Fast and Robust Detection of Crest Lines on Meshes

Shin Yoshizawa Alexander Belyaev Hans-Peter Seidel

Computer Graphics Group, MPI Informatik, Saarbrücken, Germany
{shin.yoshizawa | belyaev | hpseidel}@mpi-sb.mpg.de

Abstract

We propose a fast and robust method for detecting crest lines on surfaces approximated by dense triangle meshes. The crest lines, salient surface features defined via first- and second-order curvature derivatives, are widely used for shape matching and interrogation purposes. Their practical extraction is difficult because it requires good estimation of high-order surface derivatives. Our approach to the crest line detection is based on estimating the curvature tensor and curvature derivatives via local polynomial fitting.

Since the crest lines are not defined in the surface regions where the surface focal set (caustic) degenerates, we introduce a new thresholding scheme which exploits interesting relationships between curvature extrema, the so-called MVS functional of Moreton and Sequin, and Dupin cyclides,

An application of the crest lines to adaptive mesh simplification is also considered.

Introduction

Developing methods for fast and accurate detection of feature lines on polygonal and point-sampled surfaces is currently a subject of intensive research [Khaneja et al. 1998; Gumhold et al. 2001; Hubeli and Gross 2001; Lee and Lee 2002; Pauly et al. 2003; Page et al. 2002; Ohtake et al. 2004; Stylianou and Farin 2004; Cazals and Pouget 2004a]. In this paper, we propose a fast and robust method for detecting surface creases on surfaces approximated by dense triangle meshes.

Surface creases, curves on a surface along which the surface bends sharply can be intuitively defined as loci of sharp variation points of the surface normal. Mathematically the sharp variation points of the surface normals are described via extrema of the surface principal curvatures along their corresponding lines of curvature. These curvature extrema, called also ridges, have been thoroughly studied in connection with research on classical differential geometry and singularity theory [Koenderink 1990; Porteous 1994; Belyaev et al. 1997; Hallinan et al. 1999; Cazals and Pouget 2004b]. The ridges and their subsets have numerous applications in image and data analysis [Monga et al. 1992] quality control of free-form surfaces [Hosaka 1992], human perception [Hoffman and Richards 1985], analysis and registration of anatomical structures [Pennec et al. 2000], geomorphology [Little and Shi 2001] and non-photorealistic rendering [Interrante et al. 1995; DeCarlo et al. 2003]. See also references therein. The so-called crest lines are formed by the perceptually salient ridge points and consist of the surface points where the magnitude of the largest (in absolute value) principal curvature attains a maximum along its corresponding line of curvature.

Practical detection of the crest lines and other types of curvature extrema is a difficult computational task because it requires a high-quality estimation of the curvature tensor and curvature derivatives. In general, global fitting methods do a better job in estimating high-order surface derivatives and lead to more accurate detection of curvature extrema [Kent et al. 1996; Ohtake et al. 2004] than the local estimation schemes. On the other hand, the local schemes are much faster and often demonstrate a quite satisfactory performance [Guéziec 1993; Stylianou and Farin 2004; Cazals and Pouget 2004a].

Our procedure for detecting the crest lines combines local polynomial fitting based on a modification of the method of [Goldfeather and Interrante 2004], a finite difference scheme/test proposed in [Ohtake et al. 2004] and used for curvature maxima/minima identification, and a careful thresholding based on the MVS functional of Moreton and Sequin [Moreton and Sequin 1992]. Our method is fast since we estimate necessary surface derivatives via local polynomial fitting. For example, for the Igea model consisting more than $200K$ triangles it takes only nine seconds for estimating the curvature tensor and curvature derivatives and four seconds for detecting crest lines on a standard 1.7 GHz Pentium 4 PC. Our approach is capable of achieving high quality results comparable with those obtained via global fitting procedures [Ohtake et al. 2004]. Fig. 1 shows crest line patterns found on simple and complex geometrical models for various values of a user-specified parameter which controls the strength of detected crest lines.

We also consider applications of the crest lines to adaptive mesh simplification.

Differential Geometry Background

Consider a smooth oriented surface $\mathscr{S}$ and denote $k_{\max}$ and $k_{\min}$ its maximal and minimal principal curvatures, $k_{\max} \geq k_{\min}$. Let $\mathbf{t}_{\max}$ and $\mathbf{t}_{\min}$ be the corresponding principal directions. Denote by $e_{\max}$ and $e_{\min}$ the derivatives of the principal curvatures along their corresponding curvatures directions:

$$e_{\max} = \partial k_{\max}/\partial \mathbf{t}_{\max}, \qquad e_{\min} = \partial k_{\min}/\partial \mathbf{t}_{\min}.$$

Following [Thirion 1996] let us call $e_{\max}$ and $e_{\min}$ the extremality coefficients. The extremality coefficients are not defined at the umbilical points ($k_{\max} = k_{\min}$) since the principal directions are undefined there. The ridges are formed by the closure of points on $\mathscr{S}$ where one of the extremality coefficients vanishes. According to this definition, the umbilical points are ridge points. In [Porteous 1994; Hallinan et al. 1999] the ridge patterns in small vicinities of umbilical points are analyzed.

The crest lines consist of perceptually salient ridge points. We distinguish convex and concave crest lines. The convex crest lines are given by

$$e_{\max} = 0, \quad \partial e_{\max}/\partial \mathbf{t}_{\max} < 0, \quad k_{\max} > |k_{\min}|,$$

while the concave crest lines are characterized by

$$e_{\min} = 0, \quad \partial e_{\min}/\partial \mathbf{t}_{\min} > 0, \quad k_{\min} < -|k_{\max}|.$$

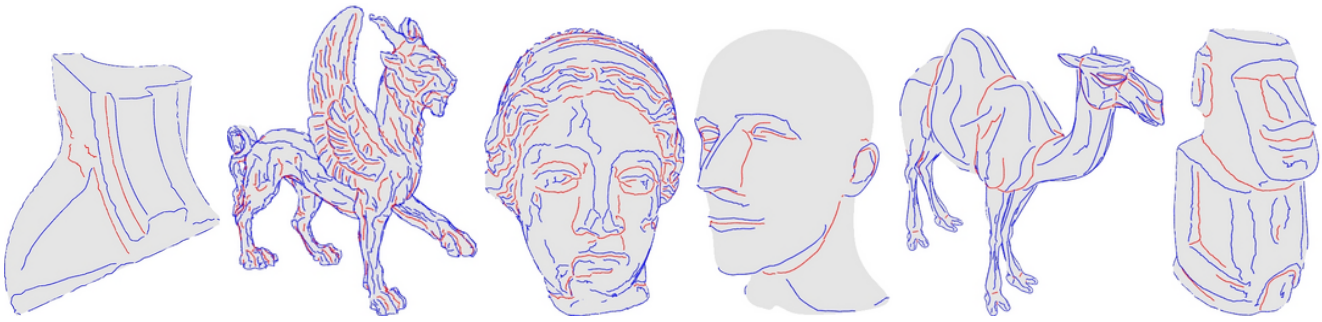


Figure 1: Crest lines detected on various triangle meshes. A scale-independent parameter T defined by (5) is used to keep the most visually important features: $T = 2.4$ for the Fan model, $T = 1.0$ for the Feline model, $T = 2.7$ for the Igea model, $T = 3.2$ for the Mannequin Head model, $T = 0.9$ for the Camel model, and $T = 2.3$ for the Moai model. For all the model one-ring neighborhood polynomial fitting is used for estimating the curvature tensor and curvature derivatives.

The convex and concave crest lines are dual w.r.t. the surface orientation: changing the orientation turns the convex crest lines into concave one and vice versa.

Denote by $\mathscr{F}$ the focal set of $\mathscr{S}$. The focal set is formed by the principal centers of curvature and consists of two sheets corresponding to the maximal and minimal principal curvatures $k_{\max}$ and $k_{\min}$. Focal set $\mathscr{F}$ is not a smooth surface and has singularities. The singularities of $\mathscr{F}$ consist of space curves which correspond to the ridges on $\mathscr{S}$. There is a special family of surfaces, the so-called Dupin cyclides, whose focal sets degenerate into space curves (a sphere and a plane can be considered as degenerate Dupin cyclides whose focal sets are isolated points; the focal point of a plane is located at infinity). The ridges are not defined on a Dupin cyclide. The Dupin cyclides were introduced by the French geometer Charles Dupin at the beginning of the 19th century and since then have been intensively studied in connection with various shape modeling tasks. See, for example, [Chandru et al. 1989] for a short historical survey of the Dupin cyclides and their usage and [Foufou and Garnier 2004] for recent applications of the Dupin cyclides in geometric modeling as a CAGD primitive. The skeleton (medial axis) of a figure bounded by a Dupin cyclide degenerates into a curved axis. The family of Dupin cyclides includes spheres, cylinders, cones, and tori.

One can show that the Dupin cyclides are characterized by the condition

$$|e_{\max}|^2 + |e_{\min}|^2 = 0. \tag{1}$$

Notice that the left-hand side of (1) is the integrand of the so-called MVS functional introduced in [Moreton and Sequin 1992] for fair surface design purposes.

Assume that $\mathscr{S}$ is given in the parametric form $\mathbf{r} = \mathbf{r}(u,v)$ and let $\mathbf{n}(u,v)$ be the unit normal vector to $\mathscr{S}$. Consider a point $\mathbf{r}_0 = \mathbf{r}(u_0,v_0)$ and assume that $(\mathbf{r}_u, \mathbf{r}_v, \mathbf{n})$ form an orthonormal basis at $\mathbf{r}_0$. Further, let us assume that the basis vectors $\mathbf{r}_u$ and $\mathbf{r}_v$ coincide with the principal directions $\mathbf{t}_{\max}$ and $\mathbf{t}_{\min}$, respectively, at $\mathbf{r}_0$. Then, according to the classical formula of Rodrigues, we have

$$\mathbf{n}_u = -k_{\max}\mathbf{r}_u \quad \text{and} \quad \mathbf{n}_v = -k_{\min}\mathbf{r}_v,$$

The focal set $\mathscr{F}$ is given by

$$\mathbf{f} = \mathbf{r}(u,v) + R(u,v)\mathbf{n}(u,v) \quad \text{with} \quad R = 1/k_{\max}, 1/k_{\min}.$$

The focal set $\mathbf{f}(u,v)$ degenerates at (u_0,v_0) if the oriented area element of $\mathscr{F}$ corresponding to the oriented area element of $\mathscr{S}$ vanishes at $\mathbf{r}_0$. It gives

$$\mathbf{f}_u \times \mathbf{f}_v = (\mathbf{r}_u + R_u\mathbf{n} + R\mathbf{n}_u) \times (\mathbf{r}_v + R_v\mathbf{n} + R\mathbf{n}_v) = 0,$$

where $\mathbf{f}_u$ and $\mathbf{f}_v$ are the partial derivatives of $\mathbf{f}(u,v)$. After obvious simplifications we arrive at

$$\mathbf{r}_u R_u (1 - Rk_{\min}) + \mathbf{r}_v R_v (1 - Rk_{\max}) + \\ + \mathbf{n}(1 - Rk_{\max})(1 - Rk_{\min}) = 0,$$

where $R = 1/k_{\max}, 1/k_{\min}$. Thus the $k_{\max}$-branch of $\mathscr{F}$ degenerates at $\mathbf{f}(u_0,v_0)$ if $e_{\max} = 0$, the $k_{\min}$-branch of $\mathscr{F}$ degenerates at $\mathbf{f}(u_0,v_0)$ if $e_{\min} = 0$, and both the branches degenerate at common point $\mathbf{f}(u_0,v_0)$ if $k_{\max} = k_{\min}$. The focal points corresponding to surface points where either $e_{\max} = 0$ or $e_{\min} = 0$ (ridges) form the so-called cuspidal edges called also focal ribs [Porteous 1987; Porteous 1994]. For a generic surface, focal ribs always go through the focal set singularities corresponding to the umbilics of the surface. Now we can conclude that a generic surface region where the left hand-side of (1) is small is close to a part of a Dupin cyclide.

Practical detections of the ridges and their subsets is extremely difficult in those surface regions which are slightly perturbed Dupin cyclide patches and where, therefore, the left hand-side of (1) is close to zero. Such regions may contain many spurious ridges (and crest lines). Thus it seems natural to use the left hand-side of (1) as a measure for selecting geometrically important crest lines.

Estimating Surface Derivatives

Given a mesh $\mathscr{M}$ approximating a smooth surface $\mathscr{S}$, in order to achieve a fast and accurate estimation of the principal curvatures and their derivatives a bivariate polynomial is fitted locally to each mesh vertex. To date, two polynomial fitting strategies are used for estimating surface derivatives at a mesh vertex. According to one strategy, it is assumed that the surface normal at vertex is preliminary estimated. It leads to the so-called adjacent-normal cubic approximation method [Goldfeather and Interrante 2004]. The second strategy [Cazals and Pouget 2003] does not assume that the mesh normal is already given. According to our numerical experience, if the vertex normal is approximated appropriately, the first strategy leads to a better estimation of the surface curvatures and their derivatives at the vertex.

In our numerical experiments we use the following enhancement of adjacent-normal cubic approximation method. For each mesh vertex $\mathbf{p} \in \mathscr{M}$ its one-link neighborhood is considered and a new vertex $\mathbf{p}'$ is obtained as the arithmetic mean of the centroids of the mesh triangles adjacent to $\mathbf{p}$. These new vertices $\{\mathbf{p}'\}$ form a new mesh $\mathscr{M}'$ which is smoother than $\mathscr{M}$. Now for each vertex $\mathbf{p}' \in \mathscr{M}'$ its unit normal is estimated via Nelson Max's method [Max 1999]. Then a cubic polynomial

$$h(x,y) = \frac{1}{2}\left(b_0x^2 + 2b_1xy + b_2y^2\right) + \\ + \frac{1}{6}\left(c_0x^3 + 3c_1x^2y + 3c_2xy^2 + c_3y^3\right) \tag{2}$$

is fitted in the least-square sense [Goldfeather and Interrante 2004] to $\mathbf{p}'$ and a set of its neighboring vertices. That set of neighbors of $\mathbf{p}'$ is obtained from the k-link neighborhood of $\mathbf{p}'$ by removing those vertices whose normals make obtuse angles with the normal at $\mathbf{p}'$. In practice we use $k = 1,2,3,4$. Next the curvature tensor and extremality coefficients are expressed via derivatives of local cubic polynomial $h(x,y)$. Finally these curvature attributes are assigned to the original vertices $\{\mathbf{p}\}$ of mesh $\mathscr{M}$.

We have also derived an elegant formula for an extremality coefficient at a surface point where $\mathscr{S}$ is locally approximated by (2)

$$e = \partial k/\partial \mathbf{t} = \begin{pmatrix} t_1^2 \\ t_2^2 \end{pmatrix}^T \begin{pmatrix} c_0 & c_1 \\ c_2 & c_3 \end{pmatrix} \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}. \qquad (3)$$

Here $\mathbf{t} = (t_1,t_2)^T$ is the principal direction corresponding to a principal curvature k. Because of its simplicity, (3) leads to a significant reduction of computational time.

To prove (3) let us consider the well-known formula for an extremality coefficient $e = \partial k/\partial \mathbf{t}$ for a surface given in implicit form $F(\mathbf{x}) = 0$, $\mathbf{x} = (x_1,x_2,x_3)$, (see, for example, [Porteous 1994, Exercise 11.8] and also [Monga et al. 1992] where a small mistake in the final formulas for the curvature derivatives is made)

$$e = \nabla k \cdot \mathbf{t} = \frac{F_{ijl}t_it_jt_l + 3kF_{ij}t_in_j}{|\nabla F|}, \qquad (4)$$

where F_{ij} and F_{ijl} denote the second and third partial derivatives of $F(\mathbf{x})$, respectively, $\mathbf{t} = (t_1,t_2,t_3)$ is the principal direction corresponding to a principal curvature k, $\mathbf{n} = (n_1,n_2,n_3)$ is the unit surface normal, and the summation over repeated indices is implied. In our case, $F = z - h(x,y)$ and at the origin of coordinates $\mathbf{n} = (0,0,1)$ and $\mathbf{t} = (t_1,t_2,0)$. Thus, since the polynomial $h(x,y)$ does not contain linear terms, at the origin of coordinates (4) simplifies into

$$e = F_{ijl}t_it_jt_l$$

and (3) immediately follows.

Fig. 2 compares the sets of crest lines detected on a 3D text mesh via the straightforward polynomial fitting (the top image) and the enhanced adjacent-normal cubic approximation method (we use $k = 1$ in this example).

Although our scheme for estimating surface derivatives seems complicated, it leads to highly effective crest line detection procedure which only slightly depends on the mesh connectivity and triangle aspect ratios. In Fig. 3 we compare the patterns of the crest lines detected on the original Stanford bunny mesh and on the mesh obtained via an implicitization of the bunny model and then polygonizing using Bloomental's method [Bloomenthal 1994]. Despite the fact that the new bunny mesh contains many sliver triangles and has irregular connectivity, the patterns of the crest lines found on the meshes are remarkably similar.

Tracing and Thresholding Crest Lines

Once the curvature tensor and extremality coefficients are estimated at each vertex of $\mathscr{M}$, we inspect the edges $\mathscr{M}$ and check whether they contain curvature maxima and minima. We detect the crest line vertices and connect them together following the procedure proposed in [Ohtake et al. 2004] with one small, but important, addition. It turns out that the procedure may generate several close disconnected crest lines in situations similar to those shown in the left image of Fig. 4. In order to reduce the fragmentation of the crest lines we inspect the mesh vertices and their one-ring neighborhoods. For each one-ring vertex neighborhood containing crest line end-points we connect two end-points if $\alpha \le \pi/3$, $\beta \le \pi/3$, $\gamma \le \pi/2$, where α, β, and γ are the angles between the end-segments and the segment connecting the end-points, as seen the right image of Fig. 4.

Figure 2: Crest lines detected on 3D text. Top: polynomial fit without preliminary estimation of mesh normals is used. Bottom: the enhanced adjacent-normal cubic approximation method is employed for estimating surface curvatures and their derivatives. In both the cases preliminary smoothing $\mathbf{p} \to \mathbf{p}'$ was applied.

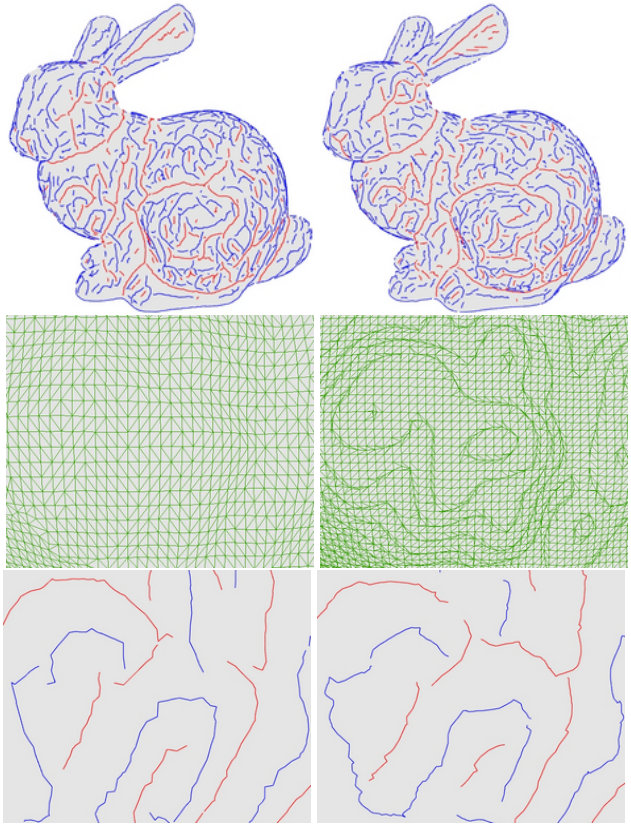


Figure 3: Patterns of crest lines and mesh triangles for two bunny models. Left: original Stanford bunny mesh with 69,451 triangles is used. Right: another bunny mesh with 279,984 triangles is used. The necessary surface derivatives are estimated via the enhanced cubic polynomial fitting with $k = 1$ for the original Stanford bunny mesh and $k = 3$ for the remeshed bunny since the latter is more than three times bigger than the original one.

As we mentioned before, the sum of squared extremality coefficients is very appropriate for measuring saliency of the crest lines.

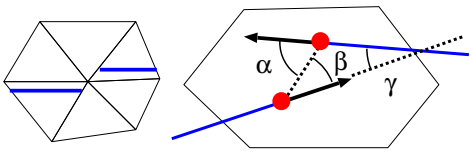


Figure 4: Left: a situation when we may want to connect the crest lines (shown in bold) together. Right: angles α, β, and γ generated by crest line end-segments and the segment connecting crest line end-points are used to measure when gap-jumping is necessary.

In practice we use the following scale-independent quantity to measure the strength of a crest line

$$T = \int ds \cdot \int \sqrt{|e_{\max}|^2 + |e_{\min}|^2}\, ds, \tag{5}$$

where the integrals are taken over the crest line. This thresholding parameter involves third-order surface derivatives and is more complex than that used in [Ohtake et al. 2004] where the integral of a principal curvature along a feature line was used. On the other hand, thresholding with (5) is simpler than the thresholding scheme proposed in [Cazals and Pouget 2004a] where a second-order curvature derivative is used for filtering out spurious ridges and crest lines.

We use a linear interpolation scheme for estimating the cyclideness

$$C = \sqrt{|e_{\max}|^2 + |e_{\min}|^2} \tag{6}$$

at crest line vertex $\mathbf{v}$ located on mesh edge $[\mathbf{p},\mathbf{q}]$:

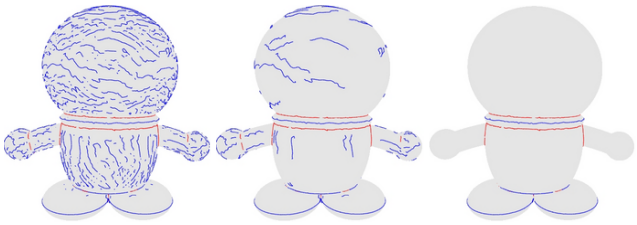
$$C(\mathbf{p}) = \frac{aC(\mathbf{p}) + bC(\mathbf{q})}{a+b},$$

where $a = |e_{\max}(\mathbf{q})|$, $b = |e_{\max}(\mathbf{p})|$ for the convex crest lines and $a = |e_{\min}(\mathbf{q})|$, $b = |e_{\min}(\mathbf{p})|$ for the concave ones. Now the integrals in (5) are estimated by a simple trapezoid approximation similar to that used in [Ohtake et al. 2004].

Roughly speaking, cyclideness (6) measures how far a surface region is from being a part of a Dupin cyclide. If $\mathbf{x}$ lies on a convex (concave) crest line, then $e_{\max}(\mathbf{x}) = 0$ and $C(\mathbf{x}) = |e_{\min}(\mathbf{x})|$ ($e_{\min}(\mathbf{x}) = 0$ and $C(\mathbf{x}) = |e_{\max}(\mathbf{x})|$).

At the first glance, it looks that (5) does not affect umbilical regions. In fact it does: by continuity cyclideness (6) vanishes at the isolated umbilics. A small perturbation of an umbilic region creates a non-umbilical region containing isolated umbilics. Further, as it was shown in [Belyaev et al. 1997], the crest lines do not pass through the generic (typical) umbilics.

Fig. 5 demonstrates how our crest line filtering scheme works for a model with spherical and cylindrical regions. Notice how well the crest lines detected at the mesh parts approximated those regions are filtered out.



$T = 0$ $T = 0.2$ $T = 6.7$

Figure 5: Detecting crest lines for a model containing spherical and cylindrical regions for various values of threshold T. For each mesh vertex, its three-ring neighborhood ($k = 3$) is used for local polynomial fitting.

Fig. 6 exposes detecting crest lines on a more complex model containing flat, cylindrical, and slightly curved regions and small features. Increasing T allows us to remove inessential crest lines while preserving salient ones. The figure also demonstrates how the size of vertex neighborhoods used for polynomial fitting affects the crest line detection procedure. A larger neighborhood leads to smoother approximation of the mesh and, therefore, allows us to disregard the crest lines located in slightly convex/concave regions. See also Fig. 1 where one-ring neighborhood polynomial fitting is used for all the models.

Crest Lines and Mesh Simplification

In this section, we develop a quadric-based mesh simplification procedure guided by the distance field from crest lines. Our use of crest lines for adaptive mesh simplification purposes is inspired by recent work [Kho and Garland 2003]. Since crest lines on a mesh are important shape features, it is natural to simplify the mesh aggressively far from the most salient crest lines and preserve the mesh in a vicinity of them.

Given a set of feature lines (crest lines, in our case) on surface $\mathscr{S}$, following [Lévy et al. 2002] for a surface point $\mathbf{p} \in \mathscr{S}$ we consider $d(\mathbf{p})$ the geodesic distance between $\mathbf{p}$ and the closest feature line (crest line) point. Let $\max(d)$ be the maximum of the geodesic distances $d(\mathbf{p})$ over all points of $\mathscr{S}$. We introduce a scale-independent weighted distance function

$$F(d) = \left(\frac{d}{\max(d)} + \varepsilon\right)^{\eta}, \tag{7}$$

where ε is a regularization parameter (in all our experiments we use $\varepsilon = 0.1$) and η is a positive user-specified parameter which is used to control a degree of influence of the crest lines.

Once the crest lines are detected and filtered, we compute a discrete feature distance d_i for each triangle $T_i \in \mathscr{M}$. Let us define the distance between two triangles T_j of T_i of $\mathscr{M}$ sharing a common edge as the sum of distances between the triangle centroids and the edge midpoint. To compute $\{d_i\}$ we use a variant the Floyd-Warshall all-pairs shortest path algorithm.

Fig. 7 visualizes the distance fields computed on the Max-Planck and Stanford bunny meshes.

Similar to [Kho and Garland 2003] a weighted quadric error metric $w_j Q(T_j)$ is assigned to each triangle T_j of mesh $\mathscr{M}$, where $Q(T_j)$ is the standard Garland-Heckbert QEM [Garland and Heckbert 1997]. We set $w_j = 1/F(d_j)$ and control the degree of influence of crest lines via parameter η in (7). Fig. 8 presents the Max-Planck mesh its eye region 90%-decimated for various values of η. The detected crest line are those shown in the left image of Fig. 7. The mesh density is changing smoothly according to geodesic distance to the crest lines.

Discussion

We have presented a fast method for detecting salient curvature extrema on surfaces approximated by dense triangle meshes. It processes about $20K/k$ triangles per second on a standard 1.7 GHz Pentium 4 PC with 512 MB RAM for k-ring neighborhood polynomial fitting and estimating the curvature tensor and curvature derivatives. The crest line tracing stage at the method is faster than the estimation stage although the former depends on geometric complexity of models. The source code of our method is available on the Web for evaluation [Yoshizawa].

The method is robust. The results of our crest line detection procedure depends only slightly on the quality of the mesh, as demonstrated in Fig. 3.

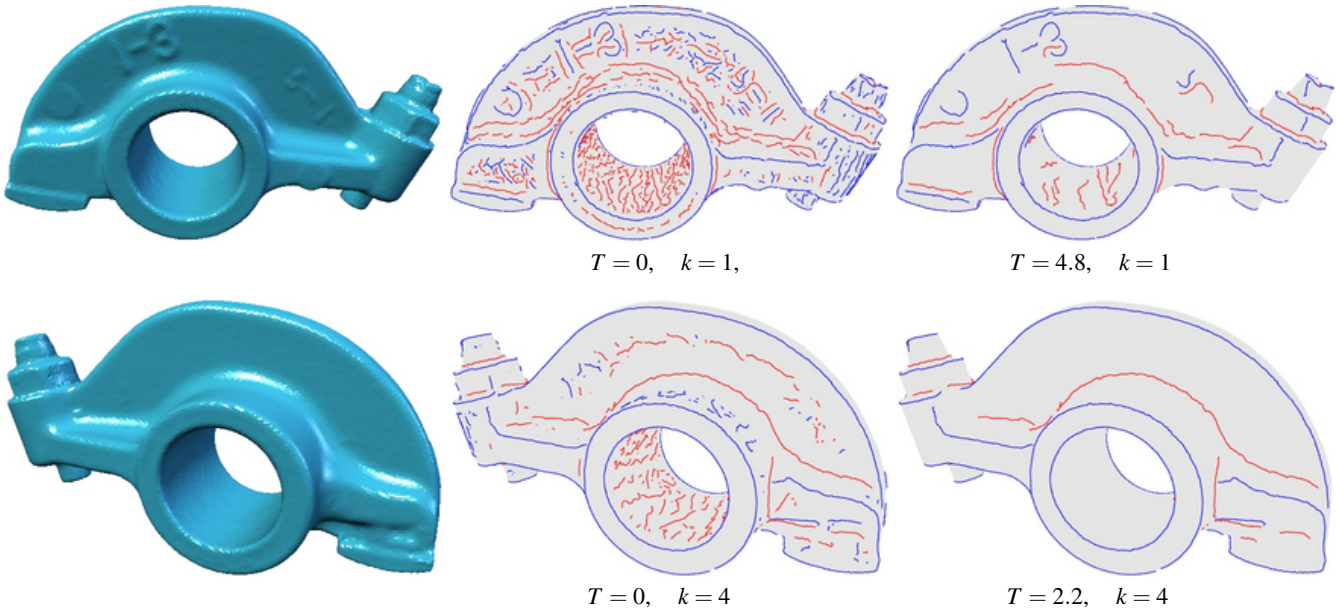


Figure 6: Crest lines detected on a mechanical part model with different values of threshold T and vertex neighborhood size k used for local polynomial fitting.

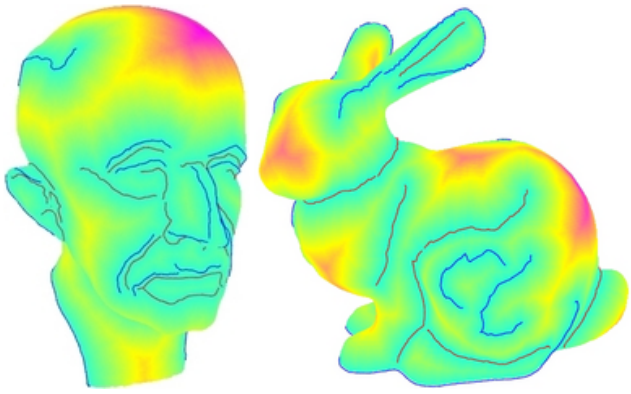


Figure 7: Distance from salient crest lines is visualized for Max-Planck and Stanford bunny meshes. The crest lines are found with three-ring neighborhood fitting ($k = 3$) and $T = 40$ for both the models.

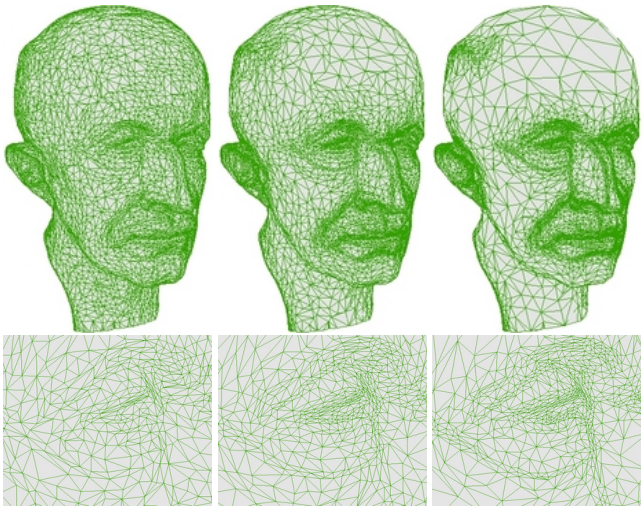


Figure 8: Max-Planck mesh and its eye part 90%-decimated for various values of η. The left image ($\eta = 0$) shows the result of the standard Garland-Heckbert decimation procedure.

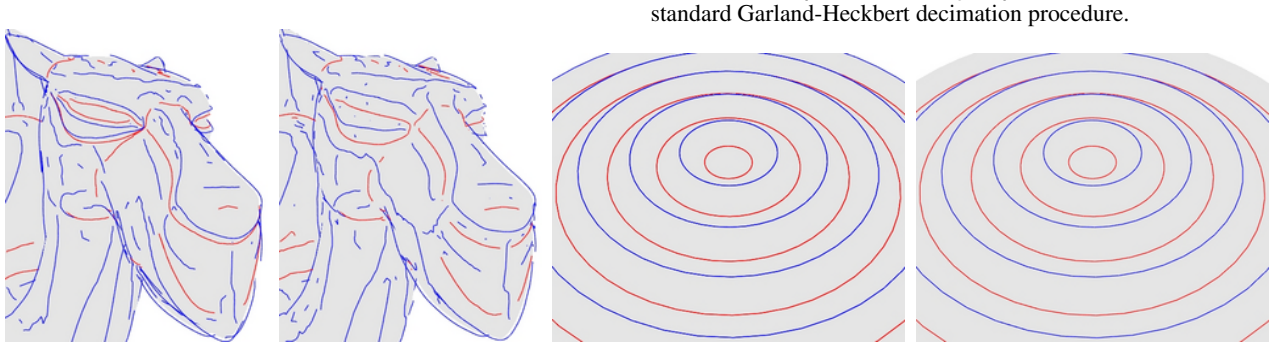


Figure 9: (a) Crest lines detected using the method of this paper, one-ring neighborhood fitting is used. (b) Crest lines detected using a global implicit fitting method [28]. In both the cases, no filtering is applied. (c) The exact crest lines on a simple analytical surface. (d) The crest lines detected with our method, one-ring neighborhood fitting is employed.

Our method is capable of achieving high quality results in detecting salient curvature extrema to compare with schemes based on global fitting procedures. In Fig. 9 we give a visual comparison of our method with that developed in [Ohtake et al. 2004] (the two left images) and with the exact detection of the crest lines on analytical waving surface $\mathbf{r}(u,v) = [u\cos v, u\sin v, \cos u]$ (two right images).[1]

Our filtering scheme for removing unessential crest lines is based on interesting relationships between Dupin cyclides, focal sets, curvature extrema, and variational functionals. We use cyclideness (6) as the main ingredient of our filtering scheme and measure the strength of crest lines by scale-independent quantity (5). Thus long but weak crest lines are preferred to strong but short ones. Of course, different filtering procedures can be also used instead of that based on (5). Similar manual thresholding schemes were also used in [Ohtake et al. 2004; Cazals and Pouget 2004a]. Manual filtering is hardly avoidable for complex geometry surfaces, since the crest lines are local surface features while saliency-based thresholding should take into account global surface shape.

Finally we have demonstrated how crest lines can be used for adaptive mesh simplification preserving visually important shape features.

Acknowledgements

Thanks to Yutaka Ohtake for fruitful discussions and the Moai model and to Alexander Pasko for the Robot Cat model. We are grateful to Stanford University, University of Washington, MPI Informatik, and Cyberware for providing other mesh models used in this paper. This work was supported in part by AIM@SHAPE, a Network of Excellence project (506766) within EU's Sixth Framework Programme.

References

BELYAEV, A. G., ANOSHKINA, E. V., AND KUNII, T. L. 1997. Ridges, ravines, and singularities. In A. T. Fomenko, and T. L. Kunii, *Topological Modeling for Visualization*, Springer, 375–383. Ch. 18.

BLOOMENTHAL, J. 1994. An implicit surface polygonizer. In *Graphics Gems IV*. 324–349.

CAZALS, F., AND POUGET, M. 2003. Estimating differential quantities using polynomial fitting of osculating jets. In *Symposium on Geometry Processing*, 177–187.

CAZALS, F., AND POUGET, M. 2004. Ridges and umbilics of a sampled smooth surface: a complete picture gearing toward topological coherence. Rapport de Recherche RR-5294, INRIA, September.

CAZALS, F., AND POUGET, M. 2004. Smooth surfaces, umbilics, lines of curvatures, foliations, ridges and the medial axis: a concise overview. Rapport de Recherche RR-5138, INRIA, March.

CHANDRU, V., DUTTA, D., AND HOFFMANN, C. M. 1989. On the geometry of Dupin cyclides. *The Visual Computer 5*, 5 (October), 277–290.

DECARLO, D., FINKELSTEIN, A., RUSINKIEWICZ, S., AND SANTELLA, A. 2003. Suggestive contours for conveying shape. *ACM Transactions on Graphics 22*, 3, 848–855. Proc. ACM SIGGRAPH 2003.

FOUFOU, S., AND GARNIER, L. 2004. Dupin cyclide blends between quadric surfaces for shape modeling. *Computer Graphics Forum 23*, 3, 321–330. Eurographics 2004 issue.

GARLAND, M., AND HECKBERT, P. 1997. Surface simplification using quadric error metrics. In *Proceedings of ACM SIGGRAPH*, 209–216.

GOLDFEATHER, J., AND INTERRANTE, V. 2004. A novel cubic-order algorithm for approximating principal direction vectors. *ACM Transactions on Graphics 23*, 1, 45–63.

[1] The mesh we used to approximate the waving surface is not dense: it consists of less than 5K triangles only. Nevertheless the max-norm error estimates for the extemality coefficients are reasonably good: 0.48 for $e_{\max}$ and 0.56 for $e_{\min}$.

GUÉZIEC, A. 1993. Large deformable splines, crest lines and matching. In *Proc. IEEE Fourth Int'l Conf. Computer Vision*, 650–657.

GUMHOLD, S., WANG, X., AND MCLEOD, R. 2001. Feature extraction from point clouds. In *Proc. 10th International Meshing Roundtable*, 293–305.

HALLINAN, P. L., GORDON, G. G., YUILLE, A. L., GIBLIN, P., AND MUMFORD, D. 1999. *Two- and Tree-Dimensional Patterns of the Face*. A K Peters.

HOFFMAN, D. D., AND RICHARDS, W. A. 1985. Parts of recognition. *Cognition 18*, 65–96.

HOSAKA, M. 1992. *Modeling of Curves and Surfaces in CAD/CAM*. Springer, Berlin.

HUBELI, A., AND GROSS, M. 2001. Multiresolution feature extraction from unstructured meshes. In *Proc. IEEE Visualization 2001*, 287–294.

INTERRANTE, V., FUCHS, H., AND PIZER, S. 1995. Enhancing transparent skin surfaces with ridge and valley lines. In *Proc. IEEE Visualization 1995*, 52–59.

KENT, J. T., MARDIA, K. V., AND WEST, J. 1996. Ridge curves and shape analysis. In *The British Machine Vision Conference 1996*, 43–52.

KHANEJA, N., MILLER, M. I., AND GRENANDER, U. 1998. Dynamic programming generation of curves on brain surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence 20*, 11, 1260–1265.

KHO, Y., AND GARLAND, M. 2003. User-guided simplification. In *ACM Symposium on Interactive 3D Graphics*, 123–126.

KOENDERINK, J. J. 1990. *Solid Shape*. MIT Press.

LEE, Y., AND LEE, S. 2002. Geometric snakes for triangular meshes. *Computer Graphics Forum 21*, 3, 229–238. Eurographics 2002 issue.

LÉVY, B., PETITJEAN, S., RAY, N., AND MAILLOT, J. 2002. Least squares conformal maps for automatic texture atlas generation. *ACM Transactions on Graphics 21*, 3 (July), 362–371. Proc. ACM SIGGRAPH 2002.

LITTLE, J. J., AND SHI, P. 2001. Structural lines, TINs and DEMs. *Algorithmica 30*, 2, 243–263.

MAX, N. 1999. Weights for computing vertex normals from facet normals. *Journal of Graphics Tools 4*, 2, 1–6.

MONGA, O., BENAYOUN, S., AND FAUGERAS, O. 1992. From partial derivatives of 3-D density images to ridge lines. In *Proc. CVPR'92*, IEEE, 354–359.

MORETON, H. P., AND SEQUIN, C. H. 1992. Functional optimization for fair surface design. In *SIGGRAPH'92 Proceedings*, 167–176.

OHTAKE, Y., BELYAEV, A., AND SEIDEL, H.-P. 2004. Ridge-valley lines on meshes via implicit surface fitting. *ACM Transactions on Graphics 23*, 3 (August), 609–612. Proc. ACM SIGGRAPH 2004.

PAGE, D. L., SUN, Y., KOSCHAN, A., PAIK, J., AND ABIDI, M. 2002. Normal vector voting: Crease detection and curvature estimation on large, noisy meshes. *Journal of Graphical Models 64*, 1–31.

PAULY, M., KEISER, R., AND GROSS, M. 2003. Multi-scale feature extraction on point-sampled models. *Computer Graphics Forum 22*, 3, 281–289. Eurographics 2003 issue.

PENNEC, X., AYACHE, N., AND THIRION, J. P. 2000. Landmark-based registration using features identified through differential geometry. In *Handbook of Medical Imaging*, I. N. Bankman, Ed. Academic Press, 499–513.

PORTEOUS, I. R. 1987. Ridges and umbilics of surfaces. In *The Mathematics of Surfaces II*, Clarendon Press, Oxford, R. R. Martin, Ed., 447–458.

PORTEOUS, I. R. 1994. *Geometric Differentiation for the Intelligence of Curves and Surfaces*. Cambridge University Press, Cambridge.

STYLIANOU, G., AND FARIN, G. 2004. Crest lines for surface segmentation and flattening. *IEEE Transactions on Visualization and Computer Graphics 10*, 5 (September/October), 536–544.

THIRION, J. 1996. The extremal mesh and the understanding of 3D surfaces. *International Journal of Computer Vision 19*, 2, 115–128.

YOSHIZAWA, S. www.mpi-sb.mpg.de/∼shin/Research/Crests/Crests.html.