Feature Line Detection on Triangulated Meshes
A Geological Application

Dimitri Kudelski
LSIS, UMR CNRS 6168
Campus de Luminy
Case 925
France, 13288 Marseille cedex 9
kudelski@univmed.fr

Jean-Luc Mari
LSIS, UMR CNRS 6168
Campus de Luminy
Case 925
France, 13288 Marseille cedex 9
mari@univmed.fr

Sophie Viseur
GSRC, EA 4234
Université de Provence
Case 67
France, 13331 Marseille cedex 3
sophie.viseur@univ-provence.fr

ABSTRACT

We present in this article an algorithm dedicated to the feature line detection on 3D triangulated outcrop meshes. These lines corresponding to geological elements can be extracted by geometrical properties. Our approach uses differential quantities and especially principal curvatures and their derivatives. The roots of these derivatives describe particular lines called *ridge lines* for convex parts and *ravine lines* for concave parts. Then it is possible to build a set of polylines matching with ridges and ravines. Finally we apply a directional filtering to keep geological structures oriented in a particular direction. The proposed algorithm fits in a basis of a tool devoted to assist geologists during the outcrop analysis and interpretation.

Keywords

geometric modeling, differential geometry, discrete curvatures, crest lines

1. INTRODUCTION

Many works dedicated to the crest line detection have been proposed these last years (*e.g.*, [PKS$^+$01, OBS04, YBS05]). Application fields of these methods are wide and various: non-photorealistic rendering [JDA07], mesh segmentation [SF04], medical imaging [MAM95], and geology [Nam08].

Since a few years, the *LIDAR*[1] scanning technology is used to capture cliffs or, more generally, *outcrops* (*i.e.*, formations of rock strata that crop out). It generates a 3D point cloud which is afterwards triangulated to obtain a surface corresponding to the outcrop geometry. Combined with photo mapping techniques, it is possible to construct 3D models called *DOMs*[2] [BKJ05]. From this point, we propose a semi-automatic method devoted to the detection of geological objects (*i.e.*, *fractures* and *stratigraphic limits*) from outcrop surfaces. This kind of elements is characterized by differential properties explained in the following. Therefore, the extraction is a problematic similar to the *crest line* detection. However before applying a method of crest line detection to outcrop surfaces, several particular constraints must be considered:

Outcrop rugosity
The intrinsic rugosity of observed outcrops makes the generated surfaces highly complex. As the crest lines are characterized by curvature derivatives, this extraction is noise-sensitive. It is then necessary to use a noise-invariant and triangulation-invariant curvature estimator.

Results matching with observations
The presented method aims at detecting geological objects. Nevertheless, when applying traditional algorithms of crest line detection, the extracted features do not entirely correspond to elements with a geological meaning. An *a priori* knowledge is then necessary to realize a filtering to only extract targeted geological structures.

Interactivity
An additional constraint is the computational time due to the final application. The detection must be performed in a few seconds in order to keep interactivity with a real-time procedure. Moreover, this is particularly crucial as LIDAR scans often generate huge data sets which are difficult to manipulate. Because of this, we take great care to implement process with low computational time.

To understand the crest line detection problem, Section 2 describes the different criteria characterizing the geological objects. We review in Section 3 the related work established in the domains of curvature estimation and crest line detection. Then we detail each step of our approach in Section 4. Section 5

[1] *LIght Detection And Ranging*
[2] *Digital Outcrop Model*

finally presents the results obtained with our algorithm applied on LIDAR data scans.

2. CHARACTERIZATION OF GEOLOGICAL OBJECTS

Fractures are like crevices more or less opened that affect a rock mass. Stratigraphic limits of geological bodies correspond to a change of rock type. Both of these geological features are displayed along the outcrop surface because of the erosion. It leads finally to step-like or a gutter-like shapes at their location. Figure 1 represents a diagram with the different patterns of targeted geological objects. Moreover, this pattern often varies along the same fracture or strata limit. Given Figure 1, it is indeed possible to see that the expected objects (depicted by the thick dashed lines) are located in the highest concave parts of the surface. These elements have a common geometrical criterion: they define lines located in areas with high curvature. Thus crest line algorithms can be applied to achieve the detection of such objects.

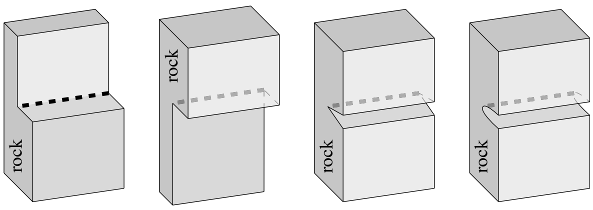

Figure 1: Diagram showing the different patterns of geological objects. The thick dashed lines illustrate the highest concave areas characterizing the expected features.

Feature lines are then defined by curvature extrema and then it corresponds to a zero-crossing of curvature derivatives. However, the rough set of crest lines extracted from a DOM does not represent the set of targeted geological objects. This is due to several factors, among which: (1) fractures often cut across strata limits. Depending on the erosion effect, an extracted crest line could then encompass features with different geological meaning; (2) the intrinsic rugosity of the rock or the variable direction of the outcrop can generate salient lines which do not represent any expected geological feature.

For these reasons, we suggest to add an *a priori* knowledge (*i.e.*, a global direction) to guide the extraction and filter feature lines. Our approach is dedicated to the detection of slightly sinuous structures. It is always the case for the fractures and very frequently for the strata limits.

The proposed method then relies on the crest line principle. It previously requires a per-vertex estimation of differential quantities. Before describing each algorithm step, the following section gives an overview of existing crest line detection techniques.

3. RELATED WORK

3.1 Curvature Estimation

Differential properties characterize the local geometry of meshes. The notion of curvature describes precisely how the surface is locally bent. These geometrical descriptors are then used since a few years and several approaches have already been proposed in this domain. Some of them are presented in the following (for additional references see [MD02, GG06]).

Continuous methods
This type of methods tends to fit locally the surface with simple primitives (*e.g.*, plane, sphere or polynomial) or parametric functions or even implicit functions. These different techniques permit an analytical computation of curvatures. For example, in [Ham93], the authors proposed to approximate locally the surface with quadratic polynomials. Alternatively in [GI04], the fitting is performed via bi-cubic polynomials. Bi-quadratic Bézier patches can also be used to fit the surface such as in [RB05].

Discrete methods
To reduce the high computational time produced by local fitting, differential operators have been proposed. In [MDSB02], the authors suggested to use a curvature estimation based on cotangent weights and Voronoï areas. In another way, the dihedral angle (*i.e.*, angle between the normals of two adjacent faces) can be used as a discriminant property to compute curvatures [CSM03]. Additionally, the curvature tensor can be estimated by studying the per-vertex normal variation such as in [Rus04, BW07].

3.2 Crest Line Detection

The properties of crest lines are widely used for their efficiency as shape descriptors. This domain has become a field of intensive researches since the last decades and several approaches have been then proposed. The first family of techniques is based on extrema searching. It can be performed either by thresholding [RKS00, SF03], curvature derivatives [CP04, OBS04, YBS05], focal surfaces [LA98, WB01, YBYS07], or discretized operators [HPW05].

The second kind of methods relies on other differential properties. The dihedral angle can be used to detect sharp features such as in [HG01, PSK+02]. Then in [GPHW05], the authors proposed to apply active contour theory stemming from image processing domain to detect characteristic lines. In addition in [LVJ05], Lee suggested to use a measure of a regional importance named *mesh saliency* based on contextual and visual criteria.

4. GEOLOGICAL FEATURE DETECTION

On the one hand, DOMs represent natural surfaces. These objects are characterized by an inherent noise due to the acquisition technology and a high intrinsic rugosity because of the surface alteration. On the other hand, due to their definition by high differential quantities, crest lines are very noise-sensitive. It is then necessary to select a robust curvature estimator. For these reasons, we chose to apply the method proposed in [GI04] considering its quality, accuracy and stable results (see [GG06]). Concerning the crest line detection method, we opted for the criteria expressed in [OBS04]. It relies on curvature derivatives and thus is scale-invariant. It is then possible to extract geological objects with different sizes.

4.1 Pre-processing Step

The inherent noise and rugosity of the data make the detection of smooth and continuous lines difficult. We then propose to use a pre-processing to increase these continuity and smoothness. Among all existing techniques, we chose to integrate a *Laplacian* smoothing (cf. Equation 1) on surface coordinates:

$$p' = p + \lambda \frac{1}{n} \sum_{i=1}^{n} (q_i - p), \tag{1}$$

where n is the number of adjacent vertices q_i to the vertex p and λ represents a step-size parameter.

Once the smoothing performed, the next step is to compute the differential quantities in order to detect the surface crest lines.

4.2 Estimation of Curvatures and their Derivatives

Several techniques of curvature estimation have been previously presented. The approach proposed in [GI04] fits locally the surface with a bi-cubic polynomial in the least-squares sense. Thus, the surface is expressed for each vertex thanks to the following equation:

$$f(x,y) = \frac{A}{2}x^2 + Bxy + \frac{C}{2}y^2 + Dx^3 + Ex^2y + Fxy^2 + Gy^3. \tag{2}$$

The *Weingarten matrix* (*i.e.*, the matrix of the second fundamental form) of the surface is therefore composed as:

$$W = \begin{bmatrix} A & B \\ B & C \end{bmatrix}. \tag{3}$$

The curvature values κ_{max} and κ_{min} (with $|\kappa_{max}| > |\kappa_{min}|$) are defined by the eigenvalues of W and the eigenvectors of W correspond to the principal curvature directions $\vec{t}_{max}$ and $\vec{t}_{min}$. To obtain the curvature derivatives, it is possible to use the coefficients D, E, F and G of Equation 2 as suggested in [YBS05]:

$$e = \frac{\partial \kappa}{\partial \vec{t}} = \begin{bmatrix} u^2 \\ v^2 \end{bmatrix}^T \begin{bmatrix} D & E \\ F & G \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \tag{4}$$

where

$$\vec{t} = (u, v) \tag{5}$$

can correspond to either $\vec{t}_{min}$ or $\vec{t}_{max}$. Consequently, two values called *extremality coefficients* (cf. [Thi96]) are then defined by:

$$e_{max} = \frac{\partial \kappa_{max}}{\partial \vec{t}_{max}} \qquad e_{min} = \frac{\partial \kappa_{min}}{\partial \vec{t}_{min}}. \tag{6}$$

These coefficients are the support for the crest line detection, as described in the next section.

4.3 Crest Line Detection

The extremality coefficients describe curvature variations and crest lines are located where curvature extrema are reached. Thus, the crest lines are characterized by:

$$e_{max} = \frac{\partial \kappa_{max}}{\partial \vec{t}_{max}} = 0, \quad \frac{\partial e_{max}}{\partial \vec{t}_{max}} < 0, \quad \kappa_{max} > |\kappa_{min}| \tag{7}$$

for the ridge lines (convex areas) and:

$$e_{min} = \frac{\partial \kappa_{min}}{\partial \vec{t}_{min}} = 0, \quad \frac{\partial e_{min}}{\partial \vec{t}_{min}} > 0, \quad \kappa_{min} < -|\kappa_{max}| \tag{8}$$

for the ravine lines (concave areas).

The curvature sign gives information about the locally convexity or concavity of the surface. Ridges and ravines are dual notions according to the surface orientation: by flipping the surface orientation, convexity and concavity are swapped as for ridge and ravine lines.

As previously mentioned, extremality coefficients as derivatives, are highly sensitive to noise. For this reason, the pre-processing of smoothing the surface geometry is applied to compute the derivatives. However, original coordinates are restored before performing the detection. In this way, noise impact is reduced and even several artifacts due to the intrinsic surface rugosity are removed while maintaining the accuracy about the locations of the extracted feature lines.

Crest line detection is performed by searching crest vertices and curvature extrema (*i.e.*, roots of curvature derivatives). Let be ε an edge composed by the vertices v_1 and v_2. A vertex is considered as a crest vertex since a set of conditions described in [OBS04] is satisfied. For the sake of clarity and simplicity, only the case of ridge vertices is explained below. As ridges and ravines are dual notions, explained conditions can be easily transposed from a ridge to a ravine detection algorithm.

First, if the angle between principal directions $\vec{t}_{max}(v_1)$ and $\vec{t}_{max}(v_2)$ is obtuse, the vector $\vec{t}_{max}(v_2)$ is flipped as the sign of $e_{max}(v_2)$. The second step is to check if there is a zero-crossing of the curvature derivative on the edge. It appears when the signs of $e_{max}(v_1)$ and $e_{max}(v_2)$ are different:

$$e_{max}(v_1) \cdot e_{max}(v_2) < 0. \tag{9}$$

Curvature must also reach a local maxima which can be verified by a derivative test:

$$e_{max}(v_1)\left[(v_2 - v_1) \cdot \vec{t}_{max}(v_1)\right] > 0. \tag{10}$$

When Equations 9 and 10 are satisfied, the coordinates of the ridge vertex are found by a linear interpolation between v_1 and v_2:

$$v_{ridge} = \frac{|e_{max}(v_2)|v_1 + |e_{max}(v_1)|v_2}{|e_{max}(v_1)| + |e_{max}(v_2)|}. \tag{11}$$

This process is applied on each edge of the mesh to obtain all the crest lines. These lines are defined by polylines built from crest vertices. Figures 2 and 3 summarize the method of crest line extraction and construction.

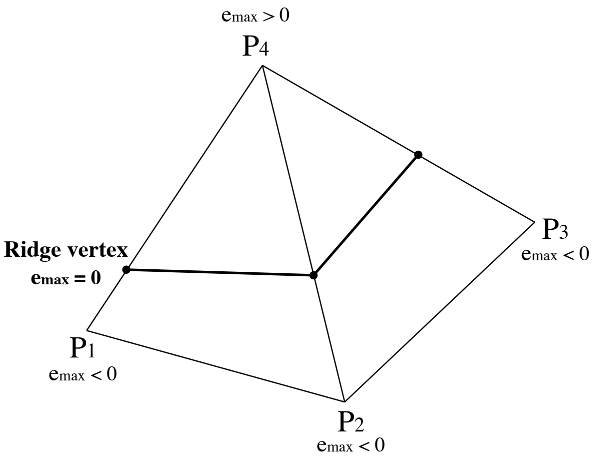

Figure 2: Process of ridge vertex extraction.

The proposed algorithm does not aim at extracting all crest lines but only geological feature lines. Thus, particular conditions must be honored during the feature extraction.

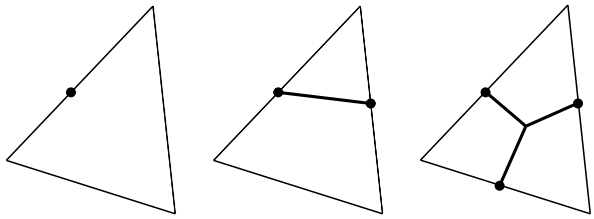

Figure 3: Construction of a feature line. On left, an isolated crest vertex can not define a line. In the middle, two crest vertices generate a straight line. Lastly, 3 crest vertices produce a T-junction between the three vertices and the triangle barycenter.

4.4 Directional Filtering

In order to keep only lines which have a geological meaning and are roughly oriented in a same user-defined direction $\vec{D}$, an *a priori* knowledge is integrated. It corresponds to a filtering process added to the detection algorithm previously described.

First, as mentioned in Section 2, only concave parts correspond to fractures or strata limits. Therefore only ravine lines characterize relevant objects. Secondly, geological structures are generally slightly sinuous. Their detection can be guided via an user-defined direction $\vec{D}$, corresponding to the rough direction of a family of targeted geological structures observed along the outcrop.

Let S be a surface of $\mathscr{R}^3$ and p a point of S. Principal directions of p are contained in a plane $\mathscr{P}$ oriented according to $\vec{N}_p$ (*i.e.,* the normal vector of p). As the shape of the geological objects can be locally described as parabolic surfaces, the curvature vector $\vec{t}_{min}$ tends to follow this shape as shown by Figure 4.

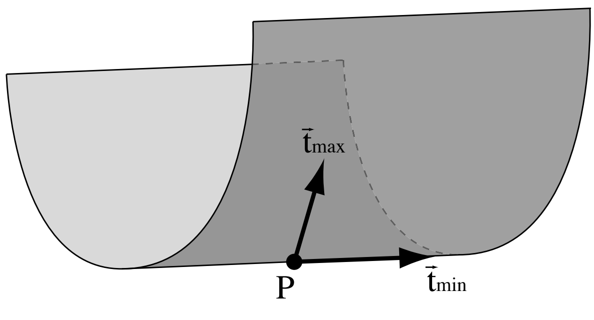

Figure 4: Principal curvature directions along a parabolic shape.

It is thus possible to use the direction of $\vec{t}_{min}$ to filter lines oriented in the same direction of $\vec{D}$. However the direction $\vec{D}$ is set globally by the user on the outcrop. The outcrop surface is not totally flat and its direction can vary locally. Thus it is not ensured that the vector $\vec{D}$ will be contained in the plane $\mathscr{P}$. Therefore, a rotation is applied to transform $\vec{D}$ into $\vec{D}'$

and to place this vector into the plane $\mathscr{P}$. This rotation has the following parameters:

$$\begin{aligned} \overrightarrow{axis} &= \vec{D} \times \vec{N}_p \\ angle &= \vec{D} \cdot \vec{N}_p. \end{aligned} \quad (12)$$

Once the rotation is applied, $\vec{D}'$ is contained in the plane $\mathscr{P}$. A projection of $\vec{D}$ onto $\mathscr{P}$ could not have been considered as it may generate a null vector $\vec{D}'$ as soon as $\mathscr{P}$ is perpendicular to $\vec{D}$.

Finally, on the edges containing a root of curvature derivative, the absolute value s' of the dot product between $\vec{D}'$ and $\vec{t}_{min}$ is computed. Therefore when both $\vec{t}_{min}$ vectors of an edge are collinear to $\vec{D}'$, the line is preserved otherwise it is removed. This step is illustrated by Figure 5.

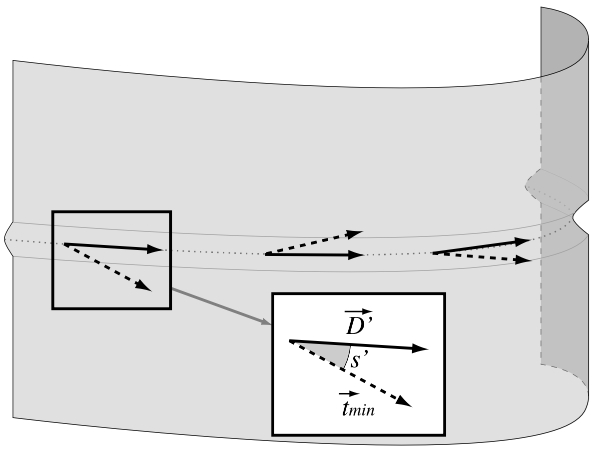

Figure 5: Directional filtering according the vectors $\vec{D}'$ and $\vec{t}_{min}$.

The direction $\vec{D}$ is specified globally by the user and corresponds to the rough direction of the expected structures. However the direction of these objects may vary locally. Thus a threshold T, ranged from 0 to 1 is applied on s' as a tolerance factor: if T is equal to 1, the vectors $\vec{t}_{min}$ and $\vec{D}'$ must be strictly collinear to keep the line and inversely if T equals 0, all the ravine lines are kept.

5. RESULTS AND VALIDATION

The proposed approach devoted to the detection of geological objects onto numerical outcrop surfaces is composed of four main operations:
- a pre-processing smoothing;
- an estimation of curvatures and their derivatives;
- a crest line extraction;
- a directional filtering.

This algorithm is dedicated to the detection of geological features (*i.e.*, fractures and strata limits) from 3D triangulated meshes built from LIDAR data points. Figure 6 shows the results obtained with different outcrop models. Figures 7 and 8 display the impact of the direction $\vec{D}$ and the threshold T onto the detection of targeted geological features.

These parameters have to be set up manually by the user. They represent an *a priori* knowledge about the targeted geological structures to interpret. The direction $\vec{D}$ can be determined by the geologists through the observation along the numerical outcrop. It may be noticed that this parameter could be also automatically deduced from a heuristic such as a *principal component analysis*. However the primary goal of the proposed approach is to assist the geologists in the outcrop interpretation. Moreover, due to the complexity of geological structure spatial organization, the full automatization of the algorithm could easily lead to several mismatch between geological reality and extracted lines which should be in fact removed *a posteriori* using manual or automated filtering. Then, the tolerance threshold T is used to constraint more or less the detection to the fixed orientation. It is set up according to the aspect of the observed limits (*i.e*, straight or slightly sinuous).

The results obtained with the presented approach match with geological objects observed on outcrops and manually modeled by geologists. We notice however that some lines are incomplete or non-significative. This is due to umbilical points (*i.e.*, points locally spherical) without principal direction.

The computational time of our algorithm is low: it only requires less than 5 seconds (in part due to the computation of curvature values and their derivatives) to detect geological features of a surface composed by about 100k triangles (computed on an Intel Core 2 Duo 2.8 Ghz).

6. CONCLUSION

Several methods of crest line detection have been proposed in the litterature. However none was directly applicable to the context of geological feature extraction from 3D digital outcrop models. By relying on existing methods, we thus present an algorithm devoted to the feature line detection from LIDAR data scan satisfying new constraints.

The proposed approach is based on the estimation of curvature values and their derivatives. The extremality coefficients are computed from curvature derivatives to obtain ridge and ravine lines. Finally, a directional filtering is applied to preserve lines with geological meaning and oriented in a particular direction.

Feature lines corresponding to fractures and strata limits are extracted. The proposed tool enables the geologists to be assisted during the outcrop interpretation stage. As mentioned previously, the obtained results match with the elements manually modeled by geologists.

This approach is promising and can be improved. We plan to add post-processing to increase the quality of results concerning, for instance, the connectivity enhancement and the artifacts removal. In addition, extracted lines are slightly sinuous which concerns most of the targeted geological objects. Though, some strata limits are actually sinuous. This requires a pertinent relaxation of the proposed directional filtering.

The feature extraction corresponding to geological objects is a first step in the outcrop interpretation workflow. The next step would be the construction, from the extracted elements, of a graph to reproduce the layout of observed geological structures.

ACKNOWLEDGMENTS

The authors would like to thank ENI S.p.A. to support this research.

REFERENCES

[BKJ05] Jerome A. Bellian, Charles Kerans, and David C. Jennette. Digital outcrop models; applications of terrestrial scanning lidar technology in stratigraphic modeling. *Journal of Sedimentary Research*, 75(2):166–176, 2005.

[BW07] Harlen Costa Batagelo and Shin-Ting Wu. Estimating curvatures and their derivatives on meshes of arbitrary topology from sampling directions. *Vis. Comput.*, 23(9):803–812, 2007.

[CP04] Frédéric Cazals and Marc Pouget. Ridges and umbilics of a sampled smooth surface: a complete picture gearing toward topological coherence. Research Report 5294, INRIA, 2004.

[CSM03] David Cohen-Steiner and Jean-Marie Morvan. Restricted delaunay triangulations and normal cycle. In *SCG '03: Proceedings of the nineteenth annual symposium on Computational geometry*, pages 312–321, New York, NY, USA, 2003. ACM.

[GG06] Timothy Gatzke and Cindy M. Grimm. Estimating curvature on triangular meshes. *International Journal of Shape Modeling*, 12(1):1–28, 2006.

[GI04] Jack Goldfeather and Victoria Interrante. A novel cubic-order algorithm for approximating principal direction vectors. *ACM Trans. Graph.*, 23(1):45–63, 2004.

[GPHW05] Y.W. Guo, Q.S. Peng, G.F. Hu, and J. Wang. Smooth feature line detection for meshes. *Journal of Zhejiang University Science*, pages 460–468, 2005.

[Ham93] B. Hamann. Curvature approximation for triangulated surfaces. *Springer Computing Supplementum*, pages 139–153, 1993.

[HG01] Andreas Hubeli and Markus Gross. Multiresolution feature extraction for unstructured meshes. In *VIS '01: Proceedings of the conference on Visualization '01*, pages 287–294, Washington, DC, USA, 2001. IEEE Computer Society.

[HPW05] Klaus Hildebrandt, Konrad Polthier, and Max Wardetzky. Smooth feature lines on surface meshes. In *SGP '05: Proceedings of the third Eurographics symposium on Geometry processing*, page 85, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.

[JDA07] Tilke Judd, Frédo Durand, and Edward H. Adelson. Apparent ridges for line drawing. *ACM Trans. Graph.*, 26(3):19, 2007.

[LA98] Gábor Lukács and László Andor. Computing natural division lines on free-form surfaces based on measured data. In *Proceedings of the international conference on Mathematical methods for curves and surfaces II Lillehammer, 1997*, pages 319–326, Nashville, TN, USA, 1998. Vanderbilt University.

[LVJ05] Chang Ha Lee, Amitabh Varshney, and David W. Jacobs. Mesh saliency. *ACM Trans. Graph.*, 24(3):659–666, 2005.

[MAM95] O. Monga, N. Armande, and P. Montesinos. Thin nets and crest lines: application to satellite data and medical images. In *ICIP '95: Proceedings of the 1995 International Conference on Image Processing (Vol.2)-Volume 2*, page 2468, Washington, DC, USA, 1995. IEEE Computer Society.

[MD02] J.-L. Maltret and M. Daniel. Discrete curvatures and applications : a survey. Rapport de recherche LSIS.RR.2002.002, Laboratoire des Sciences de l'Information et des Systèmes, 2002.

[MDSB02] M. Meyer, M. Desbrun, P. Schroder, and A.H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. *Vi-*

sualization and mathematics, 3:35–57, 2002.

[Nam08] Van Tran Nam. *Traitement de surfaces triangulées pour la construction de modèles géologiques structuraux*. PhD thesis, Université de la Méditerranée, 2008.

[OBS04] Yutaka Ohtake, Alexander Belyaev, and Hans-Peter Seidel. Ridge-valley lines on meshes via implicit surface fitting. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Papers*, pages 609–612, New York, NY, USA, 2004. ACM.

[PKS+01] D. L. Page, A. Koschan, Y. Sun, J. Paik, and M. A. Abidi. Robust crease detection and curvature estimation of piecewise smooth surfaces from triangle mesh approximations using normal voting. In *Conference on Computer Vision and Pattern Recognition*, volume 1, page 162, Los Alamitos, CA, USA, 2001. IEEE Computer Society.

[PSK+02] DL Page, Y. Sun, AF Koschan, J. Paik, and MA Abidi. Normal vector voting: Crease detection and curvature estimation on large, noisy meshes. *Graphical Models*, 64:199–229, 2002.

[RB05] Anshuman Razdan and MyungSoo Bae. Curvature estimation scheme for triangle meshes using biquadratic bézier patches. *Computer-Aided Design*, 37(14):1481–1491, 2005.

[RKS00] Christian Rössl, Christian, Leif Kobbelt, and Hans-Peter Seidel. Extraction of feature lines on triangulated surfaces using morphological operators. In Andreas Butz, Antonio Krüger, and Patrick Olivier, editors, *Smart Graphics (AAAI Spring Symposium-00)*, volume 00-04 of *Technical Report / SS / American Association for Artificial Intelligence*, pages 71–75, Stanford, USA, 2000. American Association for Artificial Intelligence, AAAI Press.

[Rus04] Szymon Rusinkiewicz. Estimating curvatures and their derivatives on triangle meshes. In *Symposium on 3D Data Processing, Visualization, and Transmission*, Sept 2004.

[SF03] G. Stylianou and G. Farin. Crest lines extraction from 3D triangulated meshes. *Hierarchical and geometrical methods in scientific visualization*, pages 269–281, 2003.

[SF04] Georgios Stylianou and Gerald Farin. Crest lines for surface segmentation and flattening. *IEEE Transactions on Visualization and Computer Graphics*, 10(5):536–544, 2004.

[Thi96] Jean-Philippe Thirion. The extremal mesh and the understanding of 3d surfaces. *Int. J. Comput. Vision*, 19(2):115–128, 1996.

[WB01] Kouki Watanabe and Alexander G. Belyaev. Detection of salient curvature features on polygonal surfaces. *Comput. Graph. Forum*, 20(3), 2001.

[YBS05] Shin Yoshizawa, Alexander Belyaev, and Hans-Peter Seidel. Fast and robust detection of crest lines on meshes. In *SPM '05: Proceedings of the 2005 ACM symposium on Solid and physical modeling*, pages 227–232, New York, NY, USA, 2005. ACM.

[YBYS07] Shin Yoshizawa, Alexander Belyaev, Hideo Yokota, and Hans-Peter Seidel. Fast and faithful geometric algorithm for detecting crest lines on meshes. In *PG '07: Proceedings of the 15th Pacific Conference on Computer Graphics and Applications*, pages 231–237, Washington, DC, USA, 2007. IEEE Computer Society.

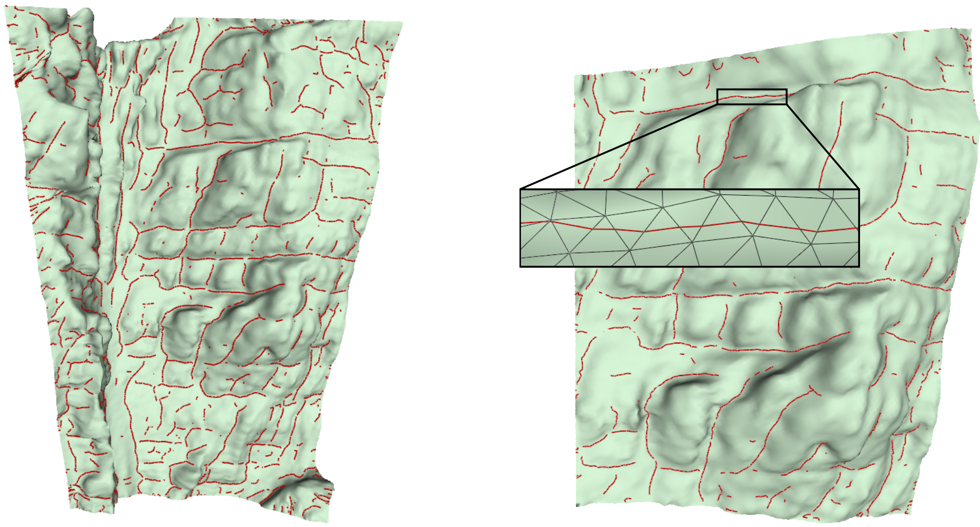

Figure 6: Application of our algorithm on LIDAR data scans without any filtering. On left, feature detection performed on the Malaval section ($\simeq$ 60000 triangles). On right, extraction of lines of the Pas-Morta section ($\simeq$ 20000 triangles).

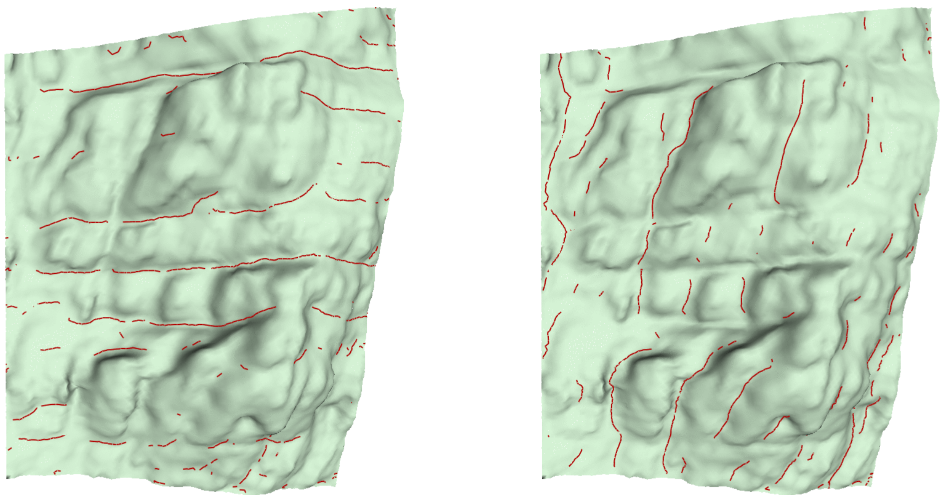

Figure 7: Application of two directional filters. The strata limits are extracted with a horizontal direction (left image) while the fractures are detected with a vertical direction (right image).

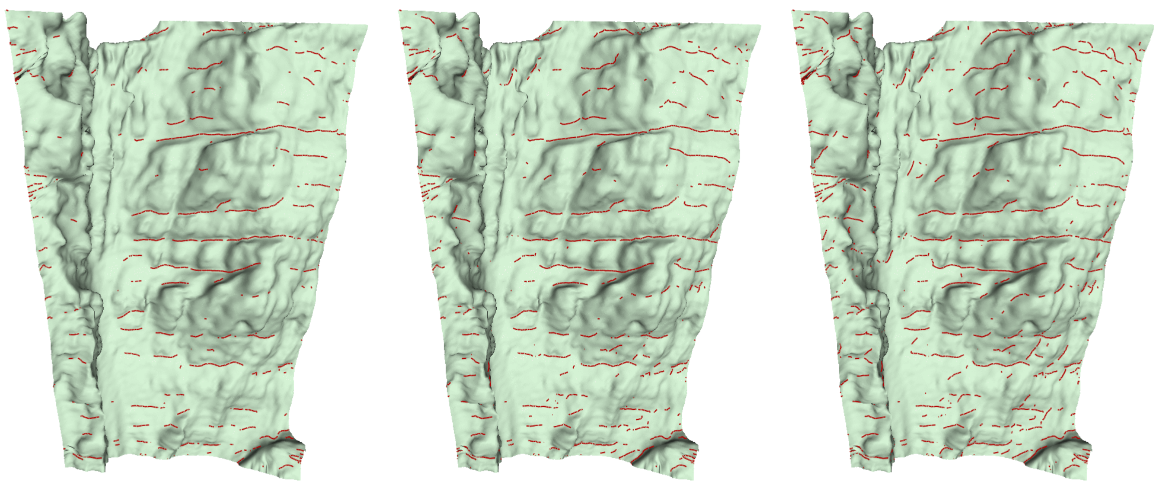

Figure 8: Influence of the tolerance threshold T with values of 0.85 (left image), 0.70 (middle image) and 0.55 (right image).