Vis Comput (2010) 26: 433–443
DOI 10.1007/s00371-010-0458-6

ORIGINAL ARTICLE

Perceptual-saliency extremum lines for 3D shape illustration

Yongwei Miao · Jieqing Feng

Published online: 9 April 2010

Abstract Owing to their efficiency for conveying perceptual information of the underlying shape and their pleasing perceiving in visual aesthetics experience, line drawings are now becoming a widely used technique for illustrating 3D shapes. Using a center-surrounding bilateral filter operator on Gaussian-weighted average of local projection height between mesh vertices and their neighbors, a new perceptual-saliency measure which can depict surface salient features, is proposed in this paper. Due to the definition of perceptual-saliency measure, our perceptual-saliency extremum lines can be considered as the ridge-valley lines of perceptual-saliency measure along the principal curvature directions on triangular meshes. The experimental results demonstrate that these extremum lines effectively capture and depict 3D shape information visually, especially for archaeological artifacts.

Keywords Non-photorealistic rendering · Shape illustration · Perceptual-saliency measure · Extremum lines · Triangular mesh

1 Introduction

In the literature of non-photorealistic rendering and smart graphics, line drawings are now becoming prevalent as a shape illustration technique. The prevalence of this technique depends on its internal inevitability. On the one hand, comparing with other shape illustration techniques, line drawings can convey meaningful information of 3D shape in a relatively succinct manner by ignoring less important details [8, 26]. It has been proven to be an effective and commonly-used way to achieve the abstractive nature of various artistic styles. On the other hand, comparing with the commonly-used photorealistic synthesis techniques, line drawings are also proved to be both more effective in communication and more pleasing in visual experience [3, 26]. Thus, the research of automatic algorithms for line drawings is flourishing in the recent computer graphics literature.

Line drawing styles can be found in many applications, including non-photorealistic rendering [28], technical illustration [8], shape simplification [24], shape segmentation [29], and the recovery of archaeological design [20]. In general, these drawings can depict shape information more efficiently and precisely than other ways, with abundant details like photographs, because of their portable representation size, efficient synthesis and resolution-independent manner.

Cole et al. [3] indicate that people can interpret certain shapes almost as well from a line drawing as from a shaded image, and that line drawing techniques in current computer graphics can effectively depict shape and even match the effectiveness of artists' drawings. However, as recent researches point out, none of the current feature techniques can capture all relevant features, such as perceptual-saliency edges, weak edges, highly curved edges, etc. [2, 12].

To convey 3D shapes efficiently, the extraction of perceptually salient information is important [18, 27, 32]. To define perceptually salient information quantitatively which is consilient with geometric features on the surface, a new type of view-independent feature lines are presented in this paper, which are called perceptual-saliency extremum lines.

Y. Miao · J. Feng
State Key Lab. of CAD&CG, Zhejiang University, Hangzhou, P.R. China

Y. Miao (✉)
College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, P.R. China
e-mail: ywmiao@zjut.edu.cn

They are the loci of extremum points of function that describes the perceptual-saliency along the principal curvature directions on the underlying 3D shape. The experimental results demonstrate that these extremum lines can effectively and comprehensively capture the visual information for 3D shapes, especially for illustrating archaeological artifacts.

Our main contributions can be summarized as follows:

- A new perceptual-saliency measure is defined, whose extremum lines can capture the visually important regions on the underlying triangular meshes and are consilient with the low-level human visual attention for the underlying 3D models;
- A unified computing framework for extracting the extremum lines of surface field is proposed, which is considered as the ridge-valley lines along the principal curvature directions on triangular meshes;
- Compared with the common ridges and valleys based on the purely geometric curvature map, our extremum lines of the perceptual-saliency map can capture the visually salient shape features and yield visually pleasing results.

The paper is organized as follows. Some related works for generating different types of feature lines are reviewed in Sect. 2. Section 3 gives the definition of perceptual-saliency measure on triangle meshes and the motivation for extracting the perceptual-saliency extremum lines. A unified framework of extremum lines extraction is described in Sect. 4. Section 5 gives the algorithm for extracting perceptual-saliency extremum lines. Experimental results and discussions are given in Sect. 6. Finally, Sect. 7 concludes the paper and gives some future research directions.

2 Related work

Since line drawings can illustrate the 3D shapes efficiently and produce aesthetic appeal for a human, researchers have proposed many algorithms for generating different types of feature lines in non-photorealistic rendering [2, 3, 26]. These algorithms can be categorized according to whether they depend on user's viewpoint, i.e., view-dependent line approaches and view-independent line approaches.

2.1 View-dependent line approaches

As a means of conveying a 3D shape, the view-dependent lines on the surface depend not only on the differentials of the underlying surface, but also on the user's viewing direction. It means that the lines will be varied whenever the virtual camera changes its position or orientation.

The traditional silhouette lines are the loci of points at which the object normal is perpendicular to the viewing direction. Koenderink [14] first establishes the relationship between the apparent curvature of the contour in the image and the Gaussian curvature of the corresponding point on the surface. Gooch et al. [8] propose a non-photorealistic rendering algorithm to draw silhouettes for shading models, and also introduce an interactive technical illustration system for 3D mesh models. Hertzmann and Zorin [9] propose fast silhouette detection techniques for line drawing rendering of smooth surfaces by dual surface intersection, cusp and visibility detection, and smooth direction fields computation. Kalnins et al. [13] try to maintain frame-to-frame temporal coherency of stylized silhouettes by propagating line stroke parameterizations between adjacent frames.

Suggestive contours are composed of those points at which occluding contours appear with minimal change at current viewpoint. They will become true contours at "nearby" viewpoints [4]. A suggestive contour is precisely defined as the loci of points on the surface at which its radial curvature is zero, and the directional derivative of radial curvature along the view projection direction is positive. Furthermore, DeCarlo and Rusinkiewicz [6] introduce the variation and stability of suggestive contours on surfaces and propose interactive suggestive contour extraction and rendering algorithms. Furthermore, DeCarlo et al. [6] introduce two new object-space line definitions, namely suggestive highlights and principal highlights, which are related to suggestive contours and geometric creases. Apparent ridges, i.e., another object-space line definition, are defined as the ridges of view-dependent curvature [12]. They are the loci of points at which the large view-dependent curvature assumes a local maximum along the principal view-dependent curvature direction.

Other view-dependent line approaches draw lines of object in the projective image space by using image edge detection algorithms. Saito and Takahashi [27] adopt 2D image processing algorithms to draw discontinuities, edges, and iso-parametric lines from 2D rendering images of 3D models. Inspired by the edge detection techniques in 2D images processing, photic extremum lines [33] are adopted as the line-drawing primitives, which are the set of points where the variation of illumination in the gradient direction reaches a local maximum. Zhang et al. [37] propose the Laplacian lines in the image space that are the zero-crossing points of the Laplacian of surface illumination. It is similar to the corresponding Laplacian-of-Gaussian edge detector in digital image processing.

View-dependent curves look visually pleasing and hence are suitable for non-photorealistic rendering applications. However, the shapes of natural objects usually contain bumps, dips, and undulations of varying geometry, and it is not clear how these geometric features should be depicted with view-dependent lines.

2.2 View-independent line approaches

View-independent lines mean that the feature curves on the object do not change with respect to the user's viewing direction. This type of feature curves can depict important shape properties and can be applied to technical illustrations such as archeology, architecture, medicine, etc.

Ridges and valleys, i.e., the lines on a surface along which the surface bends sharply, are prominent shape descriptors. Interrante et al. [10] use ridge and valley lines as auxiliary means to enhance visualization of isosurfaces. Ohtake et al. [23] estimate and fit principal curvatures and their derivatives of 3D object using implicit function, then extract ridge and valley lines from the fit implicit function. They also adopt the curvature-based filtering to preserve the most significant lines. Belyaev and Anoshkina [1] extract ridge and valley lines from noisy 3D range data, where the estimated normals are de-noised via nonlinear diffusion and Canny-like non-maximum suppression. Yoshizawa et al. [34, 35] compute the crest lines on the mesh by using differential quantities defined on the mesh surface and its focal surface. Pauly et al. [24] extract ridge and valley lines from unstructured 3D point clouds by using a scale-space analysis to find out the most important feature lines at multi-scale spaces.

However, illustrating the object only by valleys (or ridges) is often insufficient, since they do not always convey its structure. Drawing both will overload the image with too many lines. While portraying important object properties, the aforementioned feature lines sometimes fail to capture relevant features, such as weak edges, highly curved edges, and noisy surfaces [17].

Other types of curves are parabolic curves, which partition the surface into hyperbolic and elliptic regions of positive and negative mean curvature [15]. They correspond to the zeros of the Gaussian curvature and mean curvature, respectively. Kolomenkin et al. [16] define a new class of view-independent curves—demarcating curves. They are the loci of points at which there is a zero crossing of the normal curvature in the curvature gradient direction. A demarcating curve can be viewed as a separation between a valley and a ridge.

Both the parabolic curves and the curves of zero-mean curvature are isotropic operators and suffer from similar flaws as isotropic edges in images (e.g., Laplacian), such as poor behavior at corners and inexact edge localization. Demarcating curves might be noisy when the curvature along the edge varies. Our proposed perceptual-saliency extremum lines in this paper will address these problems. However, as mentioned in Cole et al. [2, 3], there is no specific curve that can fit all applications.

3 Perceptual-saliency definition and motivation for extremum lines extraction

3.1 Perceptual-saliency measure on triangle meshes

Inspired by image saliency map proposed by Itti et al. [11], Lee et al. [18] introduce the concept of mesh saliency as a measure of regional importance for triangle meshes in computer graphics. However, in their approach the mesh saliency is expressed by a scale-dependent manner using a center-surrounding operator on the Gaussian-weighted mean curvatures. It is a time-consumer operation to estimate the mesh saliency. Liu et al. [19] proposed another mesh saliency definition by using a center-surrounding filter operator on Gaussian-weighted average of the mean curvature function and explored its application for extracting the salient critical points of 3D mesh. In this paper, we propose a perceptual-saliency measure, which is expressed as a center-surrounding bilateral filter operator on Gaussian-weighted average of local projection heights between vertex $\mathbf{v}$ and its neighbors, that is,

$$PS(\mathbf{v}) = \frac{\sum_{\mathbf{x}\in N_k(\mathbf{v})} W_c(\|\mathbf{v}-\mathbf{x}\|)W_s(|(\mathbf{v}-\mathbf{x})\cdot\mathbf{n}_v|)(\mathbf{v}-\mathbf{x})\cdot\mathbf{n}_v}{\sum_{\mathbf{x}\in N_k(\mathbf{v})} W_c(\|\mathbf{v}-\mathbf{x}\|)W_s(|(\mathbf{v}-\mathbf{x})\cdot\mathbf{n}_v|)}$$

where $\mathbf{n}_v$ is the normal of vertex $\mathbf{v}$, and $N_k(\mathbf{v})$ is the k-ring neighbors of vertex $\mathbf{v}$. Unlike the global-based smooth Morse function definition [19, 22], our perceptual-saliency measure on a vertex $\mathbf{v}$ is a weighted average of local projection heights of the neighborhood vertices (see Fig. 1). The weight depends not only on the spatial distance $\|\mathbf{v}-\mathbf{x}\|$, but also on the local projection height of its neighbor $\mathbf{x}$ on the $\mathbf{n}_v$, i.e. $f(\mathbf{x}) = (\mathbf{v}-\mathbf{x})\cdot\mathbf{n}_v$. The first term $W_c(\cdot)$ is a closeness smoothing filter, and is defined as the standard Gaussian filter with spatial parameter σ_c, i.e. $W_c(x) = \exp(\frac{-x^2}{2\sigma_c^2})$. The second term $W_s(\cdot)$ is the feature preserving weight function, and it is also defined as the standard Gaussian filter with range parameter σ_s, i.e. $W_s(x) = \exp(\frac{-x^2}{2\sigma_s^2})$. In our implementation, the selections of $\sigma_c = 16.0, \sigma_s = 0.8$ work well for our experimental results.

Figure 2(b) shows the traditional saliency map of brain model via the Gaussian-weighted average of mean curvatures between vertices and their neighbors [19]. In general,

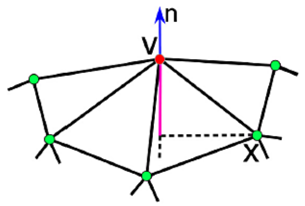


Fig. 1 Perceptual-saliency measure defined by a weighted average of local projection heights of the neighborhood vertices

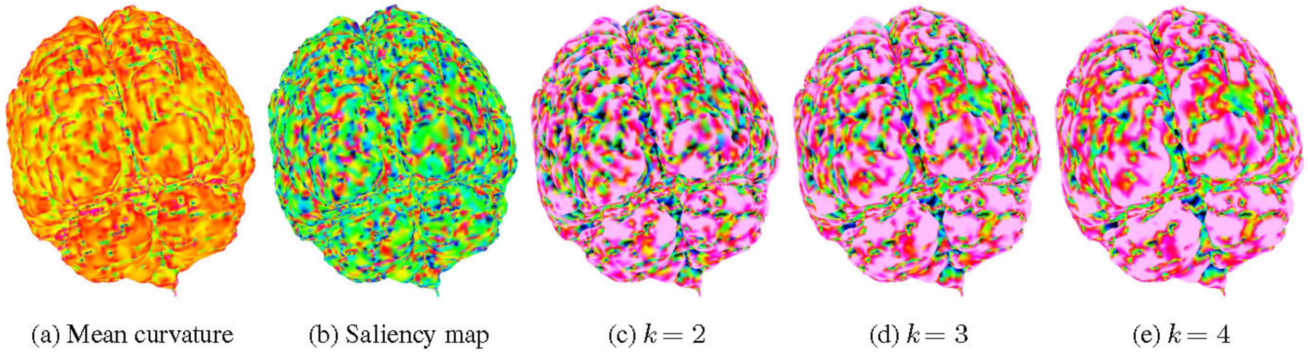


Fig. 2 The traditional saliency map (**b**) and our perceptual-saliency map of brain model via the different k-ring sizes of neighbors (**c**, **d**, **e**), in which the spatial and range parameters are selected as $\sigma_c = 16.0$, $\sigma_s = 0.8$, respectively

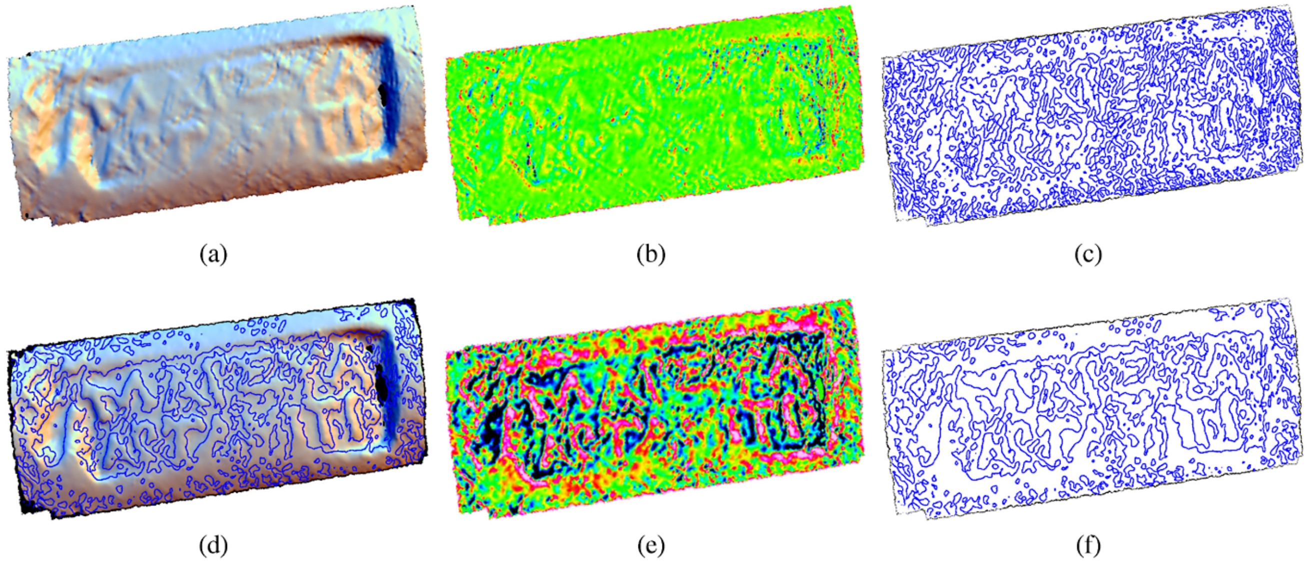


Fig. 3 The zero-crossings of mean curvature map and perceptual-saliency map of the stamped amphora handle model. (**a**) The shaded view of the model; (**b**) and (**c**) mean curvature map and the zero-crossing of mean curvature distribution of the model; (**d**) the zero-crossing of perceptual-saliency distribution on the shaded model; (**e**) and (**f**) perceptual-saliency map and the zero-crossing of perceptual-saliency distribution of the model

the traditional saliency distribution is inadequate for assessing visual saliency regions since it does not reflect the local context fully. However, the proposed perceptual-saliency measure can highlight efficiently the visually salient regions of the underlying model (see Fig. 2(d)), such as the longitudinal cerebral fissure and various sulci of human brain.

Figure 2(c, d, e) shows the different perceptual-saliency definitions via the different k-ring sizes of neighbors. A small k cannot efficiently extract the visual saliency regions. However, a large k means the smoothing effects under saliency measure on the 3D mesh. It tends to miss fine visually salient details and is time-consuming. In our implementation, the 3-ring neighbors for each vertex are adopted. It is a good trade-off between computational efficiency and perceptual saliency details.

3.2 Motivation for perceptual-saliency extremum lines extraction

In the literature of mathematics, the extremum lines of a field function defined on a surface manifold are the loci of extremum points of the field function. They are correlative with the zero-crossings of the surface field function [36]. Now, we consider the zero-crossings of mean curvature and perceptual-saliency distribution on a triangle mesh, respectively. The zero-mean curvature lines of a surface are composed of those points at which the mean curvature vanishes. They can divide the surface into regions of positive mean curvature and negative one (see Fig. 3(c) and Fig. 4(c)). But the positive and negative change of the mean curvature will concentrate on the purely geometric feature of the

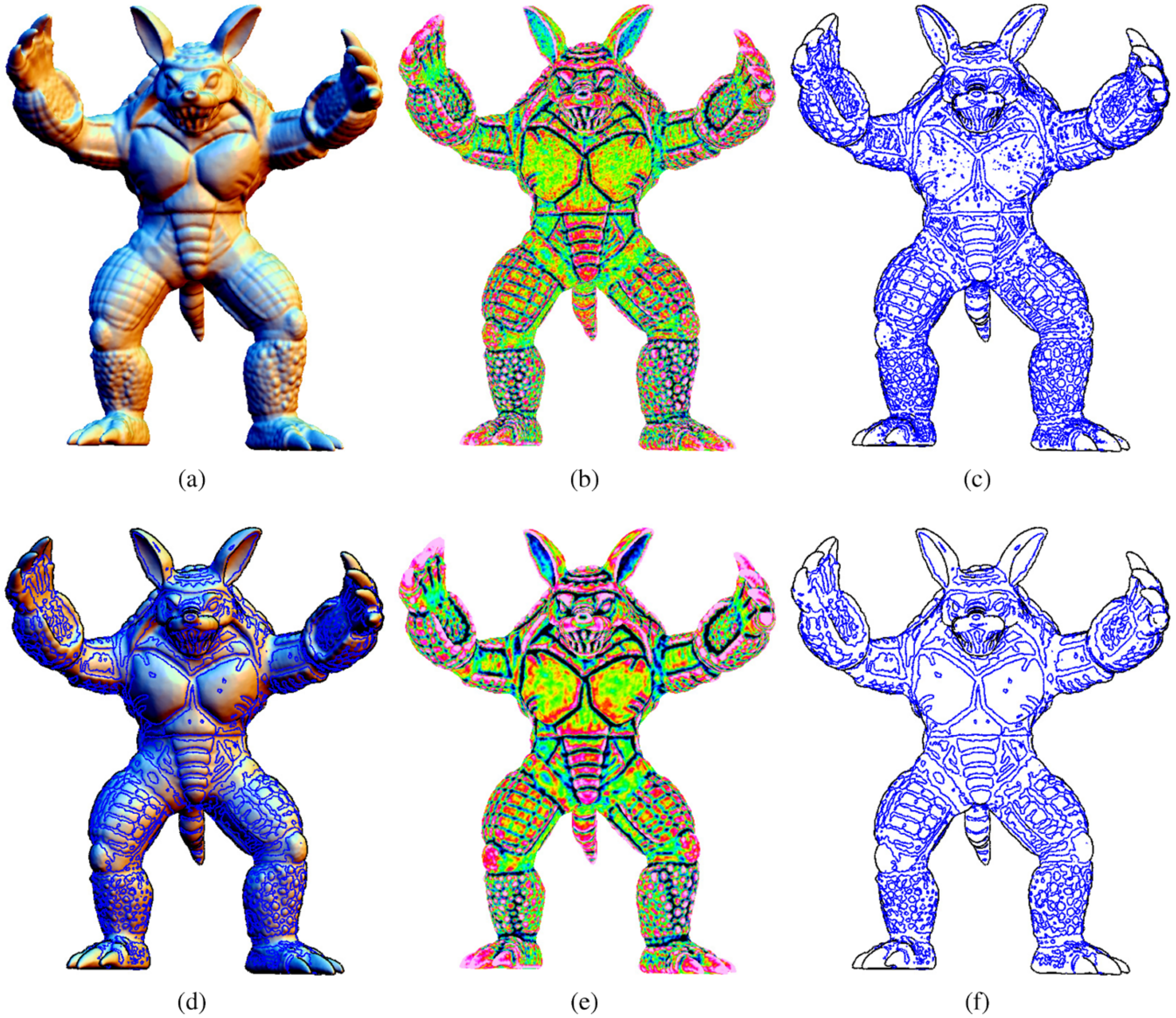


Fig. 4 The zero-crossings of mean curvature map and perceptual-saliency map of the Armadillo model. (**a**) The shaded view of the model; (**b**) and (**c**) mean curvature map and the zero-crossing of mean curvature distribution of the model; (**d**) the zero-crossing of perceptual-saliency distribution on the shaded model; (**e**) and (**f**) perceptual-saliency map and the zero-crossing of perceptual-saliency distribution of the model

underlying model, which is redundant for shape illustration.

However, since the visual attention is considered rather than purely geometric features, the perceptual salient feature described by our perceptual-saliency measure will be more suitable for shape illustration. The zero perceptual-saliency lines can partition the surface into different visually important regions (see Fig. 3(f) and Fig. 4(f)). The visual salient feature lines of 3D model always locate at the regions with extremal perceptual-saliency measure. Thus, the extraction of perceptual-saliency extremum lines is important for 3D shape illustration.

Figure 3(b, e) and Fig. 4(b, e) show the mean curvature map and perceptual-saliency map for stamped amphora handle model and Armadillo model, respectively. Compared with the purely geometric curvature measure, our perceptual-saliency measure can capture the visually important and interested regions on 3D models. The interested regions are consilient with the low-level human visual attention regions. As a result, the extremum lines of the perceptual-saliency map always produce visually important and pleasing results (see the brims of the Saddle model in Fig. 5).

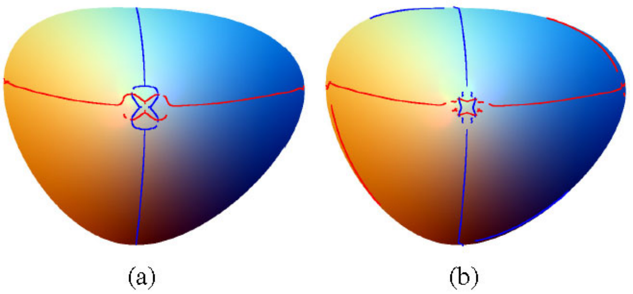


Fig. 5 The extremum lines of the Saddle model. (**a**) The extremum lines of mean curvature distribution for the Saddle model; (**b**) the extremum lines of perceptual-saliency distribution for the Saddle model. Here, the *red lines* mean the ridge lines of the corresponding surface fields, while the *blue lines* mean the valley lines of the corresponding surface fields

4 Overview of extremum lines extraction

4.1 Mathematical definition for extremum lines

Given a 3D smooth oriented surface, we can define surface differentials field and its perceptual-saliency field $F(x)$. If the surface is a triangle mesh, the field function $F(x)$ can then be represented as a piecewise linear function. Let $k_{\max}$, $k_{\min}$ be the maximal and minimal principal curvatures, and $\mathbf{t}_{\max}$, $\mathbf{t}_{\min}$ be the corresponding principal directions. From the knowledge of differential geometry [7], the extremum lines of surface field $F(x)$ are the loci of extremum points of the field $F(x)$ along the principal curvature directions $\mathbf{t}_{\max}$ and $\mathbf{t}_{\min}$. More precisely, there are two types of extremum lines, i.e. ridges and valleys. For simplicity, the ridges can be defined as the zero-crossings of the directional derivative of surface field $F(x)$ along the $\mathbf{t}_{\max}$, i.e.,

$$D_{\mathbf{t}_{\max}} F = 0, \quad \Delta F < 0.$$

Similarly, the valleys are defined as the zero-crossings of the directional derivative of surface field $F(x)$ along the $\mathbf{t}_{\min}$, i.e.,

$$D_{\mathbf{t}_{\min}} F = 0, \quad \Delta F > 0.$$

According to the definition of extremum lines, it is important to calculate the direction derivatives with respect to a particular tangent direction for the given surface field $F(x)$, which will be explained in the next section.

4.2 Calculating the directional derivatives along a given direction

Given a smooth oriented surface S and a smooth function $F(x): S \to \mathbb{R}$, the direction derivative $D_{\mathbf{v}} F(x)$ can be calculated as $D_{\mathbf{v}} F = \langle \nabla_S F, \mathbf{v} \rangle$, where $\nabla_S F$ is the gradient of function F. In particular, the direction $\mathbf{v}$ can be decomposed due to the orthogonal local frame at point $\mathbf{p}$ formed by the $\mathbf{w}$, $\mathbf{w}^{\perp}$, and surface normal $\mathbf{n}$, i.e.

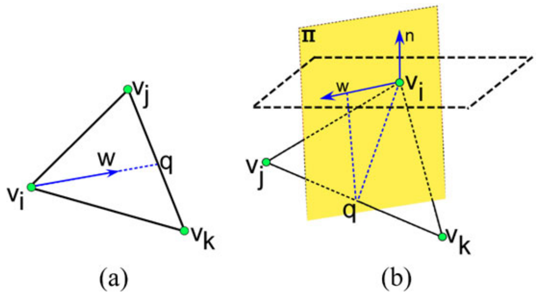


Fig. 6 The computation of directional derivatives along the given direction. (**a**) The direction derivative computation in the case of the given direction lies inside the triangle plane; (**b**) the direction derivative computation in the case of the given direction lies outside the triangle plane

$$\mathbf{v} = \cos\alpha \mathbf{w} + \cos\beta \mathbf{w}^{\perp} + \cos\gamma \mathbf{n}$$

where the angles α, β, γ mean the direction angles between the direction $\mathbf{v}$ and the corresponding local frame $\mathbf{w}$, $\mathbf{w}^{\perp}$ and $\mathbf{n}$, respectively. Thus, the directional derivative can be expressed as

$$\begin{aligned} D_{\mathbf{v}} F(x) &= \langle \nabla_S F, \cos\alpha \mathbf{w} + \cos\beta \mathbf{w}^{\perp} + \cos\gamma \mathbf{n} \rangle \\ &= \cos\alpha D_{\mathbf{w}} F(x) + \cos\beta D_{\mathbf{w}^{\perp}} F(x) + \cos\gamma D_{\mathbf{n}} F(x). \end{aligned}$$

According to differential geometry [7], the derivative $D_{\mathbf{n}} F = \langle \nabla_S F, \mathbf{n} \rangle = 0$. So, the directional derivative can be simplified to

$$D_{\mathbf{v}} F(x) = \cos\alpha D_{\mathbf{w}} F(x) + \cos\beta D_{\mathbf{w}^{\perp}} F(x).$$

Within one triangle on a mesh, if direction $\overrightarrow{\mathbf{vq}}$ is perpendicular to the $\mathbf{w}^{\perp}$, then $\cos\beta = 0$ and $\overrightarrow{\mathbf{vq}} = \cos\alpha \cdot \mathbf{w} + \cos\gamma \cdot \mathbf{n}$. The derivative $D_{\overrightarrow{\mathbf{vq}}} F = \cos\alpha D_{\mathbf{w}} F + \cos\gamma D_{\mathbf{n}} F = \cos\alpha D_{\mathbf{w}} F$. Thus, if we instead consider the derivative by the finite difference, the direction derivative can be approximated as

$$D_{\mathbf{w}} F = \frac{D_{\overrightarrow{\mathbf{vq}}} F}{\cos\alpha} = \frac{F(\mathbf{q}) - F(\mathbf{v})}{\|\overrightarrow{\mathbf{vq}}\| \cos\alpha}.$$

So, we can calculate the direction derivatives $D_{\mathbf{w}} F(x)$ along the tangent direction $\mathbf{w}$ as follows. For each vertex $\mathbf{v}_i$ on the triangular mesh, in the case the given direction lies inside the triangle plane, we simply adopt the finite difference method to calculate the directional derivatives $D_{\mathbf{w}} F(x)$ approximately (see Fig. 6(a)). Otherwise, according to the given tangent direction $\mathbf{w}$ at the vertex $\mathbf{v}_i$, we can construct the normal plane Π, which can be defined by the normal vector and the tangent direction at this vertex. Then, the intersection point $\mathbf{q}$ between the normal plane Π and the edge $\mathbf{e}_{jk} = \mathbf{v}_j\mathbf{v}_k$ can be determined. Within this triangle $\triangle_l = \mathbf{v}_i\mathbf{v}_j\mathbf{v}_k$, the direction derivative $D_{\mathbf{w}}^{\triangle_l}F(x)$ can be computed by the finite difference scheme, that is, the difference of surface field $F(\mathbf{q}) - F(\mathbf{v}_i)$ divided by the projection distance of $\overrightarrow{\mathbf{v}_i\mathbf{q}}$ along the given tangent direction $\mathbf{w}$ (see Fig. 6(b)). Finally, the direction derivative of $D_{\mathbf{w}}F(x)$ can be determined by the weighted average of the direction derivatives for all triangles $\triangle_l$ incident to the vertex $\mathbf{v}_i$. For the sake of simplicity, we take the weights to be the portion of the area of "barycentric cell" that lies closest to the vertex $\mathbf{v}_i$ [21, 25].

4.3 Framework of extremum lines extraction

Given a triangle mesh S, the procedure of extracting the extremum lines consists of the following steps:

- (Optional preprocessing) Smooth the normal field $\mathbf{n}$ and the surface field $F(x)$ of surface S;
- Compute the principal directions $\mathbf{t}_{\max}$ and $\mathbf{t}_{\min}$ at each vertex of the triangle mesh [30];
- Compute the directional derivative $D_{\mathbf{t}_{\max}}F$ and $D_{\mathbf{t}_{\min}}F$ of the surface field $F(x)$ with respect to the principal directions at each vertex;
- Detect the ridge vertices, i.e., zero-crossings of the directional derivative $D_{\mathbf{t}_{\max}}F = 0$, then filter out the vertices which are the local minima, s.t., $\Delta F \geq 0$, where ΔF can be computed as Laplace operator of the surface field $F(x)$ [30, 31];
- Detect the valley vertices, i.e., zero-crossings of the directional derivative $D_{\mathbf{t}_{\min}}F = 0$, then filter out the vertices which are the local maxima, s.t., $\Delta F \leq 0$;
- Trace the zero-crossings to get the extremum lines of the given surface field.

5 Extracting perceptual-saliency extremum lines from triangle meshes

According to the unified framework of the extremum lines extraction, we can extract the perceptual-saliency extremum vertex which is the zero-crossing of the directional derivative $D_{\mathbf{t}_{\max}}PS = 0$. For each mesh edge $\mathbf{e}_{ij} = \mathbf{v}_i\mathbf{v}_j$, it will contain a perceptual-saliency extremum vertex if $DPS(\mathbf{v}_i) = D_{\mathbf{t}_{\max}}PS(\mathbf{v}_i)$ and $DPS(\mathbf{v}_j) = D_{\mathbf{t}_{\max}}PS(\mathbf{v}_j)$ have opposite signs. The perceptual-saliency ridge vertex can then be approximated linearly by using following formula

$$\bar{\mathbf{v}} = \frac{\|DPS(\mathbf{v}_j)\|\mathbf{v}_i + \|DPS(\mathbf{v}_i)\|\mathbf{v}_j}{\|DPS(\mathbf{v}_i)\| + \|DPS(\mathbf{v}_j)\|}.$$

It is similar for the case of $D_{\mathbf{t}_{\min}}PS = 0$. For each triangle in the mesh, if two perceptual-saliency extremum vertices are detected on two triangle edges respectively, they will be connected by a line segment.

Before the ridge lines defined by the maxima of perceptual-saliency field $PS(x)$ are drawn, the vertices of the local minima should be filtered out, i.e. $\Delta PS \geq 0$. Even after that, there are still many feature lines left because our method finds all local extrema independent of whether the perceptual-saliency is high or low. To abstract a better line drawing, only those vertices with relatively high value of perceptual-saliency measure are adopted. Therefore we filtered the feature lines based on a threshold. The threshold can be determined by the mesh feature, e.g., the model's average edge length. In our implementation, the threshold is chosen as 10% of the average edge length. It is similar to drawing valley lines at the minima of perceptual-saliency field.

Obviously, instead of the perceptual-saliency field $PS(x)$, we can also extract the ridge and valley lines at maxima and minima of mean curvature field $H(x)$ by using the above procedure. The details are omitted here due to the length of the paper.

6 Experimental results and discussions

We have implemented the proposed algorithms on a PC with a Pentium IV 3.0 GHz CPU, 1024 M memory. The efficiency of extracting extremum lines defined via perceptual-saliency map or mean curvature is high for the triangle meshes. Since they are view-independent features, they can be extracted in the preprocessing step, i.e. before rendering step. However, some view-dependent features are more expensive to be extracted.

6.1 Perceptual-saliency extremum lines extraction

Because of their view-independent features, the extremum lines can be extracted efficiently in a preprocessing step for a given model. In the proposed framework, it takes about 0.7 to 4.8 seconds to extract the perceptual-saliency extremum lines from the meshes ranging from $85K$ to $525K$ triangles. Figures 7 and 8 show the perceptual-saliency extremum lines extracted from the lamp-nose model and Ottoman pipe model, respectively. They can capture the visually salient features efficiently for the underlying models, such as the central part of the lamp-nose model (see Fig. 7(d)) and the fine decoration details at the bottom section of the Ottoman pipe (see Fig. 8(d)).

6.2 Perceptual-saliency extremum lines vs. mean curvature extremum lines

The extremum lines of our perceptual-saliency map on the dense complex shapes can convey perceptually important information. Figure 9 shows the extremum lines for

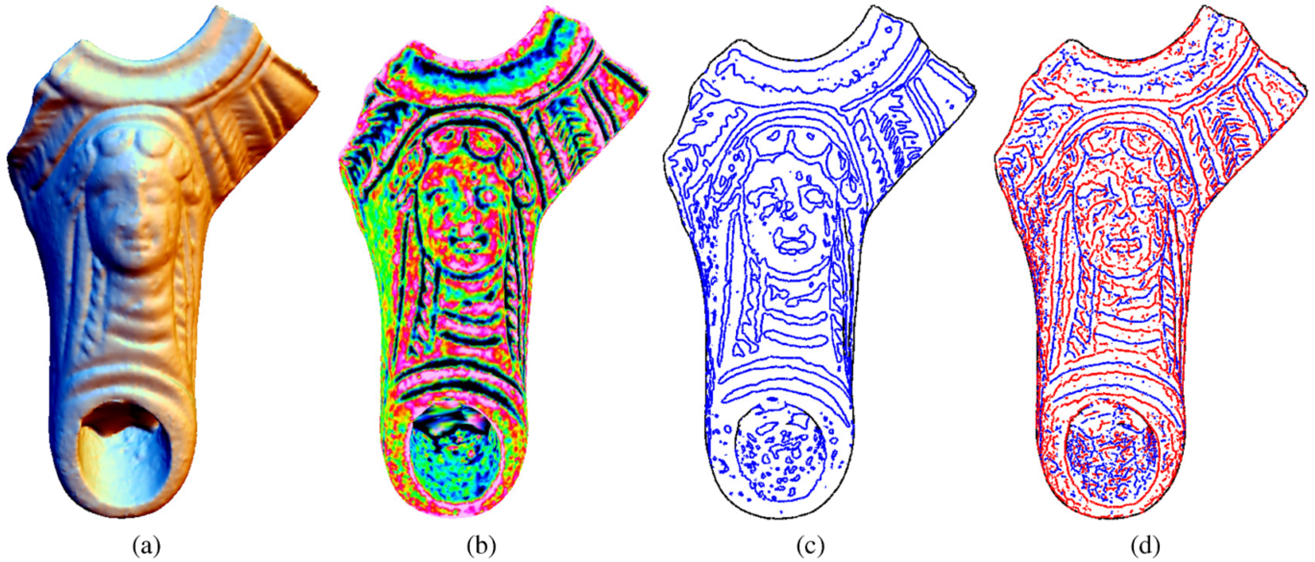


Fig. 7 The perceptual-saliency extremum lines extraction for lamp-nose model. (**a**) Lamp-nose model; (**b**) our perceptual-saliency map of lamp-nose model; (**c**) zero-crossing of perceptual-saliency distribution; (**d**) the perceptual-saliency extremum lines extracted from the lamp-nose model. In the perceptual-saliency extremum lines extraction (**d**), the *red lines* mean the ridge lines of the perceptual-saliency map, while the *blue lines* mean the valley lines of the perceptual-saliency map

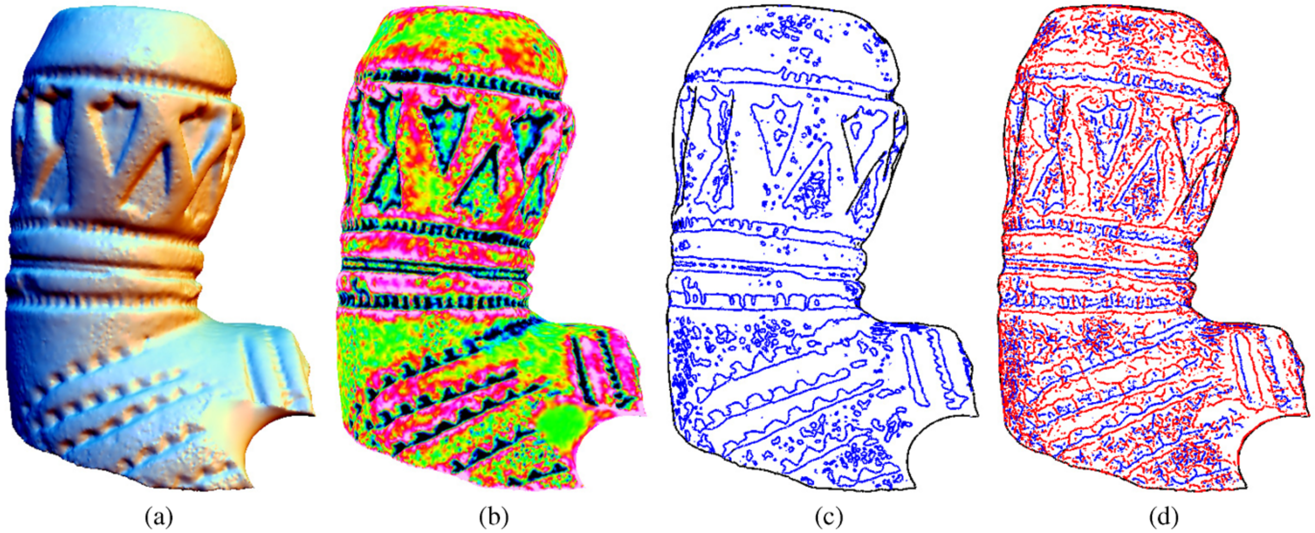


Fig. 8 The perceptual-saliency extremum lines extraction for Ottoman pipe model. (**a**) Ottoman pipe model; (**b**) our perceptual-saliency map of Ottoman pipe model; (**c**) zero-crossing of perceptual-saliency distribution; (**d**) the perceptual-saliency extremum lines extracted from the Ottoman pipe model. In the perceptual-saliency extremum lines extraction (**d**), the *red lines* mean the ridge lines of the perceptual-saliency map, while the *blue lines* mean the valley lines of the perceptual-saliency map

the Armadillo model, which include the ridge lines (in red) and the valley lines (in blue). They can capture both salient shape features and subtle shape variations. However, compared with the extremum lines of the purely geometric mean curvature map, the extremum lines of the perceptual-saliency map can illustrate the visually salient shape features more clearly because some redundant subtle shape variations unimportant for visual perceptual are omitted.

6.3 Perceptual-saliency extremum lines vs. ridges/valleys

Our perceptual-saliency measure is essentially an anisotropic smoothing operator for the local geometric variations. Thus, some redundant detail information can be smoothed. However, the common ridges and valleys (the extremum lines of the principal curvature map [10, 23]) will capture too much fine surface detail, which may be unimportant and unnecessary for visual perception. Figure 10 shows the ex-

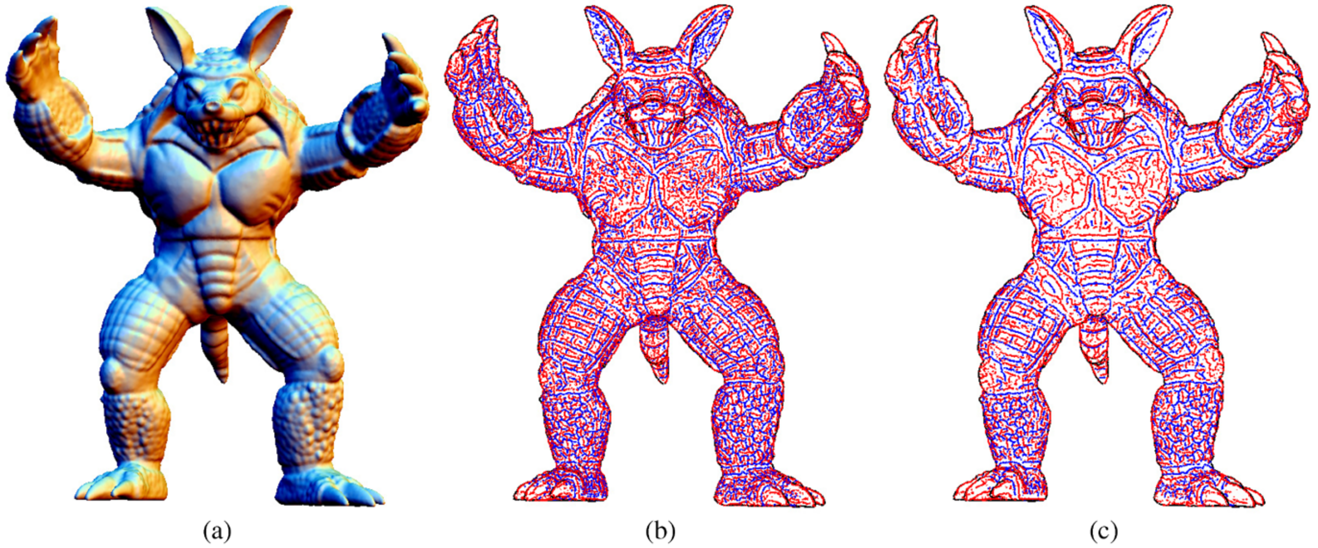


Fig. 9 The extremum lines of perceptual-saliency and mean curvature distributions for the Armadillo model. (**a**) Armadillo model; (**b**) the extremum lines extraction of mean curvature map for the Armadillo model; (**c**) the extremum lines extraction of perceptual-saliency map for the Armadillo model. Here, the *red lines* mean the ridge lines of the corresponding surface field, while the *blue lines* mean the valley lines of the corresponding surface field

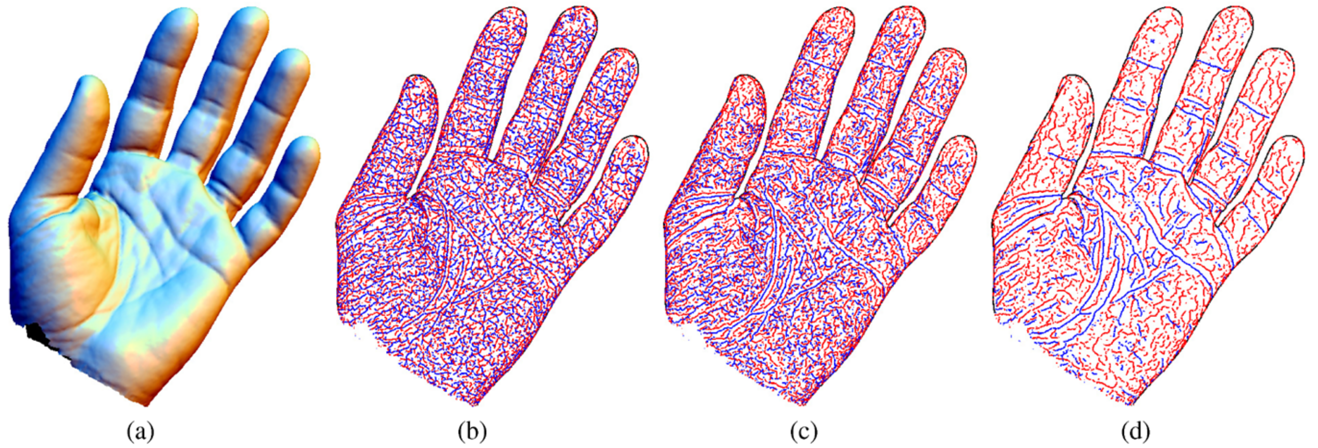


Fig. 10 The various ridge and valley lines of Olivier hand model. (**a**) Olivier hand model; (**b**) the ridge and valleys of principal curvature map of Olivier hand model; (**c**) the ridge and valleys of mean curvature map of Olivier hand model; (**d**) the ridge and valleys of perceptual-saliency map of Olivier hand model. Here, the *red lines* mean the ridge lines of the corresponding surface field, while the *blue lines* mean the valley lines of the corresponding surface field

tremum lines and the common ridges and valleys for the Olivier hand model. Compared with the common ridges and valleys based on the purely geometric curvature map, the extremum lines of the perceptual-saliency map can capture the visually salient shape features and yield more satisfied results, such as the palm-prints of the Olivier hand model (see Fig. 10(d)).

7 Conclusions and future work

In this paper, a new definition of perceptual-saliency measure is presented to capture the visually important and interesting regions on the underlying models. Meanwhile, a unified framework for extracting extremum lines is proposed, which can be considered as ridge-valley lines for various surface fields along the principal curvature directions. Under the extremum lines extraction framework, a new type of view-independent feature lines—perceptual-saliency extremum lines—are extracted, which defined as zero-crossing of directional derivative of the perceptual-saliency measure. Our extremum lines are computationally efficient and are promising for illustrating various 3D models, especially for archaeological artifacts.

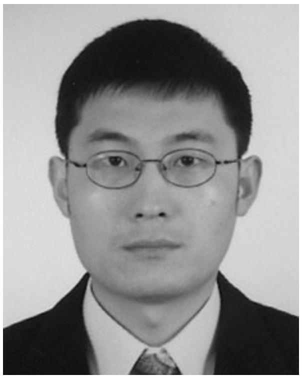
In the future, we intend to utilize the proposed line drawing techniques for shape analysis applications, such as best view selection, visualization enhancement, similarity-based retrieval, etc. In addition, we would also like to explore the utility of various types of drawing styles in archaeology applications.

Acknowledgements This work was supported by the 973 program of China under Grant No. 2009CB320801, the National Natural Science Foundation of China under Grant No. 60933007 and the Science and Technology Planning Project of Zhejiang Province under Grant No. 2009C34005. This work was also partly supported by the Natural Science Foundation of Zhejiang Province under Grant No. Z1090630. The 3D models are courtesy of Stanford University, Princeton University, Rutgers University, Haifa University, Hebrew University, and the Aim@Shape.

References

1. Belyaev, A., Anoshkina, E.: Detection of surface creases in range data. In: IMA Conference on the Mathematics of Surfaces XI. Lecture Notes in Computer Science, vol. 3604, pp. 50–61. Springer, Berlin (2005)
2. Cole, F., Golovinskiy, A., Limpaecher, A., Barros, H.S., Finkelstein, A., Funkhouser, T., Rusinkiewicz, S.: Where do people draw lines? ACM Trans. Graph. **27**(3), 88 (2008)
3. Cole, F., Sanik, K., DeCarlo, D., Finkelstein, A., Funkhouser, T., Rusinkiewicz, S., Singh, M.: How well do line drawings depict shape? ACM Trans. Graph. **28**(3), 28 (2009)
4. DeCarlo, D., Finkelstein, A., Rusinkiewicz, S., Santella, A.: Suggestive contours for conveying shape. ACM Trans. Graph. **22**(3), 848–855 (2003)
5. DeCarlo, D., Finkelstein, A., Rusinkiewicz, S.: Interactive rendering of suggestive contours with temporal coherence. In: International Symposium on Non-Photorealistic Animation and Rendering (NPAR) 2004, pp. 15–24
6. DeCarlo, D., Rusinkiewicz, S.: Highlight lines for conveying shape. In: International Symposium on Non-Photorealistic Animation and Rendering (NPAR) 2007, pp. 63–70
7. Do Carmo, M.P.: Differential Geometry of Curves and Surfaces. Prentice-Hall, Englewood Cliffs (1976)
8. Gooch, B., Sloan, P.J., Gooch, A., Shirley, P., Riesenfeld, R.F.: Interactive technical illustration. In: Proceedings of ACM Symposium on Interactive 3D Graphics 1999, pp. 31–38
9. Hertzmann, A., Zorin, D.: Illustrating smooth surfaces. In: Proceedings of ACM SIGGRAPH 2000, pp. 517–526
10. Interrante, V., Fuchs, H., Pizer, S.: Enhancing transparent skin surfaces with ridge and valley lines. In: Proceedings of IEEE Visualization 1995, pp. 52–59
11. Itti, L., Koch, C., Niebur, E.: A model of saliency based visual attention for rapid scene analysis. IEEE Trans. Pattern Anal. Mach. Intell. **20**(11), 1254–1259 (1998)
12. Judd, T., Durand, F., Adelson, E.H.: Apparent ridges for line drawing. ACM Trans. Graph. **26**(3), 19–26 (2007)
13. Kalnins, R.D., Davidson, P.L., Markosian, L., Finkelstein, A.: Coherent stylized silhouettes. ACM Trans. Graph. **22**(3), 856–861 (2003)
14. Koenderink, J.J.: What does the occluding contour tell us about solid shape? Perception **13**, 321–330 (1984)
15. Koenderink, J.J.: Solid Shape. MIT Press, Cambridge (1990)
16. Kolomenkin, M., Shimshoni, I., Tal, A.: Demarcating curves for shape illustration. ACM Trans. Graph. **27**(5), 1–9 (2008)
17. Kolomenkin, M., Shimshoni, I., Tal, A.: On edge detection on surfaces. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2009, pp. 2767–2774
18. Lee, C.H., Varshney, A., Jacobs, D.W.: Mesh saliency. In: Proceedings of ACM SIGGRAPH 2005, pp. 659–666
19. Liu, Y.-S., Liu, M., Kihara, D., Ramani, K.: Salient critical points for meshes. In: Proceedings of the ACM Symposium on Solid and Physical Modeling 2007, pp. 277–282
20. Maaten, L.J.P., Boon, P.J., Paijmans, J.J., Lange, A.G., Postma, E.O.: Computer vision and machine learning for archaeology. In: Computer Applications and Quantitative Methods in Archaeology 2006, pp. 112–130
21. Meyer, M., Desbrun, M., Schröder, M., Barr, A.H.: Discrete differential-geometry operators for triangulated 2-manifolds. In: Proceedings of VisMath, Berlin, Germany, 2002, pp. 35–57
22. Ni, X., Garland, M., Hart, J.C.: Fair Morse functions for extracting the topological structure of a surface mesh. In: Proceedings of ACM SIGGRAPH 2004, pp. 613–622
23. Ohtake, Y., Belyaev, A., Seidel, H.-P.: Ridge-valley lines on meshes via implicit surface fitting. ACM Trans. Graph. **23**(3), 609–612 (2004)
24. Pauly, M., Keiser, R., Gross, M.: Multi-scale feature extraction on point-sampled surfaces. Comput. Graph. Forum **22**(3), 281–290 (2003)
25. Rusinkiewicz, S.: Estimating curvatures and their derivatives on triangle meshes. In: Symposium on 3D Data Processing, Visualization, and Transmission 2004, pp. 486–493
26. Rusinkiewicz, S., Cole, F., DeCarlo, D., Finkelstein, A.: Line drawings from 3D models. In: Proceedings of ACM SIGGRAPH 2008, Course Notes
27. Saito, T., Takahashi, T.: Comprehensible rendering of 3-d shapes. In: Proceedings of ACM SIGGRAPH 1990, pp. 197–206
28. Strothotte, T., Schlechtweg, S.: Non-Photorealistic Computer Graphics: Modeling, Rendering and Animation. Morgan Kaufmann, San Francisco (2002)
29. Stylianou, G., Farin, G.: Crest lines for surface segmentation and flattening. IEEE Trans. Vis. Comput. Graph. **10**(5), 536–544 (2004)
30. Taubin, G.: Estimating the tensor of curvature of a surface from a polyhedral approximation. In: Proceedings of the Fifth International Conference on Computer Vision 1995, pp. 902–907
31. Taubin, G.: A signal processing approach to fair surface design. In: Proceedings of ACM SIGGRAPH 1995, pp. 351–358
32. Tood, J.T.: The visual perception of 3D shape. Trends Cogn. Sci. **8**(3), 115–121 (2004)
33. Xie, X., He, Y., Tian, F., Seah, H.-S.: An effective illustrative visualization framework based on photic extremum lines (PELs). IEEE Trans. Vis. Comput. Graph. **13**(6), 1328–1335 (2007)
34. Yoshizawa, S., Belyaev, A., Seidel, H.P.: Fast and robust detection of crest lines on meshes. In: ACM Symposium on Solid and Physical Modeling 2005, pp. 227–232
35. Yoshizawa, S., Belyaev, A., Yokota, H., Seidel, H.-P.: Fast and faithful geometric algorithm for detecting crest lines on meshes. In: Proceedings of Pacific Graphics 2007, pp. 231–237
36. Yuille, A.L.: Zero crossings on lines of curvature. Comput. Vision Graph. Image Process. **45**(1), 68–87 (1989)
37. Zhang, L., He, Y., Xie, X., Chen, W.: Laplacian lines for real-time shape illustration. In: Proceedings of ACM Symposium on Interactive 3D Graphics and Games 2009, pp. 129–136

Yongwei Miao is Associate Professor in the College of Computer Science and Technology, Zhejiang University of Technology, P.R. China. He received his Ph.D. degree in Computer Graphics from the State Key Lab of CAD&CG, Zhejiang University. From February 2008 to February 2009, he was a visiting scholar at University of Zurich, Switzerland. His research interests include virtual reality, digital geometry processing and non-photorealistic rendering.



Jieqing Feng is Professor of the State Key Lab of CAD&CG, Zhejiang University, P.R. China. He received his B.Sc. degree in Applied Mathematics from the National University of Defense Technology in 1992 and his Ph.D. degree in Computer Graphics from Zhejiang University in 1997. His research interests include geometric modeling, real-time rendering, computer animation and cartoon animation.