Crest lines extraction in volumic 3D medical images : a multi-scale approach

Olivier Monga, Richard Lengagne, Rachid Deriche*

INRIA, Domaine de Voluceau - Rocquencourt
BP. 105, 78153 Le Chesnay Cedex, France
email: om@bora.inria.fr

Abstract

Recently, we have shown that the differential properties of the surfaces represented by 3D volumic images can be recovered using their partial derivatives. For instance, the crest lines can be characterized by the first, second and third partial derivatives of the grey level function $I(x, y, z)$. In this paper, we show that :

- the computation of the partial derivatives of an image can be improved using recursive filters which approximate the Gaussian filter,
- a multi-scale approach solves many of the instability problems arising from the computation of the partial derivatives,
- we illustrate the previous point for the crest line extraction (a crest point is a zero-crossing of the derivative of the maximum curvature along the maximum curvature direction).

We present experimental results of crest point extraction on synthetic and 3-D medical data.
keywords : *volumic 3D medical images, surface modelling, curvatures, crest lines, multi-scale derivation, recursive filtering.*

1 Introduction

Volumic 3D images are now widely distributed in the medical field [8, 18, 7, 1]. They are produced from various modalities such as Magnetic Resonance Imagery (MRI), Computed Tomography Imagery (CT), Nuclear Medicine Imagery (NMI) or Ultrasound Imagery (USI). Such data are represented by a discrete 3D grey level function $I(i, j, k)$ where the high-contrast points (3D edge points) correspond to the discrete trace of the surfaces of the geometrical structures [13, 12, 21]. A motivating issue is then to extract typical features of these surfaces. The most natural way is to look for differential

*At INRIA Sophia-Antipolis, 06565 Valbonne Cedex, France.

Euclidean surface invariants such as : curvatures, crest lines, parabolic lines, umbilic points... [10, 16, 15], [17, 9, 11, 2, 19, 14]. Recently, we have shown that the differential properties of a surface defined by an iso-contour in a 3D image can be recovered from the partial derivatives of the corresponding grey level function [11]. In [11] crest lines are extracted using first, second and third order partial derivatives provided by 3D Deriche filters [12, 13]. The critical point of this approach also studied in [19] is the stability of expressions including second and third order partial derivatives such as the "extremality criterion" defined in [11, 19].

In this paper we propose recursive 3D filters to improve the computation of partial derivatives and also a multi-scale approach to extract the zero-crossings of the extremality criterion.

In Section II, we point out that derivative filters coming from isotropic (rotation invariant) smoothing filters should be used to ensure the Euclidean invariance of the curvatures. Then we derive an algorithm to compute first, second and third order partial derivatives of a 3D volumic image. These derivatives are used to obtain curvatures invariant by rigid motion (Euclidean invariant).

Section III deals with the computation of the curvatures of the surfaces traced by the iso-contours (3D edge points) from the partial derivatives of the image (for instance provided by the previous method). This section recalls the main results of the reference [11] and shows the problems induced by a single scale filtering.

In Section IV, we propose to use different widths of filters to compute the curvatures. This leads to a multi-scale curvature computation scheme where the scale is the width of the filters. We apply this principle to track the zero-crossings of the derivative of the maximum curvature points along the maximum curvature direction (extremality criterion) which correspond to the crest points. The zero-crossings coming from the different scales are merged using a valuated adjacency graph. We propose some simple and efficient strategies to extract stable zero-crossings from this graph.

In Section V we present experimental results obtained on synthetic and real data (CT and MR 3D images). We show that our approach combining a multi-scale scheme and also the use of better filters provides reliable crest lines even for noisy data.

2 Computation of the partial derivatives of a 3D image using linear filters : approximation of the Gaussian and of its derivatives using recursive separable filters

Recently R. Deriche has introduced recursive filters to approximate the Gaussian filter and its derivative [5]. First of all, we show the advantage of using such filters to compute differential Euclidean invariants. We recall the main results reported in [5]. Then, we extend Deriche's work to 3D and to the computation of third order derivatives. We also show how to normalize these filters in order to obtain coherent values for first, second and third order derivatives. Finally we develop an efficient algorithm to compute the partial derivatives of a 3D image.

Let $I(x, y, z)$ be a 3D image.

We are looking for the partial derivatives of $I(x,y,z)$: $\dfrac{\partial^n(I(x,y,z))}{\partial x^m \partial y^p \partial z^q}$; $n = m+p+q$ that we represent using the subscript notation : $I_{x^m y^p z^q}$ (we will write only the variables, the power of which is not zero, for instance $I_{x^1 y^0 z^0}$ becomes I_x).

If $f(x,y,z)$ is the impulse response of a smoothing filter, the restored image I_r is equal to $I * f$, where $*$ is the convolution product. Classically, using the properties of the convolution product we obtain

$$\frac{\partial^n I_r}{\partial x^m \partial y^p \partial z^q} = \frac{\partial^n (I * f)}{\partial x^m \partial y^p \partial z^q} = I * (\frac{\partial^n f}{\partial x^m \partial y^p \partial z^q})$$

Then the impulse response of the filter which computes $I_{x^m y^p z^q}$ is $\dfrac{\partial^n f}{\partial x^m \partial y^p \partial z^q}$.

We develop a popular scheme which reduces the search of derivative filters of any order to the search of a smoothing filter [3, 5, 4, 6]. The question is now : what are the suitable properties for our smoothing impulse response if we are interested in the computation of Euclidean differential invariants ? The response is well-known and is that the impulse response of the smoothing filter should be isotropic.

Figure 1 shows the interest of using an isotropic filter when computing the partial derivatives; we compare the curvatures at the edges of a smoothed square and those of the image of this square when applying a 45 degree-rotation. The derivatives are successively computed with the (anisotropic) Deriche filter and with the approximation of the (isotropic) Gaussian filter.

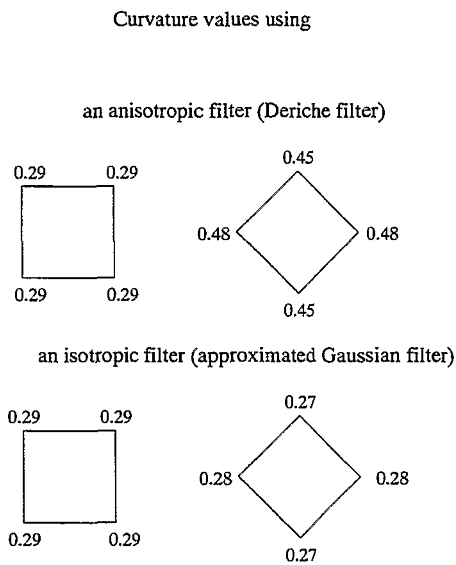

Figure 1: Comparison between curvatures from Gaussian and Deriche filters

The previous result clearly shows the great interest of computing the partial derivatives of an image using filters derived from an isotropic smoothing impulse response. Otherwise we can obtain gradient [12], Laplacian or curvatures [11] which are not invariant by a rigid motion. On the other hand, we also take interest in using separable recursive filters in order to obtain a reasonable computational cost. A way to join these two antagonist points is to use the recursive approximation of the Gaussian filter (the only separable non trivial smoothing filter) introduced by R. Deriche in the recent reference [5].

We use the main result of the reference [5]. The 1D Gaussian smoothing filter is given by :

$$g(x) = e^{-\frac{x^2}{2\sigma^2}}$$

Using Prony's method, the positive and negative part of g and of its normalized derivatives of first and second order can be approximated (with a normalized square error of about $\epsilon^2 = 10^{-6}$) by a 4th order Recursive operator (IIR)(see Appendix A) :

$$h(x) = (a_0 \cos(\frac{\omega_0}{\sigma}x) + a_1 \sin(\frac{\omega_0}{\sigma}x))e^{-\frac{b_0}{\sigma}x} + (c_0 \cos(\frac{\omega_1}{\sigma}x) + c_1 \sin(\frac{\omega_1}{\sigma}x))e^{-\frac{b_1}{\sigma}x} \quad (1)$$

Therefore, the 1D Gaussian filter and its first and second order derivatives can be recursively implemented. Using the separability property, we derive directly the recursive filters to compute the first and second order derivatives of a 2D image.

Using the separability of the filters we extend directly the previous scheme to the 3D case. Following the notations of Section II this amounts to setting :

$$f(x, y, z) = g(x)g(y)g(z)$$

We also extend this filtering scheme to the third order derivative case. We develop a set of recursive filters approximating the Gaussian and its derivatives which can be used to compute the first, second, and third order derivatives of a 3D image.

We stress that a very important point not carried out in [5] is the normalization of the filters which allows to obtain coherent values for the different derivatives. Here we use the scheme presented in [11].

3 Using the partial derivatives of a 3D volumic image to extract and to characterize 3D structures

The main stages of our algorithm allowing to extract crest lines in a 3D image are :

1. Computation of the first, second and third order partial derivatives of the image $I(x, y, z)$ ($\frac{\partial^n f}{\partial x^m \partial y^p \partial z^q}$, $m+p+q = 3$) using the recursive filters defined in Section 2 for a given value of σ ;

2. Extraction of the 3D edge points using the first order partial derivatives (gradient) of I (see section 3.2) ;

3. For each point of the 3D edge map, computation of :

   - the two principal curvatures and the corresponding principal curvature directions using the formulas of [11] ;

- the extremality criterion (derivative of the maximum curvature along the corresponding principal direction) using the formulas of [11].

4. Building of a extremality criterion image $C_\sigma(x, y, z)$ such as at each edge point (x, y, z), $C_\sigma(x, y, z)$ is set to the value of the extremality criterion and to 0 otherwise ;

5. Determination of an image $Z_\sigma(x, y, z)$ set to 1 at each edge point being a zero-crossing of the extremality criterion and to 0 otherwise.

The last stage of this algorithm consists of finding the zero-crossings of a function defined on the discrete trace of a surface (traced by the 3D edge points) which is a difficult task in itself. So far, we have only implemented simple strategies to extract these zero-crossings. But, in order to be solved properly, this delicate problem needs more attention. An interesting solution can be found in [19].

Therefore, the final output of our algorithm is an image Z_σ representing the set of edge points which are zero-crossings of the extremality criterion. Each value of σ defines an image Z_σ representing the crest line for the scale defined by σ.

4 Multi-scale approach to extract crest lines in 3D volumic images

The use of the filters presented in section 2 yields to obtain curvatures invariant by a rigid motion which was not exactly the case with the filters presented in [11]. This improves the quality of the results, but it may not be enough to provide good results in noisy data. As we have seen in the previous section the result of our algorithms is an image Z_σ where the zero-crossing of the extremality criterion are marked. Z_σ defines the crest lines for the scale defined by σ (see section 2). Generally, we see that :

- for simple data, we can obtain good results using a single value for σ but we do not know how to find this value ;

- for more complex data the suitable value for σ varies depending on the area of the 3D image ;

- for noisy data, only the crest lines that can be seen using different scales define reliable features.

Therefore, similar to the edge detection [20] and to the crest line extraction in depth maps [14], it is of great interest to use a multiscale approach. Moreover the recursive implementation of our filters makes it reasonable in terms of computational cost.

In order to merge the results obtained at different scales $\sigma_i, i = 1, n$ we propose a practical and efficient data structure that we will call the Multi-scale Adjacency Graph : $G_{\sigma_1,\sigma_2,\ldots\sigma_n}$. $G_{\sigma_1,\sigma_2,\ldots\sigma_n}$ is a valuated graph built as follows :

1. each node of $G_{\sigma_1,\sigma_2,\ldots\sigma_n}$ is attached to a point (i, j, k) such that for at least one scale σ_m we have $Z_{\sigma_m}(i, j, k) = 1$;

2. the features attached to each node are :

(a) the coordinates of the corresponding 3D point $((i, j, k))$;

(b) the values of the scales for which this point is a crest point (all the σ_p such that $Z_{\sigma_p}(i, j, k) = 1$) ;

(c) the differential characteristics extracted for all the scales : principal curvatures and principal curvatures directions, value of the extremality criterion.

3. we define an edge joining two nodes of $G_{\sigma_1,\sigma_2,...\sigma_n}$ if and only if the two corresponding points are adjacent for the 26-connectivity ;

Therefore $G_{\sigma_1,\sigma_2,...\sigma_n}$ represents the results of the crest points extraction for the different scales and their spatial relationships. This data structure is particularly efficient when the stability of the crest point locations through different scales is a good selection criterion. Our experiments performed on real and synthetic data show that generally the position of the reliable crest points remain the same for different values of the scale σ (i.e. the shifts of the crest points are less than one pixel).

For instance, the following simple pruning strategy for the graph $G_{\sigma_1,\sigma_2,...\sigma_n}$ can be used :

1. select all nodes corresponding to points which are crest points for at least a given number of scales ;

2. select the connected components having at least a given number of nodes (this threshold corresponds to the minimal number of points of a crest line).

We come up with the following algorithm :

1. Computation of the zero-crossings of the extremality criterion for a given set of scales : $\sigma_1, \sigma_2, ...\sigma_n$; the result is a set of images $Z_{\sigma_1}, Z_{\sigma_2}...Z_{\sigma_n}$ (see section 3.5) ;

2. building of the multi-scale graph $G_{\sigma_1,\sigma_2,...\sigma_n}$ (see section 4.2) ;

3. pruning of $G_{\sigma_1,\sigma_2,...\sigma_n}$ to select reliable crest points.

5 Experimental results

We present experimental results obtained on synthetic and real data from the implementation of the algorithms described in the previous section

We have tested our method on two 3D X-ray scanner images of the same skull taken at two different positions. In that case, we have chosen to extract only the extrema of the maximum curvature in the maximum curvature direction. The stability of the results we obtain for a single scale illustrates the rotation invariance of our computation of the curvatures and of the extremality criterion (see section 2). We also show that the multiscale scheme allows to remove many spurious crest points. We notice that the result obtained by thresholding the maximum curvature is acceptable for some scales but depends completely on the threshold. On the other hand the results provided by the zero-crossing of the extremality criterion may seem worse for some scales but do not depend on any threshold.

We point out that the size of the convolution mask for a direct implementation of a 3D Gaussian of variance σ^2 is $(8\sigma)^3$ (for $\sigma = 4$ we obtain 8192 !). The use of recursive filters of order 4 reduces this computational cost to about 100 operations per point for any value of σ. Of course, the previous remark applies also for the derivatives of the gaussian filter. Therefore, even for a single space scheme, the recursive filtering appears as a crucial tool.

Those results show that the multi-scale approach is much more efficient if we extract the zero-crossings of the extremality criterion instead of the high maximum curvature points. The smoothing removes many spurious points in the results coming from the high maximum curvature point extraction, but makes the crest lines thicker. On the contrary, the lines coming from the detection of the zero-crossings of the extremality criterion are still precise if σ is set to a high value, even if some points are removed. We now show the images of the zero-crossings of the extremality criterion computed on the entire skull, for the positions A and B and for some values of σ.

6 Conclusion

We have presented a multi-scale approach to extract crest lines of the surfaces represented by 3D volumic images. Compared to the method described in [11] we have developed the following points :

- we show the great theoretical interest in using filters derived from an isotropic smoothing filter to compute partial derivatives of an image ;
- we propose to use recursive filters approximating the Gaussian and its derivatives to obtain differential characteristics invariant by rigid motion ;
- in order to improve the stability of the computation of the differential characteristics (curvatures, derivative of the curvature) we use a multi-scale approach.

Moreover, we present experimental results on synthetic and real data. We stress that the same sketch could be used to extract other differential singularities such as : parabolic lines, umbilic points... Besides, this methodology could also be used in 2-D images like interior scenes, to extract corners for instance.

References

[1] Nicholas Ayache. Computer vision applied to 3d medical images : Results, trends and future challenges. *INRIA Report research*, September 1993.

[2] J. Koenderink B. Romeny, L. Florack. Invariant third order properties of isophotes: T-junction detection. In *Proc. 7th Scandinavien Conference on Image Analysis, Aalborg*, August 1991.

[3] John Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, November 1986.

[4] R. Deriche. Fast algorithms for low level vision. *IEEE Transactions on Pattern Analysis and machine Intelligence*, 1989.

[5] R. Deriche. Recursively Implementing the Gaussian and Its Derivatives. In *Proc. Second International Conference On Image Processing*, pages 263–267, Singapore, September 7-11 1992. A longer version is INRIA Research Report RR-1893.

[6] Rachid Deriche. Using Canny's criteria to derive a recursively implemented optimal edge detector. *International Journal of Computer Vision*, pages 167–187, 1987.

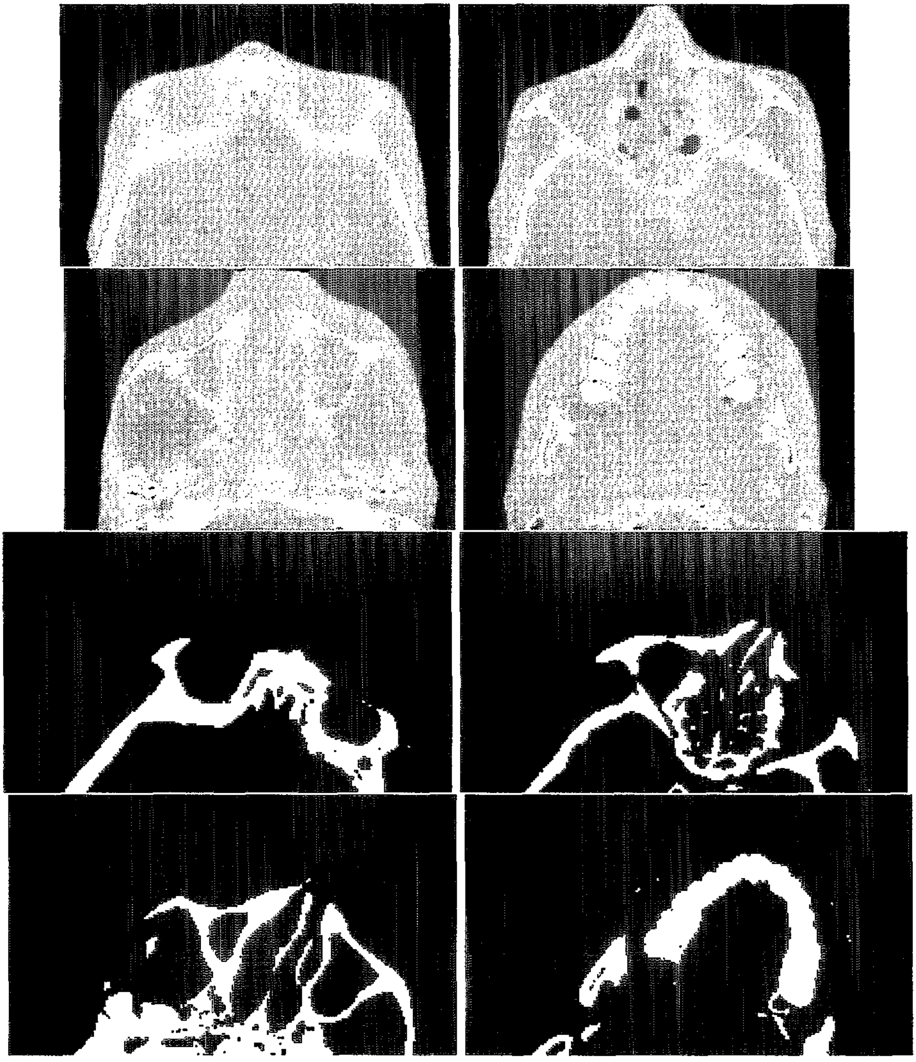

Figure 2: Cross sections of two X-scanner images of the same skull with two different positions A and B (sizes of the image : 192.128.151 for the position A and 220.128.148 for the position B). Up : cross sections corresponding to position A, bottom : cross sections corresponding to position B.

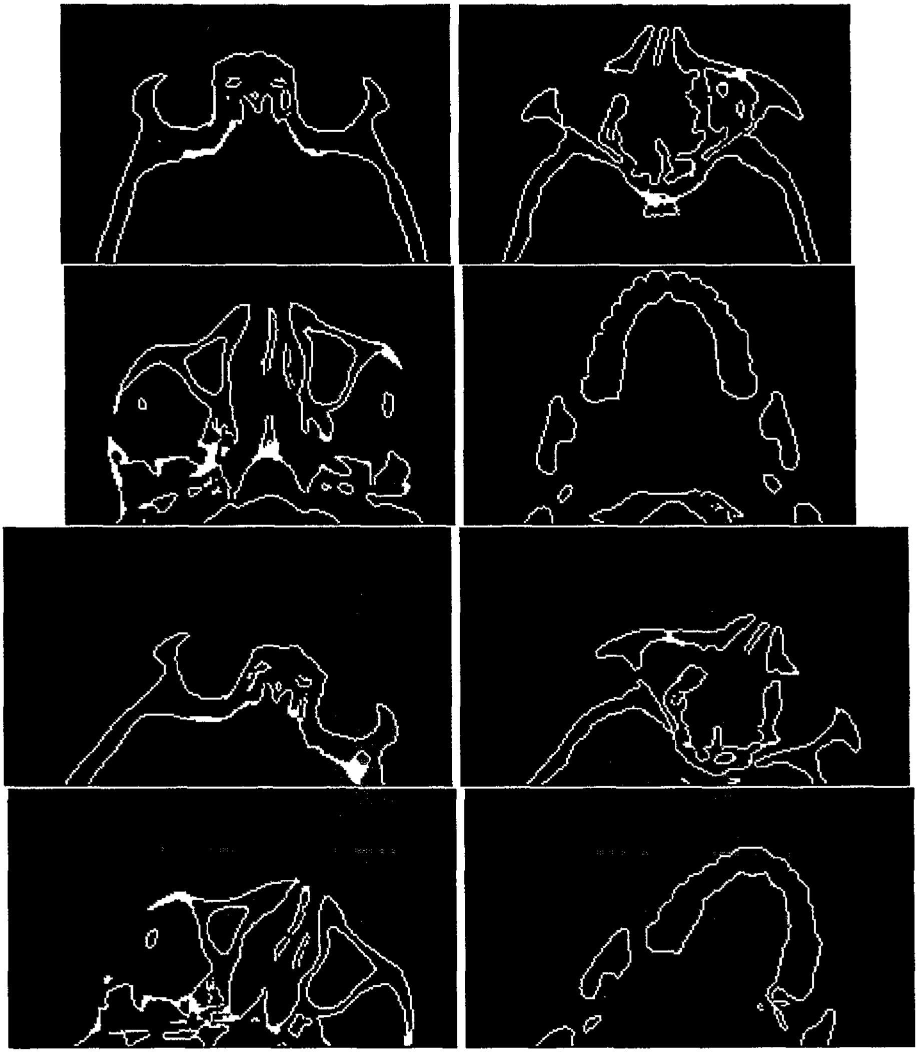

Figure 3: Cross sections of the 3D edge map corresponding to the previous figure). Up : cross sections corresponding to position A, bottom : cross sections corresponding to position B.

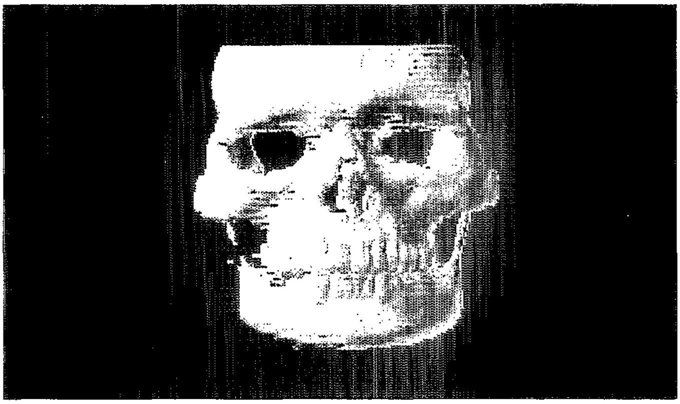

Figure 4: Perpective view of the 3D edge map for the position A

Figure 5: From left to right perpective views of : the maximum curvature, the high maximum curvature points, the zero-crossings of the extremality criterion : cross sections corresponding to position A; σ is set to 1 ; for this figure and the two following ones we only show the upper part of the skull.

Figure 6: From left to right perpective views of : the maximum curvature, the high maximum curvature points, the zero-crossings of the extremality criterion : cross sections corresponding to position A; σ is set to 3.

Figure 7: From left to right perpective views of : the maximum curvature, the high maximum curvature points, the zero-crossings of the extremality criterion : cross sections corresponding to position A; σ is set to 5.

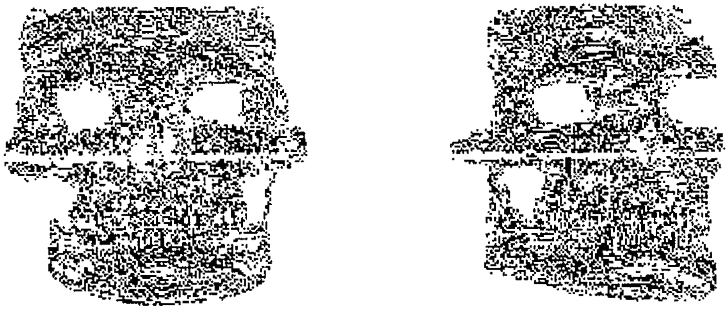

Figure 8: σ is set to 1

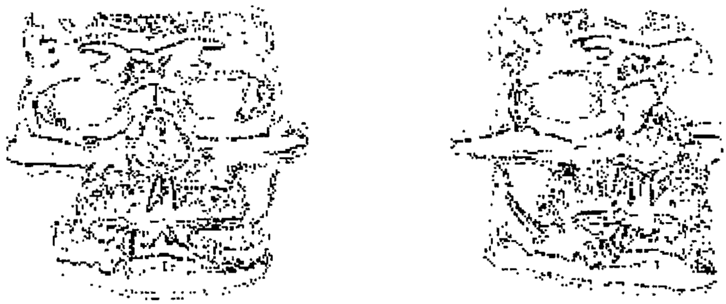

Figure 9: σ is set to 3

Figure 10: σ is set to 5

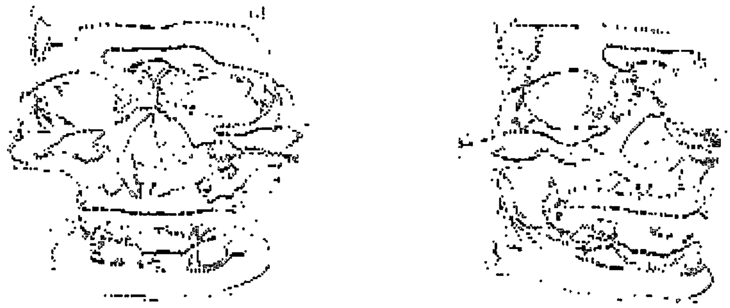

Figure 11: σ is set to 7

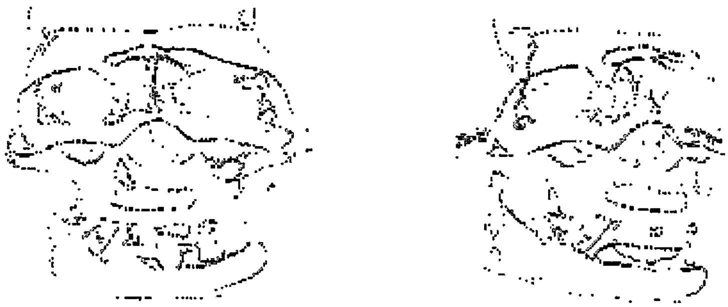

Figure 12: σ is set to 9

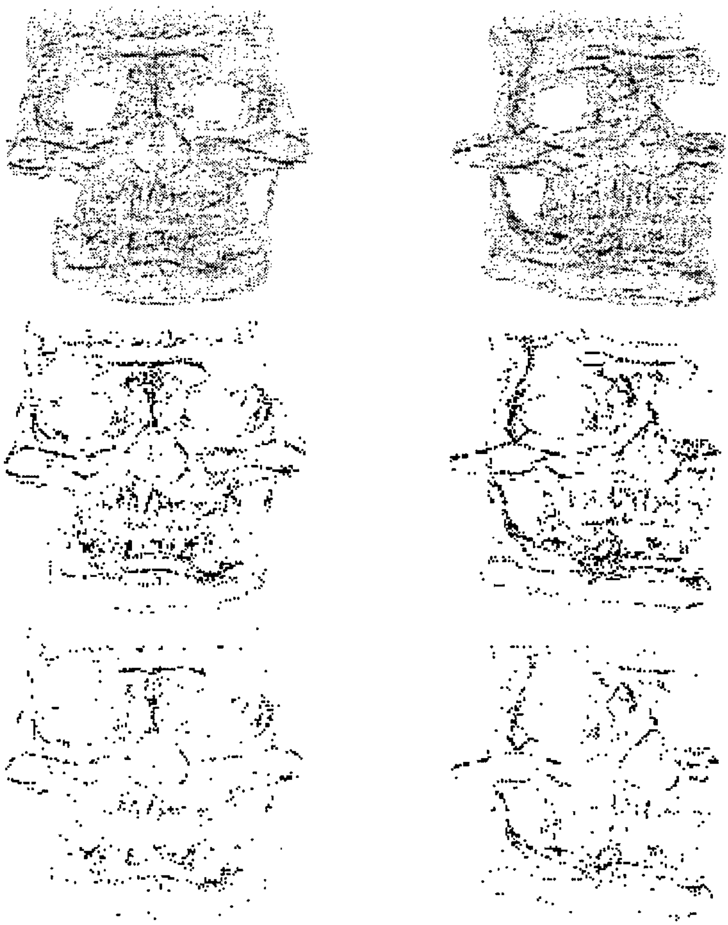

Figure 13: Up : perspective views corresponding to the positions A (left) and B (right) where the grey level is set to the number of scales such that the point is a crest point ; middle : perspective views corresponding to the positions A (left) and B (right) where only the points which are crest points for at least 4 scales are marked; bottom :perspective views corresponding to the positions A (left) and B (right) where only the points which are crest points for at least 5 scales are marked.

[7] Richard Gordon, Gabor T. Herman, and Steven A. Johnson. Image reconstruction from projections. *Scientific American*, 233(4):56–68, October 1975.

[8] Gabor T. Herman and Hsun Kao Liu. Three-dimensional display of human organs from computed tomograms. *Computer Graphics and Image Processing*, 9:1–21, 1979.

[9] Jan J. Koenderink. *Solid Shape.* MIT Press, Boston, 1990.

[10] O. Monga, N. Ayache, and P. Sander. From voxel to intrinsic surface features. *Image and Vision Computing Volume 10, Number 6*, Aout 1992. a shortened version is in IPMI'91, Lecture Notes in Computer Science 511, Springer Verlag.

[11] Olivier Monga, Serge Benayoun, and Olivier D. Faugeras. Using third order derivatives to extract ridge lines in 3d images. In *IEEE Conference on Vision and Pattern Recognition*, Urbana Champaign, June 1992.

[12] Olivier Monga, Rachid Deriche, and Gregoire Malandain. Recursive filtering and edge closing: two primary tools for 3d edge detection. *Image and Vision Computing, Vol. 9, Number 4*, August 1991. A shortened version is in proc. of ECCV'90, Lecture notes in Computer Science 427.

[13] Olivier Monga, Rachid Deriche, and Jean-Marie Rocchisani. 3d edge detection using recursive filtering: Application to scanner images. *Computer Vision Graphic and Image Processing, Vol. 53, No 1, pp. 76-87*, January 1991.

[14] Jean Ponce and Michael Brady. Toward a surface primal sketch. In *Proceedings, IJCAI*, 1985.

[15] Peter T. Sander. Generic curvature features from 3-D images. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(6):1623–1635, November 1989.

[16] Peter T. Sander and Steven W. Zucker. Singularities of principal direction fields from 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* To appear. Available as Technical Report CIM-88-7, McGill Research Center for Intelligent Machines, McGill University, Montréal.

[17] Peter T. Sander and Steven W. Zucker. Inferring surface trace and differential structure from 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(9), September 1990.

[18] Micheal M. Ter-Pogossian, Marcus E. Raichle, and Burton E. Sobel. Positron-emmision tomography. *Scientific American*, 243(4):170–181, October 1980.

[19] J-P. Thirion and Gourdon A. The 3d marching lines algorithm and its application to crest lines extraction. Technical Report 1672, INRIA, May 1992.

[20] A. Witkin. A multi-scale approach to extract zero-crossings of the laplacian. In *Proceedings of International Joint Conference on Artificial Intelligence*, 1982.

[21] S.W. Zucker and R.M. Hummel. A three-dimensional edge operator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-3(3):324–331, May 1981.