HAL
open science

The Extremal mesh and the understanding of 3D surfaces

Jean-Philippe Thirion

To cite this version:

Jean-Philippe Thirion. The Extremal mesh and the understanding of 3D surfaces. [Research Report] RR-2149, INRIA. 1993, pp.30. inria-00077150

HAL Id: inria-00077150

https://inria.hal.science/inria-00077150v1

Submitted on 29 May 2006

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

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

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

The Extremal Mesh and the Understanding of 3D Surfaces

Jean-Philippe THIRION

N° 2149

Decembre 1993

PROGRAMME 4

Robotique,
image
et vision

Rapport de recherche

1993

ISSN 0249-6399

The Extremal Mesh and the Understanding of 3D Surfaces

Jean-Philippe THIRION *

Programme 4 — Robotique, image et vision
Projet Epidaure **

Rapport de recherche n° 2149 — Decembre 1993 — 30 pages

Abstract: This paper is about a new concept for the description of 3D smooth surfaces : the *extremal mesh*. In previous works, we have shown how to extract the *extremal lines* from 3D images, which are the lines where one of the two principal surface curvatures is locally extremal. We have also shown how to extract the *extremal points*, which are specific points where the two principal curvatures are both extremal. The extremal mesh is the graph of the surface whose vertices are the extremal points and whose edges are the extremal lines : it is invariant with respect to rigid transforms. The good topological properties of this graph are ensured with a new local geometric invariant of 3D surfaces, that we call the *Gaussian extremality*, and which allows to overcome orientation problems encountered with previous definitions of the extremal lines and points. This paper presents also an algorithm to extract the extremal mesh from 3D images, and experiments with synthetic and real 3D medical images showing that this graph can be extremely precise and stable. The extremal mesh and the Gaussian extremality are new insights into the geometrical nature of 3D surfaces, with many promising consequences, some of which being listed at the end of this paper.

Key-words: Differential Geometry, Geometric Invariant, 3D Image Processing, 3D surface.

*Email: jean-philippe.thirion@sophia.inria.fr
**http://zenon.inria.fr:8003/Equipes/EPIDAURE-eng.html

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

(Résumé : tsvp)

Le Maillage Extrémal et la Compréhension des Surfaces 3D

Résumé :

Cet article présente un nouveau concept pour la description des surfaces 3D : le *maillage extrémal*. Lors de travaux précédents, nous avions montré la possibilité d'extraire les *lignes extrémales*, à partir d'images 3D, qui sont les lignes où l'une des deux courbures principales est localement extrémale. Nous avons également montré comment extraire les *points extrémaux*, qui sont les points singuliers correspondant à des maxima locaux simultanés des deux courbures principales.

Le maillage extrémal est le graphe de la surface dont les sommets sont les points extrémaux et dont les arêtes sont les lignes extrémales : ce graphe est invariant par transformation rigide. Les propriétés topologiques de ce graphe sont assurées par un nouvel invariant géométrique local, que nous avons appelé l' *extrémalité Gaussienne*, et qui permet d'éliminer tous les problèmes d'orientation rencontrés lors des définitions précédentes de lignes et de points extrémaux. Nous présentons également un algorithme original d'extraction de ce maillage à partir d'images 3D, synthétiques ou réelles, et nous montrons la remarquable stabilité de ce graphe. Le maillage extrémal et l'extrémalité Gaussienne sont deux perceptions nouvelles de la nature géométrique des surfaces 3D, pour lesquelles nous énumérons un certain nombre de conséquences possibles en fin d'article.

Mots-clé : Géométrie Différentielle, Invariant Géométr ique, traitement d'images 3D, Surface 3D.

1 Introduction

By understanding 3D surfaces, we mean finding stable and compact representations of those surfaces, which facilitate their storage, manipulation and reconstruction. For example, the piecewise linear surfaces (polyhedra) can be represented by their vertices, edges, and polygonal faces. This representation can be manipulated and displayed (Computer Graphics), or measured and extracted from real images (Computer Vision).

What about curved surfaces ? There are many works about the modeling of those surfaces with piecewise polynomial or rational functions (splines, nurbs), and more recently with high order polynomials (algebraic surfaces, see for example the work of Taubin in [11]). People are now also interested in extracting curved surfaces from 3D images, such as X-ray scanner or Magnetic Resonance Images ([16], [4]). The major problem is how to deal with the surfaces of objects whose topology is not a-priori known, and which can be arbitrarily complicated (such as a skull, for example). Even with a known topology, it can be extremely difficult to represent the curved surface by the way of piecewise polynomials, because it is very hard to maintain derivatives continuity at the junctions of the model (except for very simple cases such as a spherical, cylindrical or toric topology).

The aim of the present paper is to present a natural decomposition of the 3D surfaces into a graph very similar to the polyhedral representation of piecewise linear surfaces (vertices, edges and faces), and which is defined even for objects whose topology can be arbitrarily complicated. We call this graph *the extremal mesh* (EM) of the surface ("the" because the EM is invariant with respect to rigid transforms).

We recall first some notions of differential geometry, including the recent notions of extremal lines (ELs) and extremal points (EPs). Then we explain in more detail the orientation problems encountered in the definition of extremal lines and points, and we introduce a new notion, the *Gaussian extremality* (E_g), a local geometric invariant, which enable us to overcome those orientation problems and to define the extremal mesh. We describe a local algorithm for the practical extraction of the EM, with also experiments with synthetic and real data to show the stability of the EM. At last, we discuss some open points and possible applications of the extremal mesh, including the automatic parametrization of smooth implicit surfaces, and the automatic extraction of some umbilic points.

2 Differential geometry and 3D surfaces

A good introduction to differential geometry can be found in Do Carmo [2], or Koenderink [3], and more specialized material about ridges and umbilics can be found in Porteous [9]. We just recall here some notions of differential geometry, useful for the introduction of the Gaussian extremality E_g.

2.1 Smooth surfaces

We will call smooth surface a surface which is infinitely differentiable at any point (in fact, we will use only the continuity up to the third derivatives).

At any point P of a smooth surface S, one can define an infinite number of curvatures k, one curvature k_t for each direction t in the tangent plane of S at P. The direction is t, and not $\vec{t}$, because this direction is not oriented. $\vec{t}$ or $-\vec{t}$ give the same curvature k_t, an important point for the rest of this paper.

2.2 Principal curvatures and directions

The curvature k is a continuous function of the angle of rotation of t in the tangent plane, which implies (except when k is constant) the existence of two extremal values : the maximal and the minimal curvatures, called *principal curvatures* $k_i, i \in \{1, 2\}$. If k is constant, the surface is locally spherical : such very special points are called the *umbilic points* (see Sander and Sucker in [10]).

The directions t_i, associated with the principal curvatures k_i, are called the principal directions, k_i and t_i being respectively the eigenvalues and the eigenvectors of a 2D endomorphism called the *Weingarten endomorphism.* t_1 and t_2 are orthogonal.

The two notions of Gaussian curvature $K = k_1 k_2$, and mean curvature $H = (k_1 + k_2)/2$, have been introduced, mainly in order to have two values, strictly equivalent to the principal curvatures k_i, but for which it is not necessary to distinguish between the two principal curvatures. Hence K and H are well defined even for umbilic points, whereas k_1 and k_2 are indistinguishable there. This justification for the definition of K and H is often forgotten among many other beautiful properties of those two operators for the description of surfaces.

The equations for the principal directions and curvatures can be found in for example Do Carmo [2] for the case of parametric surfaces, and in Thirion and Gourdon [15], for the case of implicit surfaces.

2.3 Lines of curvature

There is nowaday no method to find automatically a complete parametrization of smooth surfaces, for objects with an arbitrarily complex topology. The implicit functions theorem states that we can always find a local parametrization, but tremendous problems occur trying to go from a local parametrization to a global parametrization.

In fact, the "ideal" parametrization of smooth surfaces is already known. This natural parametrization is the parametrization with the *lines of curvature*. The lines of curvatures are the envelops of the principal directions, that is, curves whose tangents are the principal directions. There are curvature lines for both principal curvatures, leading to a complete mesh in the surface : at any point, which is not an umbilic, there are two orthogonal lines of curvature.

Unfortunately, in practice, there is no way to compute explicitly the equations of the lines of curvature. If we get some knowledge of what the curvature lines look likes, we can only verify that the equation is indeed the good one. Again, this is the case of the ellipsoid, for which the parametric equations of the curvature lines are known.

Furthermore, the lines of curvature can't be computed with simple numerical integration techniques. One could think to start from a point, compute the principal direction, advance from one step in that direction, and iterate until the starting point (or an umbilic), is reached. The problem is that, because of computational errors, the reconstructed line diverges from the true line of curvature, and the algorithm never ends. Relaxation algorithms have been proposed : a closed curve evolves toward the solution trying to minimize both some internal smoothness constraints and the angle between its tangents and the computed principal directions.

Such a method can work, only if there is some information on how to place the original closed curves in the surface. This is possible for the ellipsoid, but not for an object with an arbitrary topology.

2.4 Red and blue ridges

Olivier Monga et al. have shown in [6] that ridges, that is, lines where the maximal curvature is locally maximal, can be characterized as the zero crossing of a coefficient e, that they called the *extremality*. e is the directional derivative of the maximum principal curvature (let say that it is k_1) in the corresponding principal direction $e_1 = \vec{\nabla} k_1 \cdot \vec{t_1}$.

There exists also another equivalent definition for those lines. For any smooth surface S, there exist two associated surfaces S_1 and S_2, the *evolutes* or *focal surfaces* of S, which are the loci of the center of *principal* curvatures of S. Those evolutes have cuspidal edges, the *ribs* of S, whose associated curves in S are called *ridge lines* (generally referred to as red and blue ridge lines, each color being associated with one principal curvature, see [9] or [3], pages $290 - 294$). The evolutes of S are the 3D equivalent of the evolute of a planar curve, whose cusps are related to the points of the 2D curve with extremal curvature (see figure 1).

2.5 Extremal lines

We think however that the word "ridge" is not well suited for the lines corresponding to the ribs of S, because such lines can be also lines whose associated principal curvature is minimal (in absolute value) in the corresponding principal direction : this is why we prefer the word of "extremal" line instead of "ridge" line (ridge sounds strange for the "flattest" loci of a surface). For the extraction of the extremal line, we use the following characterization :

Preliminary definition

As there are two principal directions, there are also two extremality coefficients $e_i = \vec{\nabla} k_i \cdot \vec{t_i}, i \in \{1, 2\}$. We call *extremal lines* (ELs) the lines defined by the zero crossing of anyone of those two extremality coefficients.

The principal curvatures can be distinguished as maximal and minimal curvature, but provided that $|k_1| \neq |k_2|$, we can also distinguish them by using their absolute value. We call *Largest curvature* the principal curvature whose absolute value is the largest, and *second (principal) curvature* the other principal curvature. We do this because we have noticed experimentally that

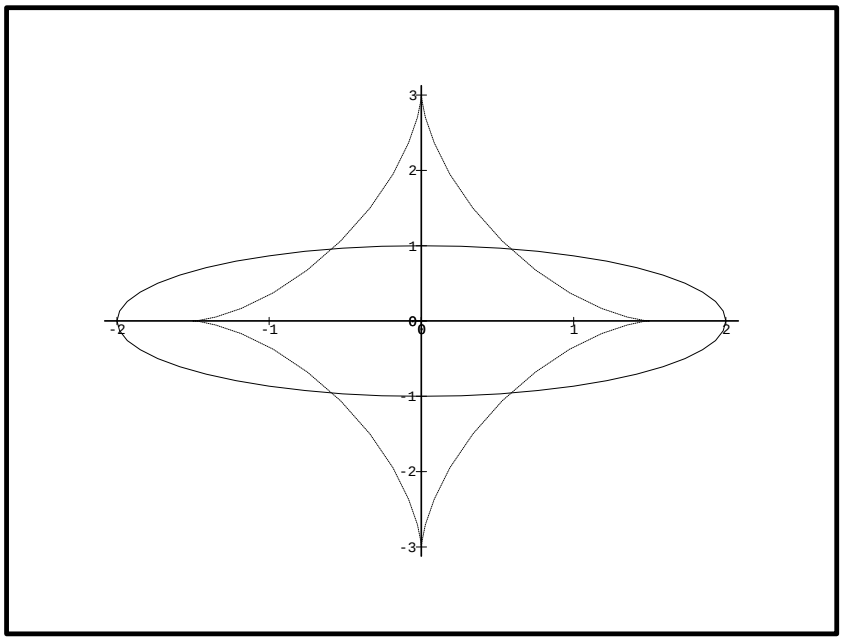

Figure 1: The evolute of the ellipse

the maxima (in absolute value) of the largest curvature are the most stable, whatever the sign of the curvature. We call those lines the *crest lines*, which is a notion which doesn't depend on the orientation of the surface : the ridge of an object is also a groove of the background : we call both ridges and grooves "crest lines".

Hence the ELs can be classified into 4 types, depending on the type of extremality of the extremality zero-crossing :

- lines of maximum largest curvature (crest lines).
- lines of minimum largest curvature
- lines of maximum second curvature
- lines of minimum second curvature

The type of extremality (minimum or maximum) can be determined with the sign of the derivative of the extremality coefficient in the associated principal direction ($\vec{\nabla e_i} \cdot \vec{t_i}$), and the sign of the curvature (k_i). Extremal lines are Crest Lines when, if $|k_1| > |k_2|$: $e_1 = 0$, and either ($\vec{\nabla e_1} \cdot \vec{t_1} > 0$ and $k_1 < 0$) or ($\vec{\nabla e_1} \cdot \vec{t_1} < 0$ and $k_1 > 0$).

For each type of Els, we can distinguish between 4 sub-types, taking into account the signs of the two principal curvatures (largest and second). In that way, our classification of extremal lines is more detailed than the classical notions of red and blue ridge lines.

We have shown in [14] and [15] how to extract automatically all the lines corresponding to the zero crossings of e_i, as the intersection of two implicit surfaces, and with an original algorithm called the "Marching Lines". Our experimentations with 3D medical images have shown the remarkable stability of those lines, which are not only geometrical invariants, but in some sense also anatomical invariants : from the 3D CT scans of the head, the longest extracted lines are always the nose and eyes orbital lines, the submandibular line, the temporal lines, and the foramen occipital. We measured the precision of those lines by using different 3D acquisitions of the same anatomical object (for example two 3D images of the same skull), performing the automatic registration of the two extracted sets of lines, and measuring the distances between registered lines.

2.6 Extremal points

We defined the *extremal points* (EPs) being the simultaneous zero crossing of the two extremalities $e_1 = 0, e_2 = 0$. This notion is close to the notion of *purple ridge point* (the crossing of red and blue ridge lines, see [9]). In [12], we have shown how to extract automatically the EPs from 3D images, and also that those points are remarkably stable, using similar experiments as for the crest lines : from an image of 5 million of voxels, we extract typically 2000 EPs, and after automatic registration, we have about 700 registered EPs, with a standard deviation of 0.4 (voxel size) in position (the extremal points have also a great potential value for anatomical measurements, we have already identified EPs of the skull, stable from patent to patient).

In [13], we have extended the notion of extremal points to any dimensions, showing for example that a new kind of EPs can be computed from the 4D hypersurface corresponding to the intensity of a 3D image $i = f(x, y, z)$ (applying results of O. Monga and S. Benayoun in [5] about the differential geometry in $\mathcal{R}^4$).

3 The Gaussian extremality and the extremal mesh

All this would have been perfect, except for one thing : $e_i = \vec{\nabla} k_i \cdot \vec{t_i}$ is a correct definition only when $\vec{t_i}$ is an oriented direction ! If we change the orientation of $\vec{t_i}$ into $-\vec{t_i}$, e_i turn into $-e_i$: what can be then the meaning of the zero crossing of a value whose sign is meaningless ?

3.1 The chess-board effect

We were aware of that problem since the beginning of our experiments. In [14], we explain that one can always orient the largest principal direction with respect to a fixed direction in space. A surprising result is that the formulae of the principal directions are symmetrical with respect to the 3 coordinates x, y, z of the euclidean space only by giving some privilege to the direction $(1, 1, 1)$. In fact, no global arbitrary orientation holds without some parts of the surface for which the extremalities are abruptly reversed, creating "ghost" extremal lines, not stable with respect to rigid transform.

We called this artefact, the "chess-board" effect. It gave us also a way to eliminate ghost extremal lines, because they were the only lines not independent to a change in the orientation. Also, we modified the original Marching Lines algorithm in order to orient locally the principal directions, using the tangent direction of the current EL being extracted.

3.2 The umbilic points

Problems occur also in the neighborhood of umbilic points. There, the change of the principal curvatures directions are swift, creating distortions and artificial endings of the extremal lines. The problem of the practical extraction of umbilic points have been studied by Sander and Zucker in [10]. They propose a solution, based on the computation of the index of the vector field corresponding to the principal directions, which is known to be $\pm 1/2$ for umbilics and 0 elsewhere.

In order to compute the field of principal directions, they propose a relaxation scheme, were the principal directions are aligned by minimizing both a regularity constraint, and a closeness to real data constraint. We will see later in this paper that the extremal mesh is another possible way to extract umbilic points.

3.3 A true local invariant

The solution that we propose, in order to suppress the orientation problems, is to use a value which is really invariant with respect to spatial orientation. The most logical choice at this point is the product of the two extremality coefficients $E_g = e_1 e_2$. We call E_g the *Gaussian extremality*, by reference to the famous mathematician, and by analogy with the Gaussian curvature $K = k_1 k_2$, more precisely (see also figure 2):

Definition : the Gaussian extremality

Let S be an oriented surface, at least 3 time differentiable, of $\mathcal{R}^3$. Let P be a point of S which is not an umbilic point and $\vec{n}$ be the oriented normal of S in P. Let k_1, k_2 be the principal curvatures at P with respect to the orientation $\vec{n}$ of S, and with $k_1 > k_2$. Let $\vec{t_1}, \vec{t_2}$ be two normalized vectors parallel with the principal directions t_1, t_2, and such that $(\vec{t_1}, \vec{t_2}, \vec{n})$ is a direct

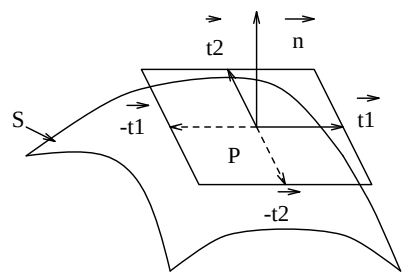

Figure 2: Orientation of the principal directions

orthonormal basis. The Gaussian extremality E_g of S in P is the product $e_1 e_2$, where $e_1 = \vec{\nabla} k_1 \cdot \vec{t_1}$ and $e_2 = \vec{\nabla} k_2 \cdot \vec{t_2}$.

Theorem

The Gaussian extremality is invariant with respect to the positive isometries of $\mathcal{R}^3$, opposite with the negative isometries and invariant with respect to the orientation of the surface.

The proof is easy : it comes directly from the definition of the Gaussian extremality, which is based on values which are invariant with respect to isometries. The only difficulty is the invariance with respect to the orientation of S. Let $\vec{n}$ be an arbitrary orientation of S at P, the maximal and minimal curvatures k_1 and k_2 are defined only because of this orientation which fixes their signs. With $(\vec{t_1}, \vec{t_2}, \vec{n})$ being direct, there are only two acceptable choices for the principal direction orientations, which are either $\vec{t_1}, \vec{t_2}$ or $-\vec{t_1}, -\vec{t_2}$. In both cases, the product $e_1 e_2$ is the same. Now, suppose that we reverse the orientation of the surface, turned into $-\vec{n}$. Then $-k_2$ becomes the maximal curvature and $-k_1$ the minimal one. We must have a direct $(\vec{t_2}, \vec{t_1}, -\vec{n})$, which is equivalent to a direct $(\vec{t_1}, \vec{t_2}, \vec{n})$. The extremalities become $-e_1$ and $-e_2$ which has no effect on the Gaussian extremality. Therefore E_g is also invariant to a change of the surface orientation.

The Gaussian extremality is similar to the Gaussian curvature in that it is invariant with respect to the orientation of the surface. However, on the

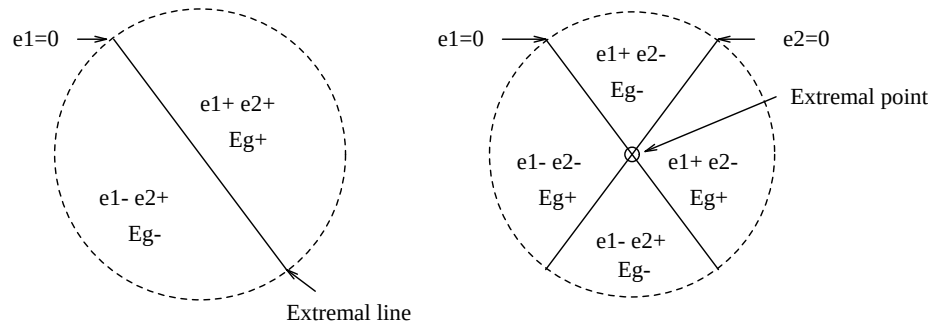

Figure 3: Zero crossings of the extremalities in closed regions of the surface

contrary to the Gaussian curvature, a negative isometry (a symmetry with respect to a plane) inverts the Gaussian extremality values (because of the convention $(\vec{t_1}, \vec{t_2}, \vec{n})$ direct).

Furthermore, we don't know yet how to define the Gaussian extremality for umbilic points. We agree that the continuation of E_g for umbilic points is the remaining theoretical problem of our method.

3.4 The extremal mesh

We have now a true local invariant, the Gaussian extremality E_g. The signs of E_g gives a partition of the surface into regions where $(E_g > 0)$, regions where $E_g < 0$, and regions where $E_g = 0$. The boundaries of those regions are the extremal lines, because, except for umbilic points, $E_g = 0$ only when $e_1 = 0$ or $e_2 = 0$. This is illustrated in figure 3 (and also figures 4 and 5). An extremal point is a singular point of those boundaries (see figure 3), being the common vertex of four neighboring E_g regions alternatively positive and negative : locally, and far from the umbilic, we can give a coherent orientation to all the principal directions, and those four regions correspond to the different possible signs for e_1 and e_2.

We conjecture that isolated umbilic points are the vertices of an even number of E_g regions. For the case of an ellipsoid (figure 4, 10), the number

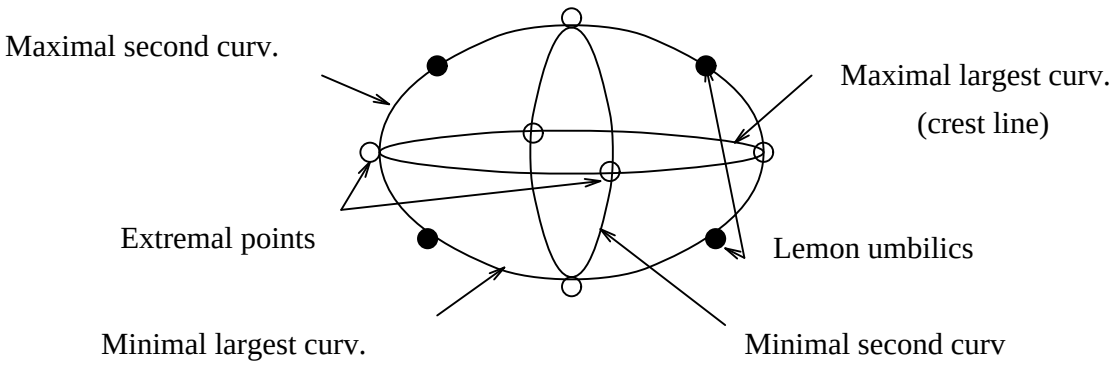

Figure 4: The extremal mesh of an ellipsoid

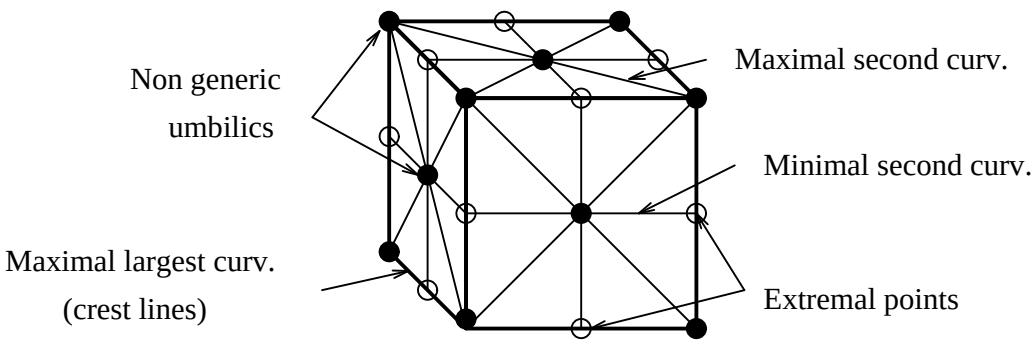

Figure 5: The extremal mesh of a rounded cube

of regions is only 2 for the umbilic points; a line of minimum largest curvature turning into a line of maximum second curvature.

For the case of a rounded cube (see figure 5, 11), the umbilics are found at the vertices of the cube and at the center of the faces. The umbilics of the cube vertices appear to be the vertices of 6 E_g regions, whereas the center of faces appear to be the vertices of 8 E_g regions (not quite apparent in figure 11 for reasons detailed later on).

Of course, those objects (ellipsoid and cube) are highly symmetric, which could lead us to wrong intuitions. Nevertheless, it seems that isolated umbilic points are similar to the extremal points, but with an arbitrary even number of adjacent E_g regions. As it is not clear to state if umbilics can be considered to be extremal points or not, we will continue to distinguish both.

We define the extremal mesh as the graph whose vertices are the extremal points or umbilic points, and whose edges are the extremal lines.

In the case of the ellipsoid, the extremal mesh has eight cells, each cell having four edges, being the succession of the four kinds of extremal lines. In the case of the rounded cube, there are only 3 edged cells, but this could be a degenerated case (one edge being reduced to an umbilic point). Note that the ellipsoid has stable singularities (i.e. not sensitive to a small perturbation) whereas the rounded cube has instable singularities, involving higher order derivatives, and is therefore less representative of the general case than the ellipsoid. Such singularities are discussed in [8].

4 The extraction of the extremal mesh

In this paper, the proposed algorithm is illustrated with the case of 3D images, but it applies also to any virtual implicit surfaces, when a procedure giving the implicit equation of the surface and its derivatives at each point of a 3D virtual regular grid is available.

A simple extraction method would be to assimilate the $(E_g = 0)$ regions to for example the $(E_g > 0)$ regions. Hence the problem is to find the interface lines in the surface between $(E_g \geq 0)$ and $(E_g < 0)$ regions, which can be efficiently extracted with the Marching Lines algorithm (see [15]).

What is extracted however, is not a wire mesh, but a set of closed curves. Those curves are infinitely close at the level of the extremal points, but, due to the discretization, the reconstructed curves may be several voxels apart,

preventing from an easy detection of the extremal points. We describe now a way to overcome this problem, based on a modification of the Marching Lines technique (in the following, we will make constant references to the notions described in [15]).

This phenomenon is illustrated in figures 3 and 6. An iso-surface extraction technique produces a given number of adjacent polygonal cells, representing the surface. We can compute for each vertex of each cell, the differentials of the image, up to order 3 (we use the filtering of the 3D image with the derivatives of the Gaussian function), and with the formulae of [14], we deduce from them all the differential characteristics of the surface, including the extremalities e_1 and e_2. If we are far from an umbilic, we can easely give a coherent local orientation to all principal directions.

Figure 3 describes simple cases, where there is a perfect correspondence between the zero-crossings of the Gaussian extremality Eg and the individual extremalities e_1 and e_2. However, in figure 6, we have representing two adjacent cells coming from the iso-surface extractor (a discretization of the surface : these cells have both 4 edges, but may have any number of edges in practice). The distribution of the extremality values at the vertices of those cells is such that the detection of the extremal points is no more a local problem : if we are looking only for the zero-crossings of E_g, we miss the extremal point.

On the other hand, we have already been able to extract the extremal points with a great precision, by extracting the $(e_1 = 0)$ and $(e_2 = 0)$ lines individually, with a propagation of a local orientation of the principal direction (see [12]). This solution is satisfying at a local level, but doesn't ensure the global topological properties of the wire mesh.

We propose to mix those two methods. The precision is achieved by computing and orienting locally the principal directions, in order to extract the individual zero-crossings of the two extremalities e_1 and e_2, whereas the Gaussian extremality E_g is used to maintain good global topological properties, suppressing possible ambiguities.

Our algorithm is purely local : the computation in each cell of the iso-surface is independent from all the other cells (easy to parallelize). As each surface cell itself is included in a cell of the 3D grid, the computation is local also in the 3D image (except for the Gaussian pre-filtering used to extract the differentials of the 3D image, whose support is several voxels).

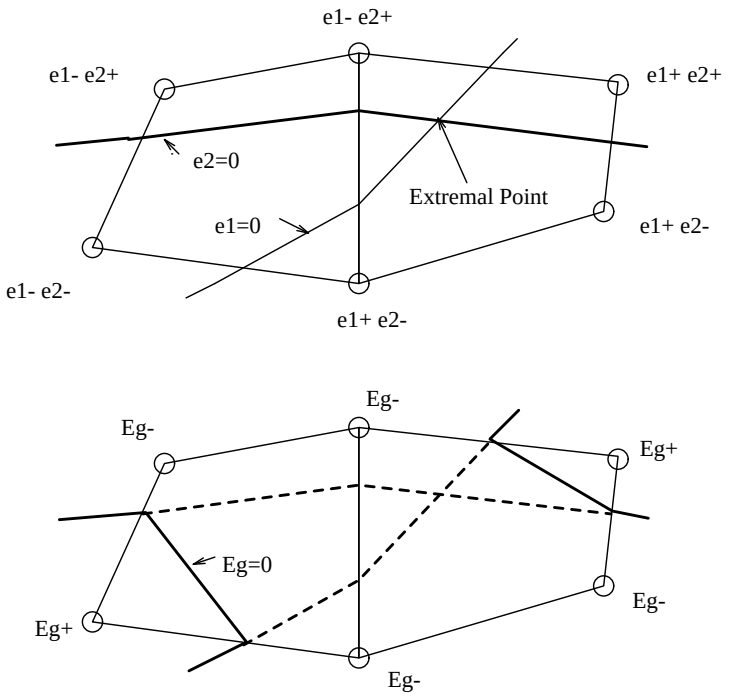

Figure 6: A bad situation : the extremal point is not detected with only the Gaussian extremality values at the vertices of the cells : the problem is no more local

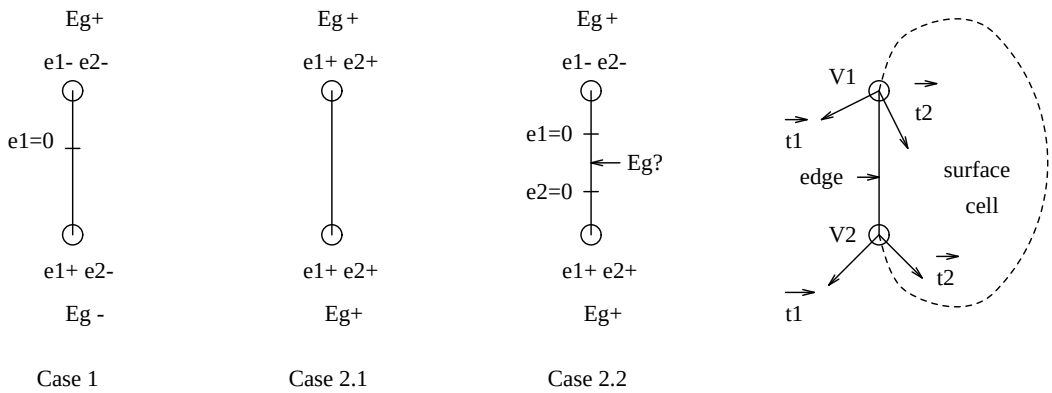

Figure 7: The processing of an edge of a cell of the iso-surface

4.1 The edge processing

We have shown in [15] that in order to ensure global topological properties, the computation of each edge of a cell of the surface must be independent from any other measured values, and deterministic, that is, it must give the same result for the two adjacent cells which share the edge.

We first compute the values e_1, e_2 and E_g for the two vertices V_1 and V_2 of the edge (see figure 7). We ensure a local compatibility between e_1, e_2 in V_1 and e_1, e_2 in V_2, for example by orienting the maximal principal directions $\vec{t_1}$ in V_1 and $\vec{t_1}$ in V_2 such that their scalar product is positive.

- **Case 1 :** If the signs of E_g differ between V_1 and V_2 : then either it is a change of e_1, or (exclusive) it is a change of e_2. There is therefore one single zero-crossing which can be labeled $(e_1 = 0)$ or $(e_2 = 0)$. The principal direction orientation, with the orientation of the segment $\vec{V_1V_2}$ and the signs of the extremality for V_1 and V_2 determines also if this point is a local maximum or minimum of the curvature (see [15]).

- **Case 2 :** If the signs of E_g are the same for V_1 and V_2, then there is two subcases. There is either no zero-crossings of e_1, e_2 (2.1), or there is a zero-crossing of e_1 and a zero-crossing of e_2 simultaneously (2.2).

- **Subcase 2.1 :** We are done. But in fact, there could be two possibilities. Either there is really no zero-crossings at all, or there are two, but due to a change of orientation too swift, the orientation of the principal directions which have been used are not compatible. Our algorithm is therefore not quite exact (with respect to the true E_g zero-crossings topology). However this error is pretty rare : each surface cell being entirely inside a grid cell, the probability of having an incompatible orientation along a segment is very small, even when the grid cell contains an umbilic point. However, the "inner" topological coherence of the wire mesh is preserved.

- **Subcase 2.2 :** In that case, either this is due to an incompatible orientation (rare) or there are really two distinct zero-crossings. We have however a way to decide between those two solutions, by computing first the two zero-crossings points ($e_1 = 0$) and ($e_2 = 0$) and then the value of E_g in a point V_g between those two points (figure 7)q. The segment $V_1, V2$ is then split into V_1V_g and V_gV_2, and the process is recursively applied to both sub-segments.

Hence the result of the computation is either $0, 1$ or 2 zero-crossings points, each being labeled $e_1 = 0$ or (exclusive) $e_2 = 0$, and labeled also local maximum or minimum. With some usual precautions relative to singular situations, the algorithm is also deterministic, that is, gives exactly the same points (a "floating point" equality) for the two adjacent cells sharing V_1V_2. As we have seen, the algorithm is not quite exact, some zero-crossing points $E_g = 0$ can be missed (2.1), but always in pairs, which doesn't perturbate the inner coherence of the wire mesh. However, no false $E_g = 0$ are generated, because E_g is really a local invariant, without orientation problems.

The only way to improve this algorithm is to subdivide each segment of the surface cell with a thiner grid, but it never ensure that two extremely close $Eg = 0$ have not been missed.

4.2 The cell processing

We can apply the edge processing to all the successive edges of the cell, which leads to an ordered list of $E_g = 0$ points, labeled $e_1 = 0$ or $e_2 = 0$. The number n of generated $E_g = 0$ points is even, because E_g is a continuous function along a closed loop (the surface cell edges).

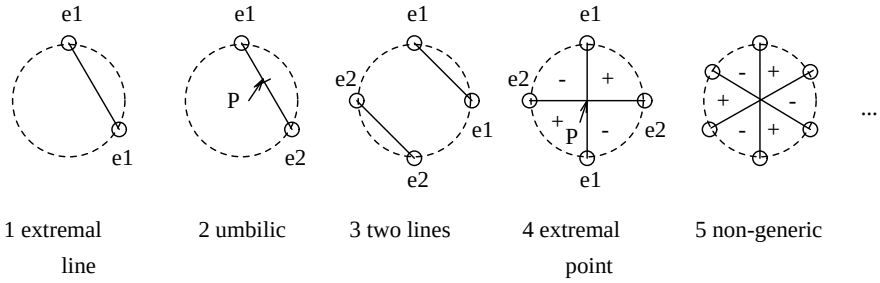

Figure 8: The processing of a cell of the iso-surface

Sophisticated algorithms can be applied to process the cells of the surface, and, with recursive subdivisions, solutions closer and closer to the true ($E_g = 0$) can be found.

In the present paper, we give only a simple way to achieve this processing which ensure the desirable topological properties for the extremal mesh : that all the vertices (of the extremal mesh, not of the surface cells) are reached by an even number of edges (of the extremal mesh) at least equal to 2 (illustrated in figure 8).

The simpler way would be to compute the barycenter P of all the points ($E_g = 0$) of the surface cell, and create segments linking them all to P (remember that we are already inside a grid cell). We reserve this drastic solution to $n \geq 6$ (case 5, figure 8) which corresponds probably to non-generic umbilics points, similar to those found at the vertices and face centers of the rounded cube (figure 5). P is labeled "non-generic umbilic". We distinguish then between the cases $n = 2$ and $n = 4$, which are the most common.

- $n = 2$ (points P_1, P_2) : if the two ($E_g = 0$) are of the same type (both $e_1 = 0$ or both $e_2 = 0$), then we create the segment P_1, P_2, and we are done (case 1). This case is the most frequent one. If the types are differents, this means that we have detected an umbilic point P between P_1 and P_2, which we can determine precisely with recursive subdivision (case 2). We give a special "umbilic" label to P, and the segments P_1P and PP_2 are created.

- $n = 4$ (points P_1, P_2, P_3, P_4) : if two consecutive points are of one kind (let say P_2, P_3), and the two other of the other kind (let say P_4, P_1), then we create two new segments P_2, P_3 and P_4, P_1, and we are done (case 3). If the opposite points are of the same type, two by two (let say P_1, P_3 and P_2, P_4, then we have detected an extremal point P, being the intersection of P_1, P_3 and P_2, P_4 (case 4). This point receives a special "extremal point" label, and we create the four segments P_1P, P_2P, P_3P, P_4P

We have not gone into the finer details of how we solve the singularities (for example, when $E_g = 0$ at a vertex of the iso-surface cell). It can be done in the same way than for the Marching Lines algorithm, as shown in [15].

4.3 The whole extremal mesh

The application of the algorithm is independent from cell to cell, and therefore independent for each cell of the 3D grid. However, the complete set of points and segments which are generated forms a wire mesh where some global and local properties are preserved (we suppose that the iso-surface is strictly inside the image, which ensures that the surface is topologically complete) :

- The vertices of the mesh are labeled either "extremal point", "umbilic" or "non-generic umbilic".
- The segments produced by the local algorithm form continuous 3D lines linking two vertices of the mesh, each of those line being an edge of the mesh (the proof is similar than the proof of the correctness of the Marching Lines algorithm, detailed in [15]).
- Each vertex of the mesh is reached by an even number of edges, at least equal to 2.
- The edges of the mesh define regions of the surface, the E_g regions ($E_g \geq 0$) and ($E_g < 0$). Each vertex is shared by an even number of E_g regions, alternatively $E_g \geq 0$ and $E_g < 0$.
- The end points of the segments (P_1P_2) which constitute the edges of the mesh are either vertices of the mesh, or linearly interpolated zero-crossings of the Gaussian extremality in the edges of the 3D grid. This

proves that, the thiner the grid is, the closer we are from the true solution $E_g = 0$.

Any correct implementation of an algorithm extracting the extremal mesh should ensure those properties, because these are direct consequences of the properties of the Gaussian extremality.

5 Experimental results

5.1 Synthetic 3D images

We have extracted the extremal mesh from a variety of 3D synthetic or real images. Figure 10 represents iso-surfaces and lines extracted from the 3D image of an ellipsoidal function (we took care to generate randomly oriented objects, to avoid simplifications). Upper left represents the sign of the extremality of the largest curvature of the surface, upper right is the sign of the second extremality. Note that in both images, there are extra zero-crossing, corresponding to the orientation problems discussed previously. Bottom left is the sign of the Gaussian extremality, which overcomes all orientation problems, and bottom right is the complete extremal mesh.

Extremal points and umbilics are represented with white dots, and the maximal lines are lighter than the minimal lines. The largest curvature extremal lines are displayed in orange, and the second curvature extremal lines in blue. The same convention is used in images 11 and 10. Note that for the rounded cube, the EPs in the middle of the edges of the cube are perfectly well defined and extracted, on the contrary to the umbilics corresponding to the vertices of the cube, which should be single vertices and face centers of the mesh shared by 6 or 8 edges. This type of umbilics corresponds to instable singularities, in real objects, three edges never merged *exactly* at the same point. Our simple algorithm detects complicated patterns there.

More evolved methods could be designed to handle the instable singularities, but they are unlikely to be exhaustive (a carved diamond has an umbilic point, its tip, with an arbitrary number of edges arriving at this point). On the contrary, the proposed algorithm handles correctly the stable singularities, which are the extremal points, and generic umbilics (such as the "lemon" umbilics of the ellipsoid).

5.2 Real 3D images

Figure 12 is an iso-surface of a 3D scan of a real skull (which has been reduced to 90^3 voxels, because of our limited visualization tool). The sign of the Gaussian extremality E_g is displayed. Remember that the sign of E_g is opposite with a plane symmetry. As a skull is symmetric with respect to a plane in essence, we can verify this property in the image : the E_g regions of the left and right sides are very similar in shapes, and antisymmetric in colors.

We possess two 3D scanner images of the same skull. The skull has been displaced of more than 10 voxels in translation and about 20 degrees in rotation (around two axes), to simulate two very different acquisitions. We have extracted the extremal mesh from both images, and registered the two meshes using their extremal points, as described in [12]. The left column represent the two superimposed extremal meshes, whereas in the right column those two meshes are slightly shifted. This is the best way that we found to make visually apparent how similar the two extremal meshes are. The bottom images correspond to a zoom of about 5 of the extremal mesh. On the left, almost each line has a corresponding line into the other mesh, with the same color, which shows that the label associated with those lines is also very significant. The bottom right image (the two meshes are shifted) helps also to appreciate the similarity of the two meshes of the left image. Those images are however poor ways for the visualization of the extremal mesh : we can explore it much more efficiently with an interactive (stereoscopic) visualization tool.

5.3 Quantitative measurements

For the two skulls, we have computed for each principal type of extremal lines and extremal points the distances between the two registered extremal meshes. We give in figure 9 the total number of points which compose the extremal lines ($nb1$ and $nb2$), the percentage of points which has a corresponding point of the same kind into the other mesh, at a distance lesser than one voxel : *matched*, and for those matched points, the standard deviation in voxel : *dev*. We give the same values for the extremal points, generic and non-generic umbilics. The "density" of the extremal mesh in the 3D image

	$nb1$	$nb2$	*matched*	*dev* (voxel)
max largest EL	5965	5749	77 %	0.31
min largest EL	6022	5960	78 %	0.30
max second EL	7959	7846	72 %	0.34
min second EL	7878	7906	72 %	0.33
extremal pts	1332	1324	66 %	0.41
umbilics	5097	4920	59 %	0.48
non-gener.	100	101	17 %	0.39

Figure 9: Standard deviation of the extremal mesh between the two acquisitions, and after registration.

of voxels is about 2 %, and about 70 % of the extremal mesh is stable with a precision better than one voxel, which is a remarkable result indeed !

6 Applications and open problems

6.1 Theoretical implications

Hence there are some fields of differential geometry, still relatively unexplored, which contain a great potential for theoretical investigations. The extremal mesh and the Gaussian extremality are notions which appear only one differential order deeper that the classical notions of surface curvatures. We have listed some theoretical questions to investigate (which are or would be solved by people perhaps more involved in the theoretical aspects of geometry than we are).

- Is it possible to define the Gaussian extremality E_g for umbilic points ?
- In that case, are all those umbilic points always part of the extremal mesh ? (equivalent to $E_g = 0$ at umbilic points). If this is true (this is the case for the ellipsoid), then the extraction of the extremal mesh incorporates also the automatic detection of the generic umbilic points, which supersedes previously proposed methods.

- What is the behavior of the extremal mesh in scale space ?
- What are the connections between the extremal mesh, and the 3D extension of the Medial Axis Transform (described for example in [7]).

6.2 Practical applications

Many people, including ourselves, have tried to use the invariance properties of the Gaussian and mean curvatures, or of the extremal curvatures, to define topological graphs in surfaces. The aim was to use those graphs for recognition tasks : due to the topological and geometrical instabilities of the region boundaries of those values, we all failed to apply this to real data.

For example, the zero crossing lines of the Gaussian curvature (the parabolic lines) separate regions of the surface which are rather differents geometrically ("saddle" like regions from "egg" like regions). In practice, the parabolic lines are extremely instable in position. For the case of our skull, they are completely unuseful, even for the registration of the same skull.

On the contrary, the extremal mesh is very stable geometrically (more precise than one voxel in position for 70% of it). Because it incorporates all the properties of the extremal lines and points into a synthetic representation, it can already be used for the automatical registration when the same object is scanned. The next question is about the stability of the extremal mesh from a topological point of view, which is a requisite before trying inter-patient registration.

Also, the extremal mesh may seem too complicated to work with in practice. We need to find ways to simplify this 3D surface graph, without loosing its topological and geometrical invariance properties. As we have listed theoretical issues, we list now also some possible applications of the extremal mesh :

- Mono-patient registration (already feasible)
- Inter-patient registration : In the case of skulls, we have already been able to do this with a small subset of the extremal lines in preliminary experiments (simplified versions of the EM would help there).
- Anatomical atlases : if we are able to perform the registration of the extremal meshes of two patients, we can then study the statistical va-

riations of the extremal mesh from patient to patient, and build an average representation of it (atlases).

- Surface reduction : a simplified representation of the EM which would keep the crest lines (they are part of it) unchanged could be a way to reduce the number of primitives (for example triangles) necessary to represent the surface, without removing the "sharp" details : the crest lines.

- Surface reconstruction : starting with only the extremal mesh, we can study methods to reconstruct a complete surface from it. Each cell of the mesh is very simple, because by definition it doesn't contain sharp edges. Such a method has been proposed by Dr Cutting et al. in [1], for the case of graphs in the surface, composed with ridge lines and geodesic lines, manually extracted from CT scans. They show a direct application of this graph to the simulation of cranio-facial surgery.

- Automatic surface parametrization : as the surface is separated into very simple regions by the extremal mesh, we can look for a parametrization of each cell with the lines of curvature, which would give a complete parametrization of the surface, *invariant* to rigid transforms.

7 Conclusion

We hope to have convinced the readers that the extremal mesh is a decomposition of the curved surfaces as natural as the decomposition of the polyhedral surfaces into faces, edges and vertices. The extremal mesh is invariant to rigid transforms, and extremely stable, even when extracted from real 3D images. The good topological properties of the extremal mesh derive from the properties of a new local operator, the Gaussian extremality E_g, which is invariant to positive isometry, opposite with negative isometry, and invariant to the surface orientation. We have proposed a local algorithm to extract the extremal mesh in practice, which is an improvement of the Marching Lines algorithm, and we have tested this algorithm with synthetic and real data. At last, we proposed a list of open questions and possible applications of those two new notions, whose answers could change in many ways

our understanding of the curved surfaces.

Acknowledgment

I wish to thank Nicholas Ayache, Olivier Monga, Alexis Gourdon and the other members of the Epidaure research team for stimulating discussions about the differential geometry of surfaces. I wish to thank General Electric France, who provided the two scans of the skull. Part of this study has been supported by the Esprit Basic Research Action VIVA. Thanks also to Digital Equipment who provided us with fast computers.

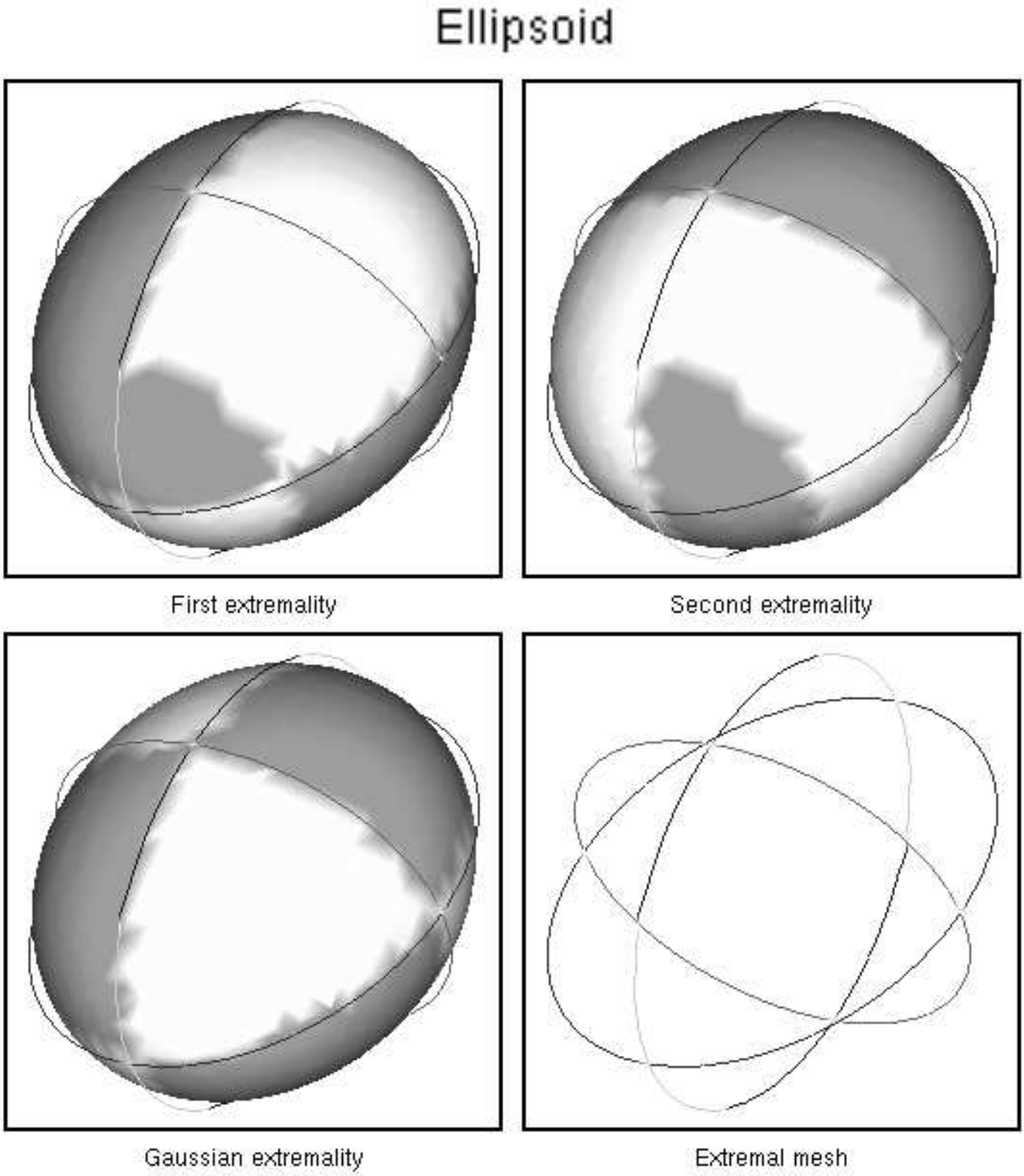

Figure 10: The two extremalities, Gaussian extremality and extremal mesh of an ellipsoid

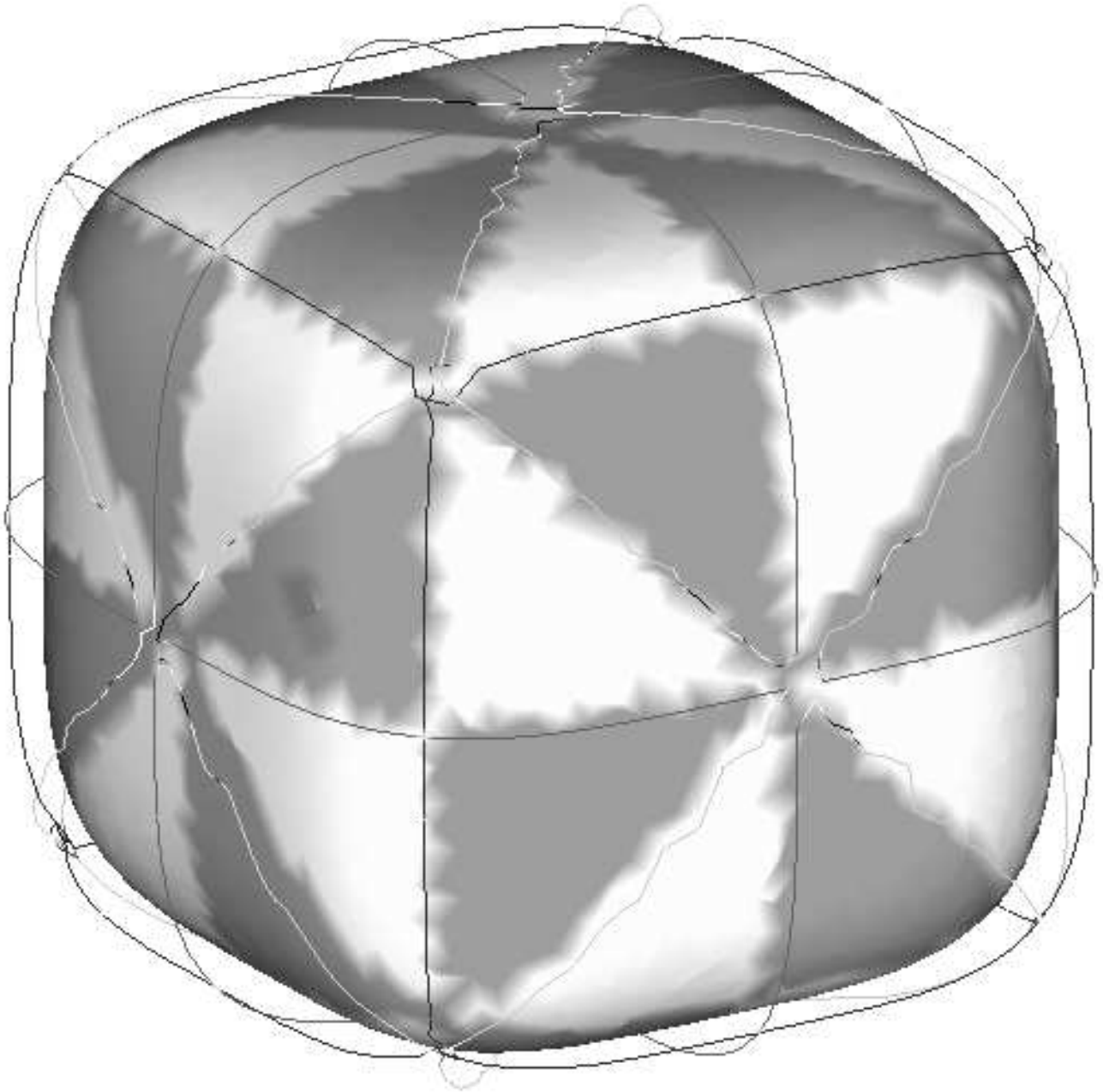

Figure 11: The measured Gaussian extremality and extremal mesh for a rounded cube

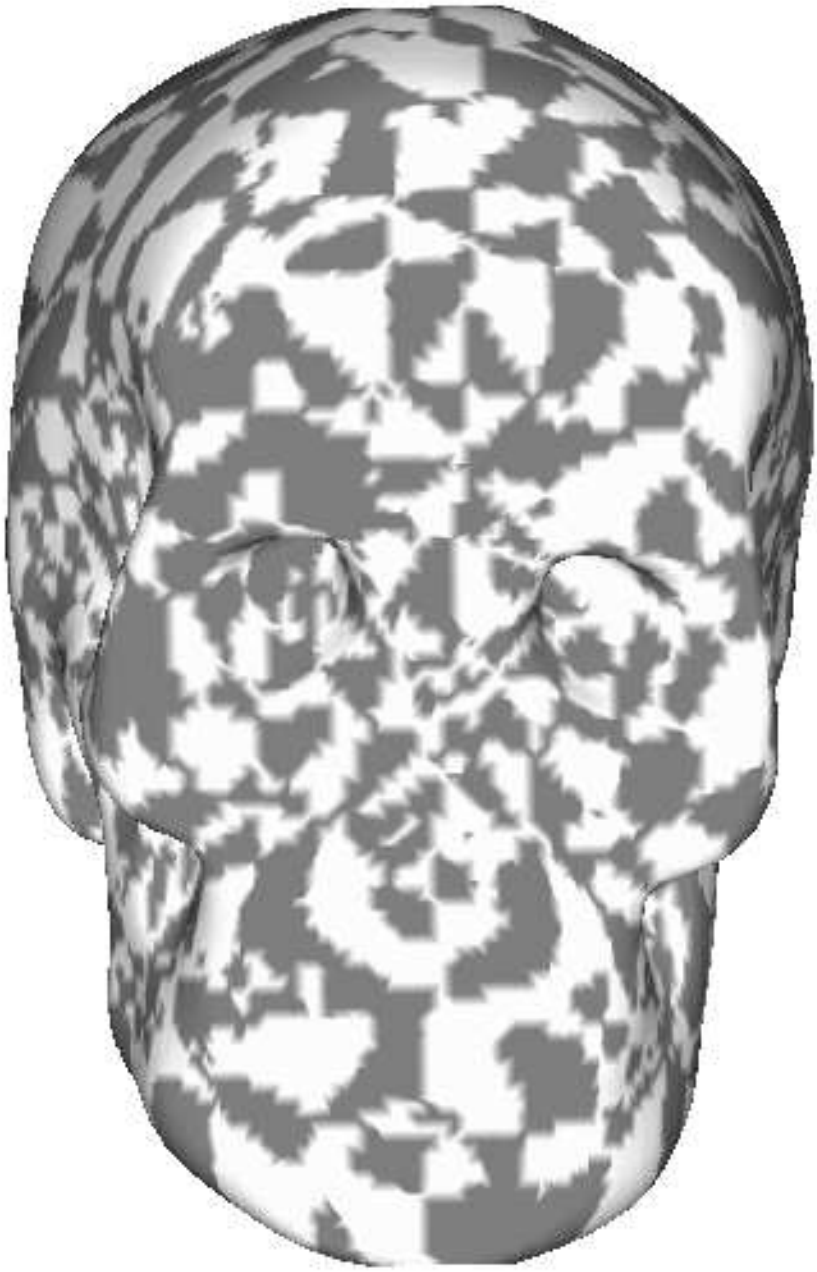

Figure 12: The Gaussian extremality E_g of a real skull. Note the antisymmetric nature of E_g

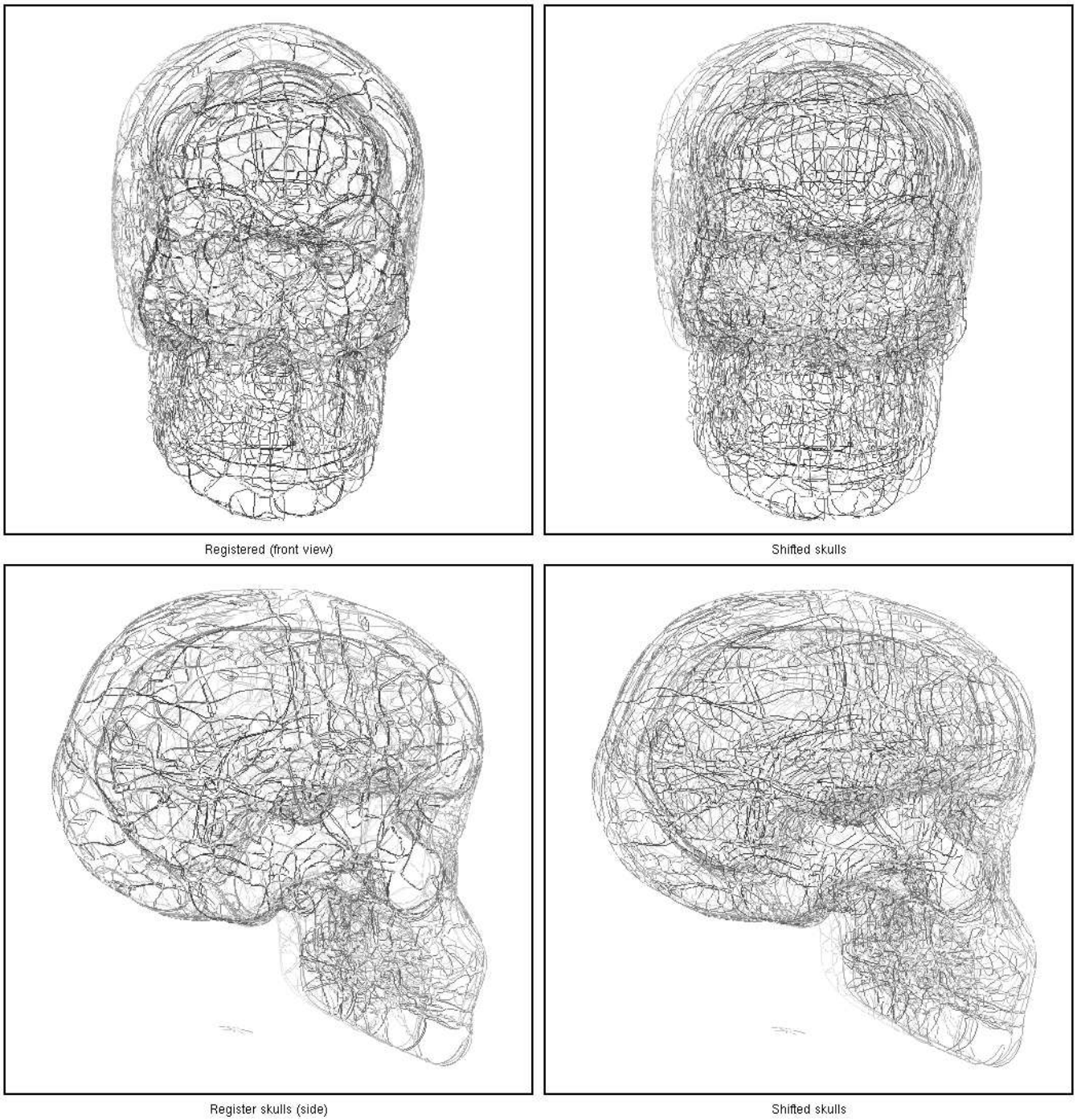

Figure 13: The superimposed extremal mesh of two acquisitions of the same skull (left). Slightly shifted (right)

References

[1] Court B. Cutting, Fred L. Bookstein, Betsy Haddad, David Dean, and David Kim. A spline-based approach for averaging three-dimensional curves and surfaces. *SPIE, Mathematical methods in Medical Imaging II*, 2035:30–44, 1993.

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

[3] Jan J. Koenderink. *Solid shape.* The MIT Press, 1990.

[4] Wiliam E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface reconstruction algorithm. *Computer Graphics*, 21(4), July 1987.

[5] O. Monga and S. Benayoun. Using differential geometry in r^4 to extract typical surface features. *IEEE Conference on Computer Vision and Pattern Recognition*, June 1993.

[6] 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).

[7] B.S. Morse, S.M. Pizer, and Liu A. Multiscale medial analysis of medical images. *IPMI'93, 13th international Conference, Springer-Verlag*, Lecture Notes in Computer Science 687:112–131, 1993.

[8] Ian R. Porteous. The normal singularities of surfaces in r^3. *Proceedings of Symposia in Pure Mathematics, American Mathematical Society*, 40:379–393, 1983.

[9] Ian R. Porteous. Ridges and umbilics of surfaces. *The Mathematics of Surfaces, edited by R.R. Martin, (Clarendon Press), Oxford*, II:447–458, 1987.

[10] Peter T. Sander and Steven W. Zucker. Singularities of principal direction fields from 3-d images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(3):309–317, March 1992.

[11] Gabriel Taubin. An improved algorithm for algebraic curve and surface fitting. *4th int. conf. on Computer Vision*, May 1993.

[12] J-P Thirion. New feature points based on geometric invariants for 3d image registration. *INRIA research report*, (1901), April 1993.

[13] J-P Thirion and S. Benayoun. Image surface extremal points, new feature points for image registration. Technical Report 2003, INRIA, July 1993.

[14] J-P. Thirion and A. Gourdon. The 3d marching lines algorithm and its application to crest lines extraction. *rapport de recherche INRIA*, (1672), May 1992.

[15] J-P. Thirion and A. Gourdon. The 3d marching lines algorithm : new results and proofs. *rapport de recherche INRIA*, (1881), March 1993.

[16] G. Wyvill, C. McPheeters, and C. Wyvill. Data structures for soft objects. *Visual Computer*, 2:227–234, 1986.

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