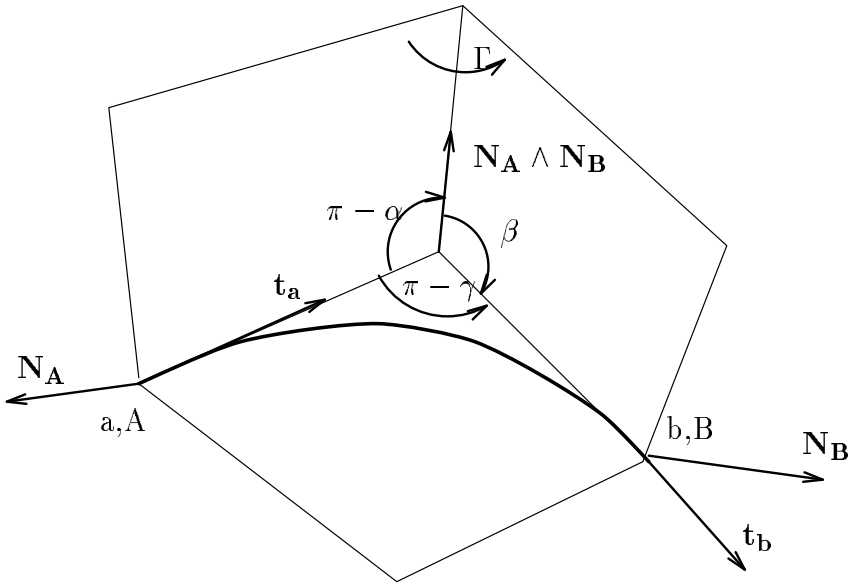

Figure 5: geometrical constraints with pair of points

The expressions of $\mathbf{AB}$ in the two bases gives us :

$$\begin{cases} \sin\Gamma\,\mathbf{AB} = \lambda\,\mathbf{N_A}\wedge\mathbf{N_B} + \mu_A\,[\cos\Gamma\,\mathbf{N_A} - \mathbf{N_B}] + \nu_A\,\sin\Gamma\mathbf{N_A} \\ \sin\Gamma\,\mathbf{AB} = \lambda\,\mathbf{N_A}\wedge\mathbf{N_B} + \mu_B\,[\cos\Gamma\,\mathbf{N_B} - \mathbf{N_A}] + \nu_B\,\sin\Gamma\mathbf{N_B} \end{cases}$$

By computing the scalar products $< D(\mathbf{t_a})|\mathbf{AB} >$ and $< D(\mathbf{t_b})|\mathbf{AB} >$ we get :

$$\begin{cases} < D(\mathbf{t_a})|\mathbf{AB} > = \lambda\,\cos\alpha + \mu_A\,\sin\alpha = < \mathbf{t_a}|\mathbf{ab} > \\ < D(\mathbf{t_b})|\mathbf{AB} > = \lambda\,\cos\beta + \mu_B\,\sin\beta = < \mathbf{t_b}|\mathbf{ab} > \end{cases} \tag{10}$$

with :

$$\begin{cases} \sin\Gamma\,\lambda &= < \mathbf{AB}|\mathbf{N_A}\wedge\mathbf{N_B} > \\ \sin\Gamma\,\mu_A &= < \mathbf{AB}|\cos\Gamma\,\mathbf{N_A} - \mathbf{N_B} > \\ \sin\Gamma\,\mu_B &= < \mathbf{AB}|\cos\Gamma\,\mathbf{N_B} - \mathbf{N_A} > \end{cases}$$

This gives us two more constraints for the angles (α, β). An equation of the form $A\cos\theta + B\sin\theta = C$ have roots if only $A^2 + B^2 \geq C^2$, the roots are given by :

$$\begin{cases} \theta = \psi + \delta\ [2\pi] \\ \theta = \psi - \delta\ [2\pi] \end{cases}$$

with $\cos\psi = A/\sqrt{A^2+B^2}$, $\sin\psi = B/\sqrt{A^2+B^2}$ and $\cos\delta = C/\sqrt{A^2+B^2}$. Hence we have solutions if :

$$\begin{cases} {\lambda^2 + \mu_A}^2 &= \mathbf{AB}^2 - < \mathbf{AB}|\mathbf{N_A} >^2 &\geq < \mathbf{ab}|\mathbf{t_a} >^2 \\ {\lambda^2 + \mu_B}^2 &= \mathbf{AB}^2 - < \mathbf{AB}|\mathbf{N_B} >^2 &\geq < \mathbf{ab}|\mathbf{t_b} >^2 \end{cases}$$

As soon as we have these two constraints we can solve the equations 10 verified by α and β and keep the roots which verify (9) whose constraints are :

$$\begin{cases} \sin^2\alpha &\leq (\dfrac{sin\gamma}{sin\Gamma})^2 \\ \sin^2\beta &\leq (\dfrac{sin\gamma}{sin\Gamma})^2 \end{cases}$$

Since we have at most two possible roots for α and β, each pair (a, b) and (A, B) gives us at most four rigid transformations.

3.2 Specific Situations

If $\mathbf{N_A} \wedge \mathbf{N_B} = 0$, then we have :

$$\begin{cases} D(\mathbf{t_a}) = \cos\alpha\,\mathbf{I} + \sin\alpha\,\mathbf{J} \\ D(\mathbf{t_b}) = \cos\beta\,\mathbf{I} + \sin\beta\,\mathbf{J} \\ \mathbf{AB} = \lambda\,\mathbf{I} + \mu\,\mathbf{J} + \nu\,\mathbf{N_A} \end{cases}$$

$(\mathbf{I}, \mathbf{J}, \mathbf{N_A})$ is an orthonormal basis. Then, we can find α and β with the same method as in the general case.

A very insteresting case happens when $\sin\gamma = 0$, (i.e. when the points (a, b) have parallel tangent vectors) [3]. Moreover if $sin\Gamma \neq 0$, by using the constraints of equation (9) we prove that (α, β) must be equal to 0 or π and that $(\mathbf{N_A} \wedge \mathbf{N_B}, \mathbf{t_a}, \mathbf{t_b})$ are proportional vectors. This can be seen in the following equation :

$$\lambda^2 = (\frac{< \mathbf{N_A} \wedge \mathbf{N_B} | \mathbf{AB} >}{\sin\Gamma})^2 =< \mathbf{ab} | \mathbf{t_a} >^2 =< \mathbf{ab} | \mathbf{t_b} >^2 \qquad (11)$$

Thus in this particular case we have two invariants on the pair of curve points and on the pair of surface points : the intrinsic distance between the two points d and the quantity λ.

3.3 Registration algorithm

Using the above-mentioned constraints, the matching algorithm is as follows:

- for each pair of curve points and for each pair of surface points being at the same distance d :
  - Solve equations (10) for α and β.
  - Keep the pairs of solutions (α, β) which satisfy the compatibility equation (9) with a given accuracy ϵ.
  - Compute for each pair of solutions (α, β) the corresponding rigid transformation.

[3] This situation occurs at least once for each point of a planar and closed differentiable smooth curve.

  - Apply this rigid transformation to the curve to verify that a sufficient number of curve points are within a tolerance threshold on the surface (this can be refined by the application of the iterative verification algorithm previously described [4].
  - If the verification stage succeeds, accumulate the rigid transformation parameters in a hash table.

- Repeat this process until a sufficient number of rigid transformations are accumulated in the same bucket (this number is expressed as a percentage of the number of curve points).

The main advantage of this algorithm with respect to the one based on the computation of φ comes from the use of first order differential invariants (curve tangents and surface normals), instead of the third order differential invariants.

On the negative side, this algorithm has a much higher algorithmic complexity. In effect, it is necessary to search for pairs of curve and scene points being at a similar distance of each other. A brute force algorithm leads to a combinatorial explosion, because it involves the comparison of pairs of points both on the curve and the surface and is in $O(n^2p^2)$).

A way to reduce the algorithmic complexity consists in reducing the number of pairs of curve points, by selecting those being at a sufficiently large distance of each other. This distance can be expressed as a large percentage (e.g. 75%) of the curve diameter. The surface can be preprocessed beforehand : pairs of surface points can be ordered by increasing distance d. Then, at recognition time, a given pair of curve points is compared to surface pairs of similar distance with a $O(\log(n^2))$ algorithm .

A further reduction in the number of such pairs can be easily obtained when dealing with planar and closed curves (wich is a common situation in the considered applications), by imposing parallel tangent vectors. In this case as we showed in the previous sub-section, it is easy to compute 2 intrinsic invariants at each curve and scene point (d and λ computed with respect to curve point of same tangent vector). The number of pairs of curve points then reduces to $O(p)$, since for a given curve point there is (in general) a finite number of points on the curve with parallel tangents.

Applying both strategies can reduce the recognition complexity to $O(p\ \log(n))$ for planar curves, and $O(p^2\ \log(n))$ in the general case.

4 Experimental results

4.1 Synthetic examples

After the generation of synthetic volume images of a 3D object (cf. figures 6 7) we extracted the object surface with the Marching Cube Algorithm [22] for a given iso-intensity value, as well as the curve in a randomly selected 2D cross-section, for the same iso-intensity value. The surface had about 7000 points while the curve had about 100 points.

Using the first approach (single points on curve and surface), and using the proposed third order differential invariants, equation (8) allowing the computation of φ revealed itself quite unstable, yielding from 0 to 12 solutions. Only in a few cases, the program returned among several solutions the correct one. This proved the validity of the code and the instability of this first approach. At the time of the conference, we shall be able to give an estimation of the required accuracy of the original data, to compute more reliably the rigid transformation.

On the other hand, the second approach (pairs of curve and surface points) yielded a correct rigid transformation, for all our trials. In fact, this second approach also yielded additional solutions, which are quite reasonable due to the symmetries of the original object as can be seen in figures 6 7.

4.2 Real Data

We used a volume image of a skull acquired by an X-ray CT-Scan[4], and we computed the iso-surface corresponding to the bone surface with the Marching Cube algorithm. In another volume image of the same skull but in a different position, we extracted a single cross section and the the iso-intensity contour corresponding again to the bone limit (see figures 8).

The algorithm using the second approach found the rigid transformation which superposes the extracted curve on the extracted surface. As we knew that the cross section had been extracted grossly at the level of the orbits, we reduced the complexity of the matching algorithm by selecting only about 10 cross-sections of the first volume image of the skull, centered about the orbits and by taking the pairs of curve points whose inter-distance was larger than 75% of the curve diameter.

[4] the images were provided by GE-CGR

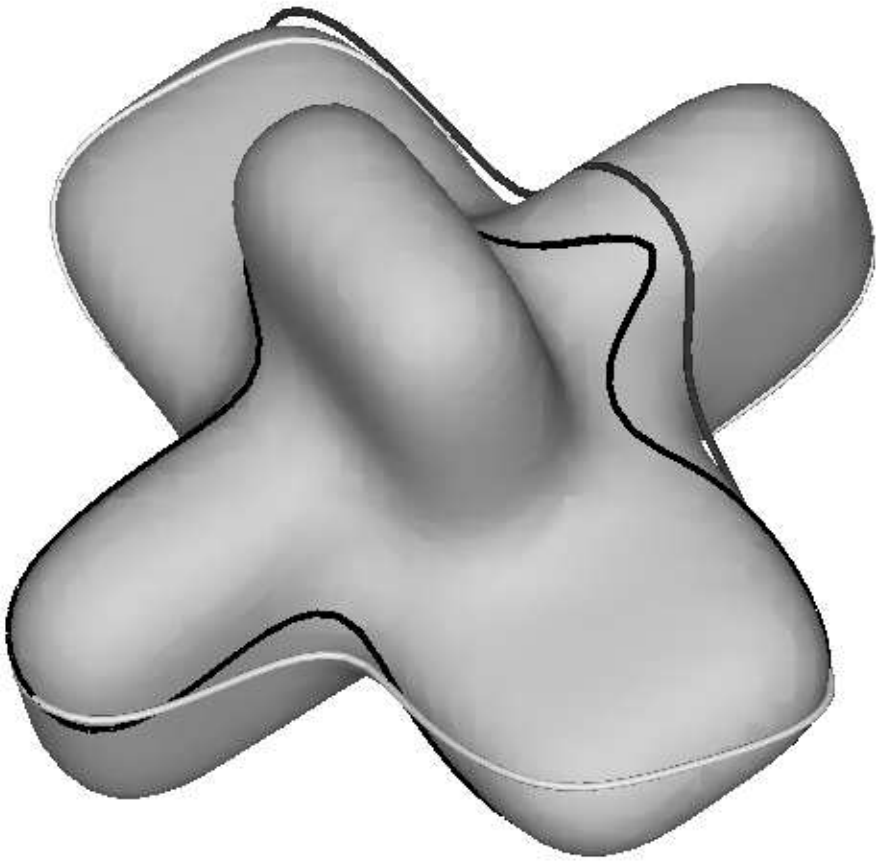

Figure 6: registration on a synthetic example

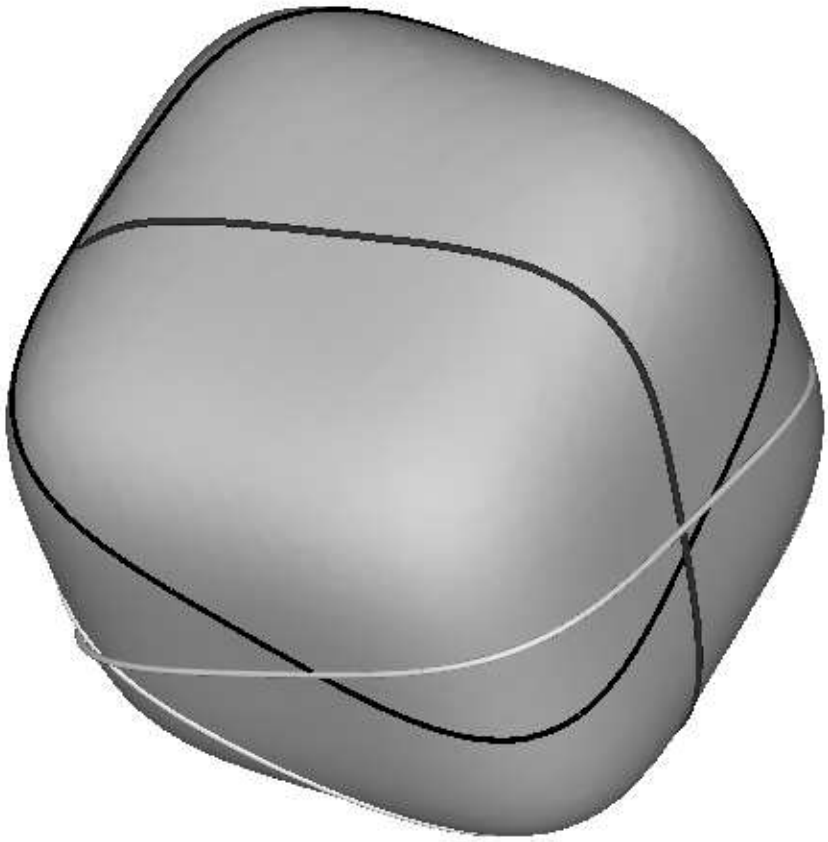

Figure 7: registration on a other synthetic example

Starting with the found solution, the iterative registration algorithm improves the solution. Typically, the initial solution is found with a tolerated distance of about 5 voxels between the transformed curve and the surface, and this distance can be decreased to 0.5 voxel after a few tens of iterations of the iterative registration algorithm.

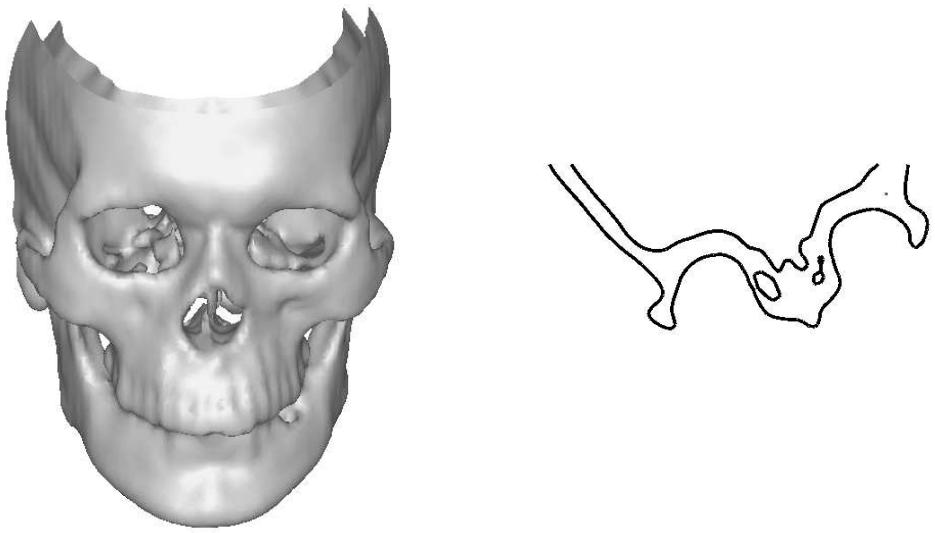

Figure 8: Left : surface from the first CT scan, right : curve from the second CT scan

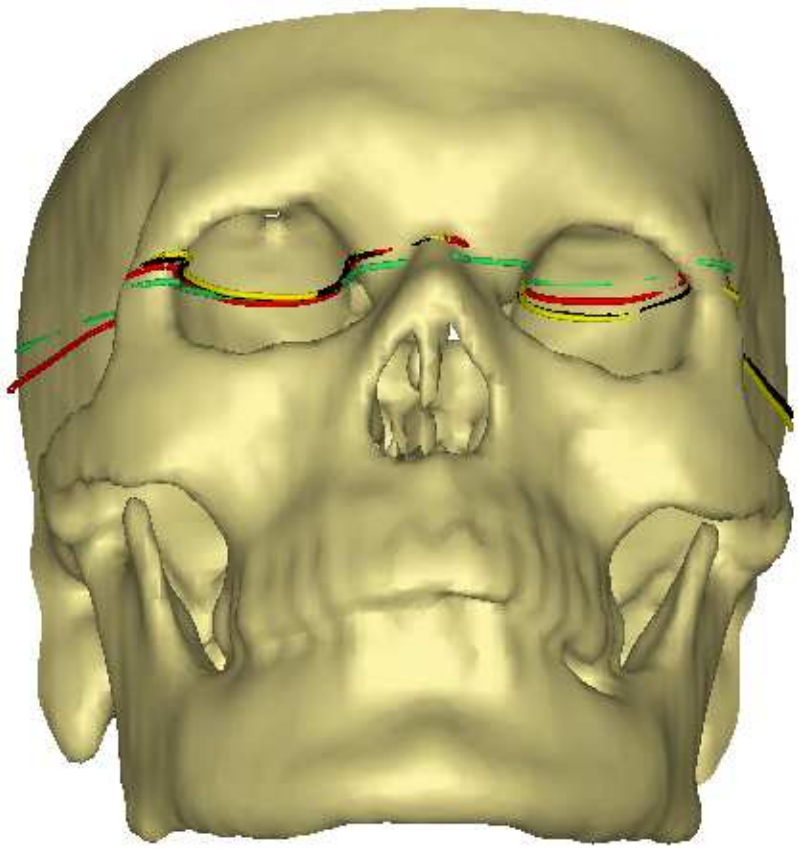

Figure 9: four solutions found with the algorithm with pairs of points

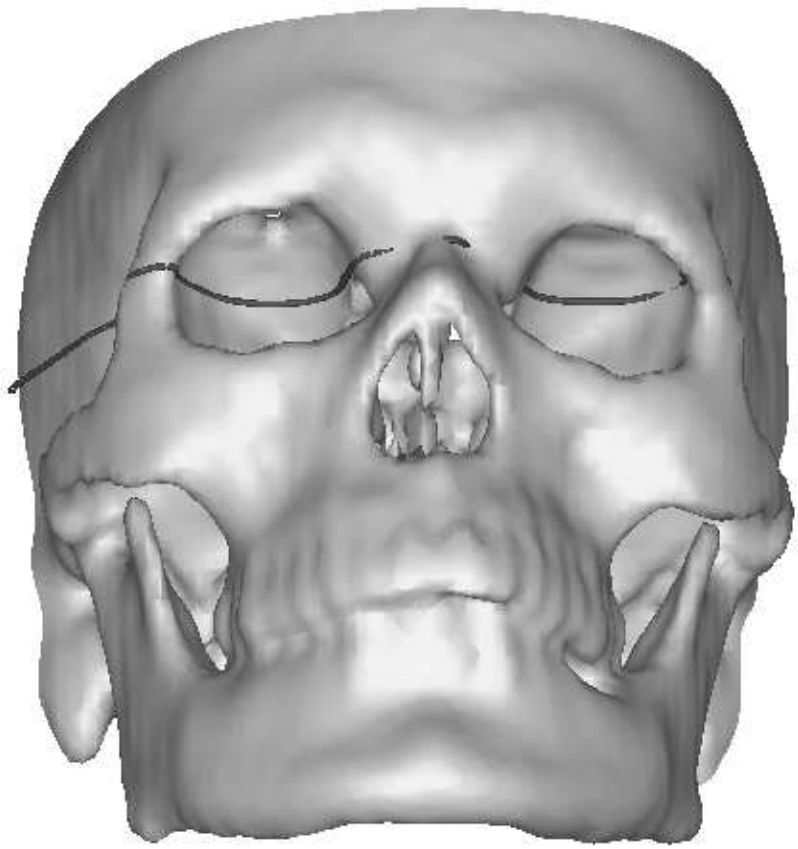

Figure 10: one particular fit found with the algorithm with pairs of points

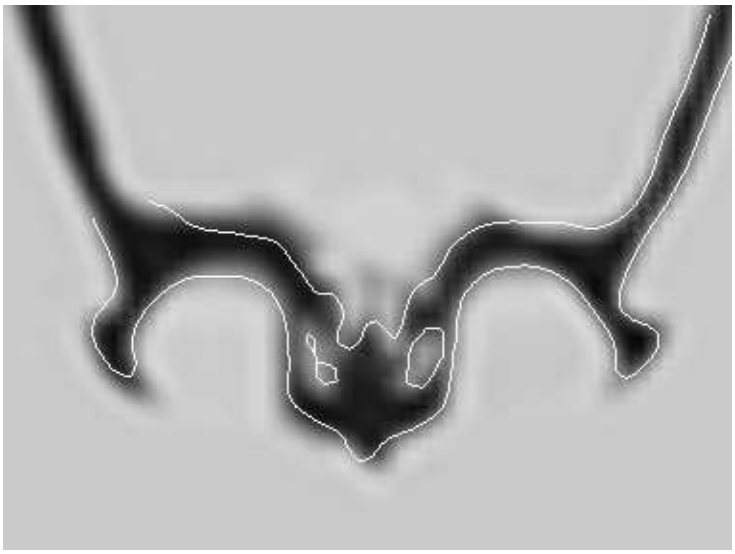

Figure 11: the curve is superimposed on the interpolated cross section image found by the registration program

5 Conclusion

In this paper we have presented :

- the differential constraints which can be exploited to register rigidly a curve on a surface.
- These constraints apply by considering either (1) homologous points or (2) homologous pairs of points, between the curve and the surface.
- Case (1) requires the computation of third order differential invariants and leads to a $O(np)$ algorithm.
- Case (2) requires the computation of first order differential invariants and leads to a brute force complexity of $O(n^2p^2)$ algorithm. This complexity can be significantly reduced, typically to $O(p\ \log(n))$ by applying additional constraints, in particular when the curve is planar and closed.

We presented results both with synthetic and real data, showing that only the second algorithm is robust enough in the presence of noise. Anyhow, the constraints used in the first algorithm can be used efficiently during the verification stage of the second algorithm, making the whole study useful in practice. We believe that in a near future we should have a much larger set of experimental results such as non-planar curves for example to further support this study.

6 Aknowledgements

The authors wish to thank Jean-Philippe Thirion for his contribution of the general formulation of the problem, Jacques Feldmar for the geometrical constraints with pairs of points and stimulating discussions, and also Gerard Medioni for his useful comments. This research was partially supported by Digital Equipment Corporation and the BRA-VIVA Esprit project.

Appendices

A Moving Frame

$(\mathbf{e_i}(s))_{(i\in\{1...n\})}$ is an orthonormal basis function of s, $(\mathbf{E_i})_{(i\in\{1...n\})}$ an other orthonormal basis and U is the rotation matrix such as for $(i \in \{1...n\})$ we have $\mathbf{e_i}(s) = U\mathbf{E_i}$. By computing the derivate with respect to s we obtain $d\mathbf{e_i}(s)/ds = dU/dsU^T\mathbf{e_i} = A\mathbf{e_i}$. As $UU^T = Id$ it infers by derivating wich respect to s that $A + A^T = 0$. if we are in the euclidean space $\mathbb{R}^3$, this property becomes :

$$\frac{d}{ds}\begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{e_3}\end{pmatrix} = \begin{pmatrix} & -r\mathbf{e_2} & +q\mathbf{e_3}\\ r\mathbf{e_1} & & -p\mathbf{e_3}\\ -q\mathbf{e_1} & +p\mathbf{e_2} & \end{pmatrix} \tag{12}$$

for more convenient use we write :

$$\frac{d}{ds}\begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{e_3}\end{pmatrix} = \begin{pmatrix}p\\ q\\ r\end{pmatrix} \wedge \begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{e_3}\end{pmatrix} = \mathbf{\Omega} \wedge \begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{e_3}\end{pmatrix} \tag{13}$$

B Differential Geometry of Surfaces

B.1 Definitions

A parametrized surface S is a differentiable map $\mathbf{X}$ from an open set U of $\mathbb{R}^2$ into $\mathbb{R}^3$, S is regular if the differential dX is one-to-one. On each point $\mathbf{p} = \mathbf{X}(u, v)$ of a regular surface we can construct the normal vector :

$$\mathbf{N}(u,v) = \frac{\mathbf{X_u} \wedge \mathbf{X_v}}{\| \mathbf{X_u} \wedge \mathbf{X_v} \|}^{5} \tag{14}$$

The natural inner product of $\mathbb{R}^3$ induces on each tangent plane $T_p(S)$ of a regular surface an inner product which corresponds a quadratic form I_p called the first fondamental form :

$$\begin{aligned} I_p(\mathbf{w}) &= <\mathbf{w}|\mathbf{w}> = \| \mathbf{w} \|^2 \\ I_p(\mathbf{X_u}du + \mathbf{X_v}dv) &= Edu^2 + 2Fdudv + Gdv^2 \end{aligned} \tag{15}$$

[5] where $\mathbf{X_u}$ is $\dfrac{\partial \mathbf{X}}{\partial u}$ and $\mathbf{X_v}$ is $\dfrac{\partial \mathbf{X}}{\partial v}$

with

$$E =< \mathbf{X_u}|\mathbf{X_u} > \quad F =< \mathbf{X_u}|\mathbf{X_v} > \quad G =< \mathbf{X_v}|\mathbf{X_v} > \tag{16}$$

The linear map h_p of $T_p(S)$ defined by :

$$h_p(\mathbf{X_u}) = -\mathbf{N_u} \ \& \ h_p(\mathbf{X_v}) = -\mathbf{N_v} \tag{17}$$

is self-adjoint (see [5] [16]). We can define the second fondamental form of $T_p(S)$ by :

$$\begin{aligned} II_p(\mathbf{w}) &= < \mathbf{w}|h_p(\mathbf{w}) > \\ II_p(\mathbf{X_u}du + \mathbf{X_v}dv) &= edu^2 + 2fdudv + gdv^2 \end{aligned} \tag{18}$$

The eigenvalues of h_p are called the principal curvatures (k_1, k_2) in p and the eigenvectors are the principal directions $\mathbf{e_1}, \mathbf{e_2}$.

from the expression of h_p in the basis $(\mathbf{X_u}, \mathbf{X_v})$:

$$H_p = \begin{pmatrix} E & F \\ F & G \end{pmatrix}^{-1} \begin{pmatrix} e & f \\ f & g \end{pmatrix} \tag{19}$$

we get :

$$\begin{aligned} H &= \frac{1}{2}(k_1 + k_2) = \frac{1}{2}\frac{Eg - 2Ff + Ge}{EG - F^2} && \text{(mean curvature)} \\ K &= k_1 k_2 = \frac{eg - f^2}{EG - F^2} && \text{(gaussian curvature)} \end{aligned}$$

B.2 Christoffel Symbols

By expressing the derivatives of the vectors $\mathbf{X_u}$ $\mathbf{X_v}$ and $\mathbf{N}$ in the basis $(\mathbf{X_u}, \mathbf{X_v}, \mathbf{N})$ we obtain :

$$\left\{ \begin{array}{lcl} \mathbf{X_{uu}} &=& \Gamma^1_{11}\ \mathbf{X_u} + \Gamma^2_{11}\ \mathbf{X_v} + e\ \mathbf{N} \\ \mathbf{X_{uv}} &=& \Gamma^1_{12}\ \mathbf{X_u} + \Gamma^2_{12}\ \mathbf{X_v} + f\ \mathbf{N} \\ \mathbf{X_{vv}} &=& \Gamma^1_{22}\ \mathbf{X_u} + \Gamma^2_{22}\ \mathbf{X_v} + g\ \mathbf{N} \\ \mathbf{N_u} &=& a_{11}\ \mathbf{X_u} + a_{21}\ \mathbf{X_v} \\ \mathbf{N_v} &=& a_{12}\ \mathbf{X_u} + a_{22}\ \mathbf{X_v} \end{array} \right.$$

the coefficients Γ^i_{jk} are called the Christoffel symbols (see [5]). with :

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = -H_p \tag{20}$$

By taking the scalar product of $\mathbf{X_{uu}}, \mathbf{X_{uv}}, \mathbf{X_{vv}}$ with $\mathbf{X_u}$ and $\mathbf{X_v}$ we obtain :

$$\left\{ \begin{array}{rcl} \begin{pmatrix} \Gamma^1_{11} \\ \Gamma^2_{11} \end{pmatrix} & = & \begin{pmatrix} E & F \\ F & G \end{pmatrix}^{-1} \begin{pmatrix} \frac{1}{2}E_u \\ F_u - \frac{1}{2}E_v \end{pmatrix} \\ \begin{pmatrix} \Gamma^1_{12} \\ \Gamma^2_{12} \end{pmatrix} & = & \begin{pmatrix} E & F \\ F & G \end{pmatrix}^{-1} \begin{pmatrix} \frac{1}{2}E_v \\ \frac{1}{2}G_u \end{pmatrix} \\ \begin{pmatrix} \Gamma^1_{22} \\ \Gamma^2_{22} \end{pmatrix} & = & \begin{pmatrix} E & F \\ F & G \end{pmatrix}^{-1} \begin{pmatrix} F_v - \frac{1}{2}G_u \\ \frac{1}{2}G_v \end{pmatrix} \end{array} \right.$$

B.3 Codazzi-Meinardi Equations

Since we have :

$$\left\{ \begin{array}{rcl} (\mathbf{X_{uu}})_v & = & (\mathbf{X_{uv}})_u \\ (\mathbf{X_{vv}})_u & = & (\mathbf{X_{uv}})_v \end{array} \right. \tag{21}$$

by taking the scalar product with $\mathbf{N}$ we obtain Codazzi - Meinardi equations:

$$\left\{ \begin{array}{rcl} e_v - f_u & = & e\Gamma^1_{12} + f(\Gamma^2_{12} - \Gamma^1_{11}) - g\Gamma^2_{11} \\ f_v - g_u & = & e\Gamma^1_{22} + f(\Gamma^2_{22} - \Gamma^1_{12}) - g\Gamma^2_{12} \end{array} \right. \tag{22}$$

For practical use, it is convenient to observe how Christoffel symbols and Codazzi-Mainardi simplify when the coordinate neighborhood contains no umbilical points and the coordinate curves are lines of curvature ($F = f = 0$) (it is shown in [5] that it is always possible to choose such a parametrization) In this case the principal curvatures and the principal directions may be written :

$$\left\{ \begin{array}{rcl} \mathbf{e_1} & = & \dfrac{\mathbf{X_u}}{\sqrt{E}} \qquad k_1 = \dfrac{e}{E} \\ \mathbf{e_2} & = & \dfrac{\mathbf{X_v}}{\sqrt{G}} \qquad k_2 = \dfrac{g}{G} \end{array} \right.$$

For Christoffel symbols we get :

$$\left\{\begin{array}{lcrlrlrl}
\mathbf{X_{uu}} & = & & \frac{E_u}{2E} & \mathbf{X_u} & -\frac{E_v}{2G} & \mathbf{X_v} & +e\ \mathbf{N} \\
\mathbf{X_{uv}} & = & & \frac{E_v}{2E} & \mathbf{X_u} & +\frac{G_u}{2G} & \mathbf{X_v} & \\
\mathbf{X_{vv}} & = & - & \frac{G_u}{2E} & \mathbf{X_u} & +\frac{G_v}{2G} & \mathbf{X_v} & +g\ \mathbf{N} \\
\mathbf{N_u} & = & - & k_1 & \mathbf{X_u} & & & \\
\mathbf{N_v} & = & & & & -k_2 & \mathbf{X_v} &
\end{array}\right.$$

and Codazzi-Meinardi equations take the following form :

$$\left\{\begin{array}{lcl} e_v & = & E_v H \\ g_u & = & G_u H \end{array}\right.$$

Since $\mathbf{N_{uv}} = \mathbf{N_{vu}}$ we obtain :

$$\left\{\begin{array}{lcl} (k_1)_v & = & \frac{E_v}{2E}(k_2 - k_1) \\ (k_2)_u & = & \frac{G_u}{2G}(k_1 - k_2) \end{array}\right.$$

Now we can express the derivatives of $(\mathbf{e_1}, \mathbf{e_2}, \mathbf{N})$ with respect to u and v :

$$\left\{\begin{array}{l} \frac{\partial}{\partial u}\begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{N}\end{pmatrix} = \begin{pmatrix} 0 \\ k_1\sqrt{E} \\ \frac{E_v}{2\sqrt{EG}}\end{pmatrix} \wedge \begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{N}\end{pmatrix} \\ \\ \frac{\partial}{\partial v}\begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{N}\end{pmatrix} = \begin{pmatrix} -k_2\sqrt{G} \\ 0 \\ \frac{-G_u}{2\sqrt{EG}}\end{pmatrix} \wedge \begin{pmatrix}\mathbf{e_1}\\ \mathbf{e_2}\\ \mathbf{N}\end{pmatrix} \end{array}\right. \tag{23}$$

Using the fact that [6]

$$\left\{ \begin{array}{rcl} \dfrac{\partial M}{\partial e_1} & = & \dfrac{1}{\sqrt{E}} M_u \\ \dfrac{\partial M}{\partial e_2} & = & \dfrac{1}{\sqrt{G}} M_v \end{array} \right.$$

we obtain :

$$\left\{ \begin{array}{rcl} \dfrac{\partial}{\partial e_1} \begin{pmatrix} \mathbf{e_1} \\ \mathbf{e_2} \\ \mathbf{N} \end{pmatrix} & = & \begin{pmatrix} 0 \\ k_1 \\ \dfrac{1}{k_2 - k_1} \dfrac{\partial k_1}{\partial e_2} \end{pmatrix} \wedge \begin{pmatrix} \mathbf{e_1} \\ \mathbf{e_2} \\ \mathbf{N} \end{pmatrix} \\ \dfrac{\partial}{\partial e_2} \begin{pmatrix} \mathbf{e_1} \\ \mathbf{e_2} \\ \mathbf{N} \end{pmatrix} & = & \begin{pmatrix} -k_2 \\ 0 \\ \dfrac{1}{k_2 - k_1} \dfrac{\partial k_2}{\partial e_1} \end{pmatrix} \wedge \begin{pmatrix} \mathbf{e_1} \\ \mathbf{e_2} \\ \mathbf{N} \end{pmatrix} \end{array} \right. \tag{24}$$

C Differential Geometry of Curves

C.1 Frenet and Darboux Frames

A regular parametric curve α is a differentiable map f of an open interval of $\mathbb{R}$ into $\mathbb{R}^3$ with $df/dx \neq 0$. the tangent in x is $\mathbf{t} = \dfrac{df}{dx} / \parallel \dfrac{df}{dx} \parallel$ we can construct the arc-length s by :

$$\frac{ds}{dx} = \parallel \frac{df}{dx} \parallel \tag{25}$$

then in each point M of α,the Frenet frame $(\mathbf{t}, \mathbf{n}, \mathbf{b})$ verifies :

$$\frac{d}{ds} \begin{pmatrix} \mathbf{t} \\ \mathbf{n} \\ \mathbf{b} \end{pmatrix} = \begin{pmatrix} \tau \\ 0 \\ -k \end{pmatrix} \wedge \begin{pmatrix} \mathbf{t} \\ \mathbf{n} \\ \mathbf{b} \end{pmatrix} \tag{26}$$

k is called the curvature and τ the torsion. $\mathbf{t}$ is called the tangent, $\mathbf{n}$ the normal and $\mathbf{b}$) the binormal. When a curve α lies on a surface S, we can define the Darboux frame by :

[6] $M\ (u, v)$, $\dfrac{\partial M}{\partial e_1}(m) = \dfrac{dM}{dt}(m + t\mathbf{e_1})$ for $(t = 0)$ is called the derivative of M with respect to $\mathbf{e_1}$ in m .

- $\mathbf{t}$ is the tangent at the curve α in M.
- $\mathbf{N}$ is the normal of the surface S in M.
- $\mathbf{g} = \mathbf{N} \wedge \mathbf{t}$ is called the geodesic normal.

then we have :

$$\frac{d}{ds}\begin{pmatrix} \mathbf{t} \\ \mathbf{g} \\ \mathbf{N} \end{pmatrix} = \begin{pmatrix} \tau_g \\ k_n \\ -k_g \end{pmatrix} \wedge \begin{pmatrix} \mathbf{t} \\ \mathbf{g} \\ \mathbf{N} \end{pmatrix} \tag{27}$$

τ_g is called the geodesic torsion, k_g the geodesic curvature and k_n the normal curvature.

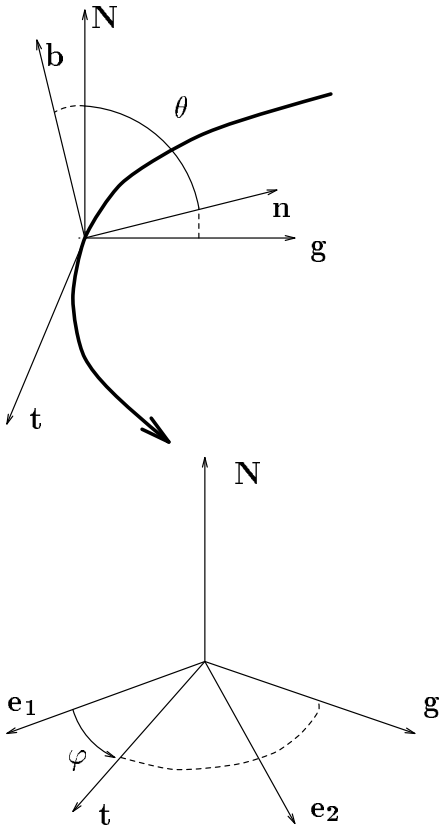

Figure 12: Up : Frenet and Darboux frames, down : Darboux frame and principal frame of the surface

C.2 Variation of the surface principal frame along the curve

Since that for each function P of the coordinates of the surface (u, v), the derivative of the restriction of P on the curve may be written :

$$\frac{dP}{ds} = \frac{\partial P}{\partial e_1}\cos\varphi + \frac{\partial P}{\partial e_2}\sin\varphi \tag{28}$$

The result showed in appendix B, yields to the expression of the variation of the surface principal frame along the curve :

$$\frac{d}{ds}\begin{pmatrix} \mathbf{e_1} \\ \mathbf{e_2} \\ \mathbf{N} \end{pmatrix} = \begin{pmatrix} -k_2\sin\varphi \\ k_1\cos\varphi \\ \dfrac{1}{k_2 - k_1}\left(\dfrac{\partial k_1}{\partial e_2}\cos\varphi + \dfrac{\partial k_2}{\partial e_1}\sin\varphi\right) \end{pmatrix} \wedge \begin{pmatrix} \mathbf{e_1} \\ \mathbf{e_2} \\ \mathbf{N} \end{pmatrix} \tag{29}$$

References

[1] N. Ayache. Volume image processing, results and research challenges. Technical Report 2050, INRIA, 1993.

[2] Paul J. Besl. *Surface in Range Image Understanding.* Springer-Verlag, 1986.

[3] Paul J. Besl. *The Free-Form Surface Matching Problem (Machine Vision for Three-Dimensional Scenes).* H. Freeman, Ed., 1990.

[4] P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on PAMI*, 14(2), February 1992.

[5] M.P. Do Carmo. *Differential geometry of curves and surfaces.* Prentice Hall, 1976.

[6] Eric L. Grimson. *Object Recognition by Computer, the Role of Geometric Constraints.* Mit Press, 1990.

[7] W.E.L Grimson, T. Lozano-Pérez, N. Noble, and S.J. White. An automatic tube inspection system that finds cylinders in range data. In *Proceedings CVPR '93, New York City, NY.* IEEE, June 1993.

[8] André Gueziec. *Reconnaissance automatique de surfaces et courbes gauches. Application l'analyse d'images volumiques.* PhD thesis, Université de Paris Sud, Centre d'Orsay, 1993.

[9] André Guéziec and Nicholas Ayache. Smoothing and matching of 3D-space curves. In *Proceedings of the Second European Conference on Computer Vision*, Santa Margherita Ligure, Italy, May 1992. also an INRIA Research Report (1544).

[10] Stéphane Lavallée, Richard Szeliski, and Lionel Brunie. Matching 3-d smooth surfaces with their 2-d projections using 3-d distance maps. In *SPIE, Geometric Methods in Computer Vision*, San Diego, Ca, July 1991.

[11] Grégoire Malandain. *Filtrage, topologie et mise en correspondance d'images médicales multidimensionnelles.* PhD thesis, Université de Paris Sud, Centre d'Orsay, 1992.

[12] Grégoire Malandain and Jean-Marie Rocchisanni. Registration of 3d medical images using a mechanichal based method. In *IEEE EMBS satellite symposium on 3D Advanced Image Processing in M edecine*, Rennes, November 1992.

[13] Olivier Monga and Serge Benayoun. Using partial derivatives of 3d images to extract typical surface features. *rapport de recherche INRIA*, March 1992.

[14] Olivier Monga, Serge Benayoun, and Olivier D. Faugeras. Using partial derivatives of 3d images to extract typical surface features. In *Proceedings CVPR '92, Urbana Champaign, Illinois*. IEEE, July 1992. also an INRIA Research Report (1599).

[15] Joseph L. Mundy and Andrew Zisserman. *Geometric Invariance in Computer Vision*. Mit Press, 1990.

[16] J. Odoux., E. Ramis, and C. Deschamps. *Cours de mathémathiques spéciales,Tome 5 : applications de l'analyse à la géométrie.* Masson, 1981.

[17] Jean Ponce, Anthony Hoogs, and David J. Kriegman. On using cad models to compute the pose of curved 3d objects. *CVGIP : Image Understanding*, March 1992.

[18] William H. Press, Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C, The Art of Scientific Computing.* Cambridge University Press, 1990.

[19] B. Harr Romeny, L. Florack, A. Salden, and M. Viergever. Higher order differential structure of images. In H.H. Barrett and A.F. Gmitro, editors, *Information Processing in Medical Imaging*, pages 77–93, Flagstaff, Arizona (USA), June 1993. IPMI'93, Springer-Verlag.

[20] Gabriel Taubin, Ruud M. Bolle, and Baba C. Vermuri. Constrained implicit function fitting. In *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 14,No 2*. IEEE, February 1992.

[21] J-P. Thirion, A. Gourdon, O. Monga, A. Gueziec, and N. Ayache. Fully automatic registration of 3d cat-scan images using surface curva ture.

In *IEEE EMBS satellite symposium on 3D Advanced Image Processing in M edecine*, Rennes, November 1992.

[22] Jean-Philippe Thirion and Alexis Gourdon. The 3D marching lines algorithm and its application to crest lines extraction. Technical Report 1672, INRIA, May 1992.

Unité de recherche INRIA Lorraine, Technôpole de Nancy-Brabois, Campus scientifique,
615 rue de Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, IRISA, 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