Robust Feature Detection and Local Classification for Surfaces Based on Moment Analysis

Ulrich Clarenz, Martin Rumpf, and Alexandru Telea

Abstract—The stable local classification of discrete surfaces with respect to features such as edges and corners or concave and convex regions, respectively, is as quite difficult as well as indispensable for many surface processing applications. Usually, the feature detection is done via a local curvature analysis. If concerned with large triangular and irregular grids, e.g., generated via a marching cube algorithm, the detectors are tedious to treat and a robust classification is hard to achieve. Here, a local classification method on surfaces is presented which avoids the evaluation of discretized curvature quantities. Moreover, it provides an indicator for smoothness of a given discrete surface and comes together with a built-in multiscale. The proposed classification tool is based on local zero and first moments on the discrete surface. The corresponding integral quantities are stable to compute and they give less noisy results compared to discrete curvature quantities. The stencil width for the integration of the moments turns out to be the scale parameter. Prospective surface processing applications are the segmentation on surfaces, surface comparison, and matching and surface modeling. Here, a method for feature preserving fairing of surfaces is discussed to underline the applicability of the presented approach.

Index Terms—Surface classification, surface processing, edge detection, nonsmooth geometry.

1 Introduction

Feature detection is known to be an indispensable tool in image processing. Features such as edges and corners have to be classified in a stable way to enable edge preserving image denoising [19], [1], [25] and robust segmentation of image subdomains bounded by edges [18], [2], [25]. Correspondingly, the local classification of surfaces, in particular the detection of edge and corner features or the distinction of concave and convex areas on the surface, turns out to be an important ingredient for many surface processing applications. Indeed, some applications are:

- *Surface fairing:* A given initial, noisy surface is smoothed, while edges on it are simultaneously preserved or even enhanced (cf. [23], [8], [4]).
- *Mesh decimation:* A given surface mesh is simplified while edge features are preserved (cf. [26], [11]).
- *Surface segmentation:* Identification of homogeneous regions, indicated by characteristics such as convexity and concavity, or bounded by feature lines.
- *Surface matching:* Surfaces are reduced to a skeleton of features lines to enable a better comparison.

In image processing, a straightforward identification of edges can be based on an evaluation of the image gradient. A sufficiently large gradient is supposed to indicate an edge. Alternatively, a frequently considered edge indicator is the Canny edge indicator, which searches for extrema of the second derivatives in the gradient direction [6]. Furthermore, the structure tensor (cf. [24]) enables a robust classification of edges and edge directions in images. On surfaces, gradients are no more natural objects for the identification of intrinsic surface characteristics. Here, the canonical quantity for the detection of edges is the curvature tensor, in the case of codimension 1, represented by the symmetric shape operator. An edge is supposed to be indicated by one sufficiently large principle curvature and the corresponding principle curvature direction is perpendicular to the edge on the surface. This method is well-suited for smooth objects. Curvature evaluation for discrete surfaces can be based on the algorithm proposed by Moreton and Séquin [17]. In particular, in surface processing, curvature evaluation is used to detect important features [4], [23], [14], [7]. Another method of computing the curvature on discrete data sets is given by [4], where the surface is locally projected on polynomial graphs over the tangent space. In [7], the mean curvature is defined as the first variation of the area functional. This approach is closely related to the minimal surface algorithm by Pinkall and Polthier [20]. Furthermore, in [16], discrete curvatures are computed, where the discrete Gaussian curvature is based on the quotient of a local spherical image and the corresponding local surface area. Curvature is an object which comes along with smoothness. Consequently, on discrete surfaces, we can separate areas of low curvature from nonsmooth areas. Nevertheless, we cannot obtain information about the structure of the nonsmooth part. For that reason, curvature estimation requires a tedious treatment and is unstable on irregular surfaces or

- *U. Clarenz and M. Rumpf are with the Institute für Mathematik, Duisburg University, Germany. E-mail: {clarenz, rumpf}@math.uni-duisburg.de.*
- *A. Telea is with the Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands. E-mail: alext@win.tue.nl.*

Manuscript received 30 Apr. 2003; revised 19 Dec. 2003; accepted 6 Apr. 2003.
For information on obtaining reprints of this article, please send e-mail to: tvcg@computer.org, and reference IEEECS Log Number TVCGSI-0030-0403.

on surfaces perturbed by noise. In addition, discrete curvature evaluation is a local process without a scale and therefore very noise sensitive (see [16, Fig6(c)]).

Here, in contrast to the curvature-based approach for surface classification, we propose a novel set of surface classification criteria based on local zero and first *moments*. This approach is related to the structure tensor approach for images [24] and provides a stable way to distinguish smooth regions from the vicinity of edges and corners on surfaces. Furthermore, it allows us to robustly extract information on the geometry close to, or even on a feature singularity (cf. Section 3). In [22], moments have already been applied for the computation of skeletons from distance function singularities in 2D and 3D. For an approach to skeletons of 3D point clouds representing surfaces, we refer to [15]. In this paper, we generalize the techniques presented in [22] and [5] to yield a general tool for the classification of feature singularities. A subsequent advantage of our method is that it comes along with an embedded scale factor that allows a simple and natural way for detecting important surface structures at or above a user-selected scale. In addition, this paper provides a detailed quantitative analysis of the local surface classifiers which we believe to be the backbone of the observed robustness in the application. Here, in particular, the different scalings of the zero moment in smooth and nonsmooth situations, respectively, and the corresponding scaling and the eigenvalues of the first moment are derived.

We apply the presented surface classification tool for a number of fairly distinct triangular surfaces of different origin, complexity, and resolution. Finally, as an application in actual surface processing, we consider edge preserving surface fairing. Hence, we apply an anisotropic geometric diffusion. Here, moment-based local classification is used to distinguish edges from smooth surface areas and to identify edge directions on the surface. For the corresponding background and literature, we refer to [4], [5].

The paper is organized as follows: In Section 2, we recall how surfaces may be classified using curvature information. In Section 3, the local moment analysis is presented. We give proofs of the fact that the zero moment scales quadratically in smooth domains of a surface, where it scales only linearly in nonsmooth domains (cf. Theorem 2 and Theorem 4) and that the scaling of the first moment is quadratically in smooth and nonsmooth areas. Indeed, to distinguish smooth from nonsmooth regions based on first moments requires a comparison of the eigenvalues of the first moment (cf. Theorem 3 and Theorem 5). Furthermore, we discuss the implementation on triangle meshes. In Section 4, we compare curvature and moment classification and demonstrate experimentally the robustness of the approach. The application in surface fairing is described in Section 4.2. Finally, in Section 5, we draw conclusions.

Notation. Let us summarize notations and conventions we are going to use in the sequel. We consider a parameter manifold $\mathcal{M}$ which essentially fixes the topological type of immersed surfaces $x : \mathcal{M} \to \mathbb{R}^{d+1}$. Hence, the actual surface is $x(\mathcal{M})$. With a slight misuse of notation, we will denote the surface x again by $\mathcal{M}$. The parameter on $\mathcal{M}$ is denoted by ξ. By $\mathcal{T}_\xi\mathcal{M}$, we denote the tangent space at ξ and by $\mathcal{TM}$ the tangent bundle of $\mathcal{M}$. For immersions x, the differential Dx induces, canonically, a metric on $\mathcal{M}$ via the relation $g(v,w) = Dx(v) \cdot Dx(w)$ which holds for all $v, w \in \mathcal{TM}$. Here, the scalar product in $\mathbb{R}^m$ is denoted by $\cdot$. In coordinates, we obtain $g_{ij} = g(\partial_i, \partial_j)$. Suppose $\int_\mathcal{M} f\, dA$ denotes the usual integral of a function f over $\mathcal{M}$. Thus, the area element dA is given by $\sqrt{\det g}\, d\xi$, where $g = (g_{ij})_{ij}$ is the metric tensor corresponding to $g(\cdot,\cdot)$. Integration over $\mathcal{M}$ leads to the L^2-scalar product of L^2-functions f, g on $\mathcal{M}$: $(f,g) := \int_\mathcal{M} f \cdot g\, dA$. We will make use of the gradient $\nabla_\mathcal{M}$ and the divergence $\operatorname{div}_\mathcal{M}$ on the manifold. For a function f on $\mathcal{M}$, the gradient $\nabla_\mathcal{M} f \in \mathcal{T}_\xi\mathcal{M}$ in $\xi \in \mathcal{M}$ is defined by $g(\nabla_\mathcal{M} f, w) = \frac{d}{dt} f(c(t))|_{t=0}$, where $c(t)$ is a curve on $\mathcal{M}$ with $c(0) = \xi$ and $\dot{c}(0) = w$. Furthermore the divergence is defined as the corresponding dual differential operator. In particular $\int_\mathcal{M} \operatorname{div}_\mathcal{M} v\, \zeta\, dA = -\int_\mathcal{M} g(v, \nabla\zeta)\, dA$. The manifolds $\mathcal{M}$ are assumed to be oriented and the normal mapping will be denoted by $n : \mathcal{M} \to S^d$, where d is the dimension of the manifold $\mathcal{M}$. This enables us to define the shape operator $S : \mathcal{T}_\xi\mathcal{M} \to \mathcal{T}_\xi\mathcal{M}$ by $g(S_{\mathcal{T}_\xi\mathcal{M}} w, v) := \partial_w n \cdot Dx(v)$. The trace of the shape operator is the classical mean curvature $h = \operatorname{tr} S_{\mathcal{T}_\xi\mathcal{M}}$. We often write $S = S_{\mathcal{T}_\xi\mathcal{M}}$ if a misunderstanding is ruled out. The Laplacian $\operatorname{div}_\mathcal{M} \operatorname{grad}_M$ is denoted by $\Delta_\mathcal{M}$. Finally, let us from now on use Einstein summation convention.

2 A Review of Curvature-Based Local Surface Classification

In this section, we briefly recall how to locally classify surfaces based on curvature analysis. The quantity for the detection of highly curved surface areas—namely, edges—is the curvature tensor: In the codimension 1 case, it is represented by the symmetric shape operator $S_{\mathcal{T}_\xi\mathcal{M}}$. An edge is supposed to be indicated by one sufficiently large eigenvalue of $S_{\mathcal{T}_\xi\mathcal{M}}$. The main drawback of this approach is that it involves derivatives of "noisy" data, which is usually a risky enterprise. In particular, in the interesting case of surfaces with sharp edges, these quantities are not even defined on the actual surface.

Thus, we have to stabilize the evaluation of the shape operator. This can either be done by a straightforward "geometric Gaussian" filter, which turns out to be a short time-step $\tau = \epsilon^2/2$ of mean curvature motion (for details cf. [4], [9]) or by local L^2 projection of the surface onto quadratic polynomial graphs. Let us denote this prefiltered surface by $\mathcal{M}_\epsilon$. Hence, for stability reasons, we compute a shape operator $S_{\mathcal{T}_\xi\mathcal{M}_\epsilon}$ on $\mathcal{M}_\epsilon$. Here, the parameter ϵ either indicates the "geometric Gaussian" filter-width or the size of the neighorhood, which we take into account for the local L^2 projection [21]. Let us emphasize that we are able to reduce noise by this filter operator. But, we might also remove features in a hard to predict way. Now, we introduce, for every point on $\mathcal{M}_\epsilon$, a classification tensor $a^\epsilon_{\mathcal{T}_\xi\mathcal{M}_\epsilon}$. It is supposed to be a symmetric, positive definite, linear mapping on the tangent space $\mathcal{T}_\xi\mathcal{M}_\epsilon$. Let us suppose that $w^{1,\epsilon}$, $w^{2,\epsilon}$ are the principal directions of curvature—the orthogonal eigendirections of the shape operator—and $\kappa^{1,\epsilon}$, $\kappa^{2,\epsilon}$ the principle curvatures—the corresponding eigenvalues. Then, we define the tensor $a^\epsilon_{\mathcal{T}_\xi\mathcal{M}_\epsilon}$ in the basis $\{w^{1,\epsilon}, w^{2,\epsilon}\}$ as follows:

$$a^{\epsilon}_{\mathcal{T}_\xi\mathcal{M}_\epsilon} = \begin{pmatrix} G(\kappa^{1,\epsilon}) & 0 \\ 0 & G(\kappa^{2,\epsilon}) \end{pmatrix}, \quad (1)$$

where the function G is given by $G(x) := \frac{1}{1+(x/\beta)^2}$. Here, β serves as a user-defined threshold parameter which classifies the significance of surface features. Hence, a point is supposed to belong to an edge if there is one principal direction of curvature on $\mathcal{M}_\epsilon$ with large curvature compared to β. If the second principal curvature is small w.r.t. β, we consider the first direction as being orthogonal to an edge on the surface. At corners, both principal curvatures of $\mathcal{M}_\epsilon$ are large. Summarizing this, our tensor leads to the following surface classification:

- *Smooth areas* are characterized by $a^{\epsilon}_{\mathcal{T}_\xi\mathcal{M}_\epsilon} \sim \mathrm{diag}[1,1]$.
- *Edges* are defined by $a^{\epsilon}_{\mathcal{T}_\xi\mathcal{M}_\epsilon} \sim \mathrm{diag}[1,0]$. In this case, the edge direction is given by $w^{2,\epsilon}$ and we assume $|\kappa^{1,\epsilon}| >> |\kappa^{2,\epsilon}|$.
- *Corners* are defined by $a^{\epsilon}_{\mathcal{T}_\xi\mathcal{M}_\epsilon} \sim \mathrm{diag}[0,0]$.

As a simple edge indicator, we can use the function $\mathcal{C}^{curv}_\epsilon(x) = \mathrm{tr}\, a^{\epsilon}_{\mathcal{T}_\xi\mathcal{M}_\epsilon}$. Depending on the threshold parameter β, edges and corners are given by $\mathcal{C}^{curv}_\epsilon(x) < 1$. However, the above classifier encounters serious problems on noisy data or data containing features (e.g., edges) at several spatial scales (cf. Section 4).

Finally, let us mention that we can make use of this classification tensor as a diffusion tensor for surface fairing. Generalizing the algorithm introduced by Dziuk [10] for mean curvature flow, we can define an anisotropic geometric diffusion process which smoothes the surface in regions being classified as smooth and preserves edges on the surface. In addition, we can allow smoothing along an edge. The corresponding generalized mean curvature flow is given by the following equation:

$$\partial_t x - \mathrm{div}_{\mathcal{M}}(a^{\epsilon}_{\mathcal{T}_\xi\mathcal{M}_\epsilon}\nabla_{\mathcal{M}} Mx) = 0.$$

For a more detailed discussion of this type of equation, we refer to Section 4.2 and to [4]. This approach generalizes the image processing methodology presented and discussed by Perona and Malik [19], Alvarez et al. [1], and Weickert [24]. Let us mention that our approach here differs from Kimmel's method [13], where denoising of textures on surfaces is discussed, whereas we consider the denoising of the surface itself.

3 Moment-Based Surface Analysis

In the following, we will introduce and discuss local surface classification based on zero and first order surface moments. This will, in particular, allow us to robustly distinguish smooth regions from the vicinity of edges on the curve or surface. To begin with, we introduce the corresponding definitions:

Definition 1. *For an embedding $x : \mathcal{M} \to I\!R^{d+1}$, the zero moment is given by the barycenter $M^0_\epsilon(x(\xi)), \xi \in \mathcal{M}$, of $x(\mathcal{M}) \cap B_\epsilon(x(\xi))$, where $B_\epsilon(x(\xi))$ is the Euclidean ϵ-ball in $I\!R^{d+1}$ centered at $x(\xi)$:*

$$M^0_\epsilon(x(\xi)) = \fint_{B_\epsilon \cap \mathcal{M}} x\, dA.$$

The parameter ϵ is called the scanning width. Furthermore, the first moment is defined as:

$$M^1_\epsilon(x(\xi)) = \fint_{B_\epsilon\cap\mathcal{M}} (x - M^0_\epsilon(x(\xi))) \otimes (x - M^0_\epsilon(x(\xi)))\, dA,$$

where $y \otimes z := (y_i z_j)_{i,j=1,\ldots,d+1}$.

Due to the definition via local integration, the zero and the first moment are expected to be robust with respect to noise. To study the effectiveness of our moment-based classifiers, we next examine their behavior on smooth (Section 3.1), respectively, nonsmooth (Section 3.2) surfaces.

3.1 Smooth Surface Case

We first show that a locally smooth surface is characterized by a *quadratic scaling* of the zero moment shift $n_\epsilon = M^0_\epsilon(x(\xi)) - x(\xi)$ and the first moment $M^1_\epsilon(x(\xi))$. Here, n_ϵ plays the role of a scaled approximate normal.

Indeed, for a smooth function η on a Euclidean ϵ-ball $B_\epsilon(0) \subset I\!R^d$, we have:

$$\begin{aligned}\fint_{B_\epsilon}\eta &= \fint_{B_\epsilon}\eta(0) + \nabla\eta(0)\cdot x \\ &\quad + \frac{1}{2}\nabla^2\eta(0)x\cdot x\, dx + o(\epsilon^2) \\ &= \eta(0) + \frac{1}{2}\fint_{B_\epsilon(0)}\nabla^2\eta(0)x\cdot x\, dx + o(\epsilon^2) \\ &= \eta(0) + \frac{1}{2}\lambda_i\fint_{B_\epsilon(0)} x_i^2 dx + o(\epsilon^2),\end{aligned}$$

where the λ_i for $i = 1,\ldots,d$ are the eigenvalues of $\nabla^2\eta(0)$. Note that, in the above equation, we used the fact that:

$$\int_{B_\epsilon}\nabla\eta(0)\cdot x\, dx = \sum_{i=1}^{d}\int_{B_\epsilon}\partial_i\eta(0)x_i\, dx$$

and $\int_{B_\epsilon} x_i\, dx = 0$. Therefore, we have:

$$\begin{aligned}&\fint_{B_\epsilon}\eta \\ &= \eta(0) + \frac{1}{2}\cdot\frac{1}{d}\fint_{B_\epsilon(0)}|x|^2 dx\cdot \mathrm{tr}\nabla^2\eta(0) + o(\epsilon^2) \\ &= \eta(0) + c(d)\,\epsilon^2\Delta\eta(0) + o(\epsilon^2),\end{aligned}$$

where the dimension depending constant $c(d)$ is given by $c(d) = 1/[2(d+2)]$.

Now, we consider the surface $\mathcal{M}$ in the vicinity of some point ξ on $\mathcal{M}$. In a first step, we replace the Euclidean ball $B_\epsilon(x(\xi)) \subset I\!R^{d+1}$ by a geodesic ball $\widetilde{B}_\epsilon(x(\xi)) \subset \mathcal{M}$ and compute the barycenter $\widetilde{M}^0_\epsilon(x(\xi))$ of $\widetilde{B}_\epsilon(x(\xi))$. To evaluate $\widetilde{\int}_{\widetilde{B}_\epsilon(x(\xi))} x(\xi) dA(\xi)$, we take into account normal coordinates, i.e.,

$$g_{ij}(0) = \delta_{ij}, \quad \partial_k g_{ij}(0) = 0.$$

For more details on normal coordinates see e.g., the textbook [12, p. 19]. Then, one obtains for the Laplacian on $\mathcal{M}$ at $x(\xi)$:

$$(\Delta_{\mathcal{M}}\eta)(x(\xi)) = (\partial_i\partial_i\eta)(0) = \Delta\eta(0). \tag{2}$$

Hence, for the difference $\widetilde{M}^0_\epsilon(x(\xi)) - x(\xi)$ we get. Note that we identify $x(\xi)$ and $x(0)$ as well as $\widetilde{B}_\epsilon(x(\xi))$ and $B_\epsilon(0)$ in the chart and on the surface, respectively. Due to the choice of normal coordinates, the Euclidean ball in $\mathbb{R}^d$ and the geodesic ball on $\mathcal{M}$ coincide via the chart map.

$$\begin{aligned}\widetilde{M}^0_\epsilon(x(\xi)) - x(\xi) &= \fint_{B_\epsilon(0)} x\,dx - x(0)\\ &= c(d)\epsilon^2\Delta x(0) + o(\epsilon^2)\\ &= c(d)\epsilon^2\Delta_{\mathcal{M}}x(\xi) + o(\epsilon^2)\\ &= -c(d)\epsilon^2 h(\xi)n(\xi) + o(\epsilon^2),\end{aligned}$$

where h is the mean curvature of $\mathcal{M}$. Here, we have taken into account the classical [3] relation between the Laplace-Beltrami operator applied to the coordinate mapping and the mean curvature vector:

$$\Delta_{\mathcal{M}}x = -h\,n.$$

Let us point out here that, for the Euclidean ball on the surface $B_\epsilon(x(\xi)) \cap \mathcal{M}$ and for the geodesic ball $\widetilde{B}_\epsilon(x(\xi))$, we have the relation

$$|\,|B_\epsilon(x(\xi)) \cap \mathcal{M}| - |\widetilde{B}_\epsilon(x(\xi))|\,| = O(\epsilon^{d+2}).$$

We observe an order $d+2$ instead of d due to the vanishing first order terms $\partial_k g_{ij}(0)$ for normal coordinates. Therefore, one can replace the geodesic ball by the Euclidean ball without losing the corresponding scaling order.

So far, we obtain:

Theorem 2. *Let $x : \mathcal{M} \to \mathbb{R}^{d+1}$ be an immersion. For $\xi \in \mathcal{M}$ consider a ball of radius ϵ with center $x(\xi)$ and the ϵ-normal $n_\epsilon(\xi) = M^0_\epsilon(x(\xi)) - x(\xi)$. Then, n_ϵ scales quadratically in ϵ and*

$$n_\epsilon(\xi) = -\epsilon^2 c(d) h(\xi) n(\xi) + o(\epsilon^2).$$

Using (2), we can give a corresponding scaling result for the first moment. Choosing $\eta = x_i x_j$, one easily evaluates the components of the first moment:

$$\begin{aligned}&\fint_{B_\epsilon\cap\mathcal{M}} x_i x_j\,dA\\ &= x_i(\xi)x_j(\xi) + c(d)\,\epsilon^2\,\Delta_{\mathcal{M}}(x_i x_j) + o(\epsilon^2)\\ &= x_i(\xi)x_j(\xi) + c(d)\,\epsilon^2\,(2\,\nabla_{\mathcal{M}}x_i\cdot\nabla_{\mathcal{M}}x_j\\ &\qquad + x_j\Delta_{\mathcal{M}}x_i + x_i\Delta_{\mathcal{M}}x_j) + o(\epsilon^2).\end{aligned}$$

Therefore, the scaling of the first moment is also quadratic:

$$\begin{aligned}&(M^1_\epsilon)_{ij}\\ &= \fint_{B_\epsilon\cap\mathcal{M}} x_i x_j\,dA - \fint_{B_\epsilon\cap\mathcal{M}} x_i\,dA \fint_{B_\epsilon\cap\mathcal{M}} x_j\,dA\\ &= 2\,c(d)\,\epsilon^2\nabla_{\mathcal{M}}x_i\nabla_{\mathcal{M}}x_j + o(\epsilon^2)\\ &= 2\,c(d)\,\epsilon^2(\Pi_{\mathcal{T}_\xi\mathcal{M}})_{ij} + o(\epsilon^2),\end{aligned}$$

where $\Pi_{\mathcal{T}_\xi\mathcal{M}}$ is the linear projection onto the tangent space of $\mathcal{M}$ at point $x(\xi)$. More precisely, $\Pi_{\mathcal{T}_\xi\mathcal{M}}$ is the matrix representation w.r.t. the canonical basis of $\mathbb{R}^{d+1}$ and $(\Pi_{\mathcal{T}_\xi\mathcal{M}})_{ij}$ is the corresponding matrix entry. We can state:

Theorem 3. *For an embedding $x : \mathcal{M} \to \mathbb{R}^{d+1}$, the first moment w.r.t. $x(\xi)$ and scanning width ϵ scales quadratically:*

$$M^1_\epsilon(x(\xi)) = 2\,c(d)\,\epsilon^2\Pi_{\mathcal{T}_\xi\mathcal{M}} + o(\epsilon^2).$$

3.2 Nonsmooth Surface Case

We now discuss the case of nonsmooth surface features, such as edges and corners. To this aim, let $x : \mathcal{M} \to \mathbb{R}^{d+1}$ be a Lipschitz continuous immersed surface, which is smooth up to a one-dimensional subset $\Sigma_{\mathcal{M}}$. Here, $\Sigma_{\mathcal{M}}$ is the edge set on the surface. With respect to the scaling behavior of the shift n_ϵ of the zero moment centered at $x_0 = x(\xi_0)$, $\xi_0 \in \Sigma_{\mathcal{M}}$, it suffices to assume that, for a small open domain $U = U(x_0) \subset \mathbb{R}^{d+1}$, the set $\mathcal{M}\cap U$ is of cone type, i.e., $x \in \mathcal{M}\cap U$ implies $\alpha(x - x_0) + x_0 \in \mathcal{M}\cap U$ for all $\alpha \in [0,1)$. This is a first order, i.e., linear approximation of the curvilinear case. Furthermore, consider ϵ_0 such that $B_{\epsilon_0}(x_0) \subset U(x_0)$ and let $\epsilon_1, \epsilon_2 < \epsilon_0$. If we assume for a moment $x_0 = 0$, we derive from the cone property the following identity:

$$\begin{aligned}&\epsilon_1^{-d-1}\int_{B_{\epsilon_1}\cap\mathcal{M}} x\,dA = \epsilon_1^{-d}\int_{B_{\epsilon_1}\cap\mathcal{M}} x/\epsilon_1\,dA\\ &= \epsilon_2^{-d}\int_{B_{\epsilon_2}\cap\mathcal{M}} x/\epsilon_2\,dA = \epsilon_2^{-d-1}\int_{B_{\epsilon_2}\cap\mathcal{M}} x\,dA,\end{aligned}$$

and this leads to

$$\epsilon_1^{-1}\fint_{B_{\epsilon_1}\cap\mathcal{M}} x\,dA = \epsilon_2^{-1}\fint_{B_{\epsilon_2}\cap\mathcal{M}} x\,dA. \tag{3}$$

These equations are a result of the self-similarity of a cone. For the general case, where x_0 is not necessarily the origin, we have the following result:

Theorem 4. *Let $x : \mathcal{M} \to \mathbb{R}^{d+1}$ be a Lipschitz continuous embedding that is smooth on $\mathcal{M} - \Sigma_{\mathcal{M}}$; then, for $\xi_0 \in \Sigma_{\mathcal{M}}$, consider a ball of radius ϵ with center $x_0 = x(\xi_0)$ and $n_\epsilon(\xi_0) = M^0_\epsilon(x(\xi_0)) - x(\xi_0)$. Then, there is a vector $\tilde{\boldsymbol{n}}$ such that*

$$n_\epsilon(\xi_0) = \epsilon\,\tilde{\boldsymbol{n}} + o(\epsilon).$$

In the case $d = 1$, we are able to compute the length of the above vector $\tilde{n}$. This enables us to determine the apex angle of an edge of a given curve explicitly. The corresponding computation and result can be found in [5].

Next, let us consider the first moment and consider the following special situation for $d = 2$:

Let $D_\epsilon = \{(x_1, x_2, x_3) \in \mathbb{R}^3 | x_3 = 0, x_1 \geq 0, x_1^2 + x_2^2 \leq \epsilon^2\}$ be a semidisc in $\mathbb{R}^3$. Rotating D_ϵ around the x_2-axis by an angle φ, resp. $-\varphi$, we obtain two semidiscs S^1_ϵ and S^2_ϵ, respectively. Now, we compute the first moment of the union $S = S^1_\epsilon \cup S^2_\epsilon$. This tensor coincides up to higher order terms with the actual first moment $M^1_\epsilon(x(\xi))$, where $x(\xi)$ is contained in the singularity set $\Sigma_{\mathcal{M}}$ of the Lipschitz continuous surface, where S^1_ϵ and S^2_ϵ are the two tangential planes at $x(\xi)$. Here, due to the invariance of the first moment w.r.t. translations, we assume that $x(\xi_0) = 0$. We observe that, for $\epsilon_1, \epsilon_2 < \epsilon$, one has

$$\epsilon_1^{-2} \fint_{S\cap B_{\epsilon_1}} x \otimes x \, dA = \epsilon_2^{-2} \fint_{S\cap B_{\epsilon_2}} x \otimes x \, dA.$$

By Theorem 4, one obtains for the zero moment $M^0_{\epsilon_1}$ of $S \cap B_{\epsilon_1}$:

$$M^0_{\epsilon_1} \otimes M^0_{\epsilon_1} = \epsilon_1^2 \, \tilde{\boldsymbol{n}} \otimes \tilde{\boldsymbol{n}} + o(\epsilon_1^2).$$

Hence, we obtain *quadratic scaling* of the first moment, as in the smooth case.

We now explicitly compute the eigenvalues of the first moment of S. Because of the above considerations concerning the scaling behavior, we set $\epsilon = 1$ as well as $S_1^1 = S^1$, $S_1^2 = S^2$, $D_1 = D$. The two subsets S^1 and S^2 are of the same area. Thus, we can express the first moment $M^1(S)$ of S by the first moments of S^1 and S^2 and an additional correction term:

$$\begin{aligned}
&M^1(S) = M^1(S^1 \cup S^2)\\
&= \frac{1}{4} M^1(S^1) + \frac{1}{4} M^2(S^2) + \frac{1}{4} \Big\{ \fint_{S^1} x \otimes x \, dA\\
&\quad - \fint_{S^1} x \, dA \otimes \fint_{S^2} x \, dA + \fint_{S^2} x \otimes x \, dA\\
&\quad - \fint_{S^2} x \, dA \otimes \fint_{S^1} x \, dA \Big\}\\
&= \frac{1}{2} M^1(S^1) + \frac{1}{2} M^2(S^2) + \frac{1}{4} \Big\{ \fint_{S^1} x \, dA \otimes \fint_{S^1} x \, dA\\
&\quad - \fint_{S^1} x \, dA \otimes \fint_{S^2} x \, dA + \fint_{S^2} x \, dA \otimes \fint_{S^2} x \, dA\\
&\quad - \fint_{S^2} x \, dA \otimes \fint_{S^1} x \, dA \Big\}\\
&= \frac{1}{2} M^1(S^1) + \frac{1}{2} M^2(S^2) + \{TPT\},
\end{aligned} \tag{4}$$

where, by TPT, we denote all tensor product terms above. The first moment of the semidisc D in the x_3-plane is

$$\begin{aligned}
M^1(D) &= \begin{pmatrix} 1/4 - [4/(3\pi)]^2 & 0 & 0 \\ 0 & 1/4 & 0 \\ 0 & 0 & 0 \end{pmatrix}\\
&= \begin{pmatrix} \delta \approx 0.0699 & 0 & 0 \\ 0 & \gamma = 0.25 & 0 \\ 0 & 0 & 0 \end{pmatrix}.
\end{aligned}$$

Introducing the matrix $Q = \mathrm{diag}(1, 1, -1)$ one gets $M^1(S^2) = Q\, M^1(S^1)\, Q^T$. Analogous relations are valid for each tensor product term of TPT in (4). Finally, by taking into consideration these arguments, we obtain:

$$\begin{aligned}
M^1(S) &= \begin{pmatrix} \delta \cos^2\varphi & 0 & 0 \\ 0 & \gamma & 0 \\ 0 & 0 & \delta \sin^2\varphi \end{pmatrix}\\
&\quad + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \left(\fint_{S^1} x_3 \, dA\right)^2 \end{pmatrix}.
\end{aligned}$$

Because of the equation $\delta \sin^2\varphi + \left(\fint_{S^1} x_3 \, dA\right)^2 = \gamma \sin^2\varphi$, we finally obtain:

Theorem 5. *Let $x : \mathcal{M} \to \mathbb{R}^3$ be a $C^{0,1}$-surface which is smooth up to a one-dimensional set $\Sigma_{\mathcal{M}} \subset \mathcal{M}$. We assume that, for $\xi_0 \in \Sigma_{\mathcal{M}}$, the surface $x(\mathcal{M})$ has an edge of apex angle 2φ. In that case, the first moment $M^1_\epsilon(x(\xi_0))$ scales quadratically as in the smooth case. Furthermore, the eigenvalues of the first moment are $\epsilon^2\gamma$, $\epsilon^2\gamma \sin^2\varphi$, and $\epsilon^2\delta\cos^2\varphi$ (up to higher order terms) if γ and δ are the eigenvalues of the first moment of D.*

Let us summarize what we have obtained so far: The zero moment shift n_ϵ on surfaces scales quadratically with respect to the scanning width ϵ in smooth surface areas, whereas it scales linearly in nonsmooth areas. Even though the scaling behavior of the first moment in the smooth and the nonsmooth case is identical, the eigenvalues give additional information on the presence of an edge and the corresponding edge angle. This justifies the usage of moments as detectors for surface features. For a given, usually small, parameter ϵ, only features larger than ϵ will be detected. In Section 4.1, we will derive feature classifiers based on these results and illustrate their performance by a number of examples.

3.3 Implementation of Zero and First Moment Computation

Above, we have treated arbitrary surfaces. In the applications, we usually deal with two-dimensional, irregular, triangular grids. In the following, we will detail the discretization of the presented local surface classification in this case. Hence, we consider a polyhedron $\mathcal{M}_h$ consisting of triangles. In our implementation, we compute the moments centered at each node of the triangulation.

Let us fix one node X_i and denote the moments by $M^0_{\epsilon,h}$ and $M^1_{\epsilon,h}$. Here, h indicates the grid size and ϵ is, as before, the radius of the corresponding Euclidean ball. Given this radius ϵ, first of all, one collects all triangles $\{T_1, \ldots T_m\}$ of the triangulation such that $T_i \cap B_\epsilon(X_i) \neq \emptyset$, $i = 1, \ldots m$. This set of triangles splits into two subsets. The first one —denoted by $\mathcal{T}^o$—consists of all elements with $T_i \cap B_\epsilon = T_i$. The second one, $\mathcal{T}^\partial$, is supposed to be the complement, i.e., $T_i \in \mathcal{T}^\partial$ implies $T_i \cap B_\epsilon \neq \emptyset$. Now, we iteratively compute the integrals $\fint_{\mathcal{T}^o} x \, dA$ and $\fint_{\mathcal{T}^o} x \otimes x \, dA$. To this aim, we use the following relation for averaged integrals over disjoint sets A, B:

$$\fint_{A\cup B} f = \frac{|A|}{|A|+|B|} \fint_A f + \frac{|B|}{|A|+|B|} \fint_B f. \tag{5}$$

On each triangle of $\mathcal{T}^0$, we use the following exact integration formulas:

$$\begin{aligned}
M^0(T_i) &= \frac{1}{3}(X_0 + X_1 + X_2),\\
\fint_{T_i} x \otimes x \, dA &= \frac{1}{3}(Y_0 \otimes Y_0 + Y_1 \otimes Y_1 + Y_2 \otimes Y_2),
\end{aligned}$$

where X_0, X_1, X_2 are the nodes of T_i and $Y_0 = (X_0 + X_1)/2$, $Y_1 = (X_1 + X_2)/2$, and $Y_2 = (X_0 + X_2)/2$. For the corresponding computations on $\mathcal{T}^\partial \cap B_\epsilon$, we apply an approximation. For each triangle $T_l \in \mathcal{T}^\partial$, the intersection of the sphere ∂B_ϵ and the edges of the triangle consists of two points, denoted by P_1, P_2. We replace the curvilinear connection $T_l \cap \partial B_\epsilon$ by the line segment connecting P_1 and P_2. Hence, we replace $T_l \cap B_\epsilon$ by a polygon which we again

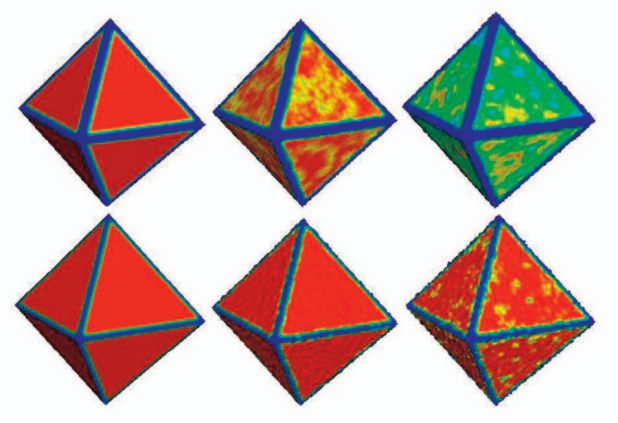


Fig. 1. From left to right: zero moment classifier (top row) and combined classifier (bottom row) for white noise perturbations of maximal amplitudes h, 2h, 3h in normal direction (from left to right). We always choose the scanning width $\epsilon = 4h$.

can split into triangles. Now, we proceed as above using exact integration on all these virtual triangles. Using (5), one is able to compute the average integrals over the quadrilaterals. Next, once again, we iteratively compute $⨍_{\mathcal{T}^\partial} x\, dA$ and $⨍_{\mathcal{T}^\partial} x \otimes x\, dA$. Finally, we get

$$M^0_{\epsilon,h} = ⨍_{\mathcal{T}^o \cup \mathcal{T}^\partial} x\, dA = \frac{|\mathcal{T}^o|}{|\mathcal{T}^o| + |\mathcal{T}^\partial|} M^0(\mathcal{T}^o) + \frac{|\mathcal{T}^\partial|}{|\mathcal{T}^o| + |\mathcal{T}^\partial|} M^0(\mathcal{T}^\partial)$$

and an analogous relation for $⨍_{\mathcal{T}^o \cup \mathcal{T}^\partial} x \otimes x\, dA$ and achieve

$$M^1_{\epsilon,h} = ⨍_{\mathcal{T}^o \cup \mathcal{T}^\partial} x \otimes x\, dA - M^0_{\epsilon,h} \otimes M^0_{\epsilon,h}.$$

4 Feature Classification and Applications

In this section, we will derive feature detectors on surfaces and experimentally verify their robustness. We propose the following two classification methods:

- *zero moment classification:* $\mathcal{C}^0_\epsilon = G(|n_\epsilon|/\epsilon)$,
- *combined zero and first moment classification:* $\mathcal{C}^{0,1}_\epsilon = G(\frac{|n_\epsilon|}{\epsilon}\frac{\lambda_{min}}{\lambda_{max}})$ with $\lambda_{min}, \lambda_{max}$, the smallest and largest eigenvalue of $M^1_\epsilon(x)$.

Here, the function G is $G(s) = \frac{1}{\alpha + \beta s^2}$ with suitably chosen $\alpha, \beta > 0$. The quotient $\lambda_{min}/\lambda_{max}$ is approximatively given by Theorem 5:

$$\lambda_{min}/\lambda_{max} = \delta/\gamma \cos^2\varphi = 0.2796 \cos^2\varphi,$$

where 2φ is the apex angle as in Theorem 5. This relation for $\lambda_{min}/\lambda_{max}$ is valid for φ larger than $0.2726 \approx 16^o$. Especially, in the smooth case ($\varphi = \pi/2$), this quotient vanishes where it increases for decreasing φ. For φ smaller than 16^o, the quotient again tends to 0 when $\varphi \to 0$. In this sense, very sharp features will be detected in a weaker sense as they should be. Nevertheless, as our experiments show, this seems to be a theoretical detail. In real-world applications, the above observation does not play an important role and is partly compensated by the zero moment.

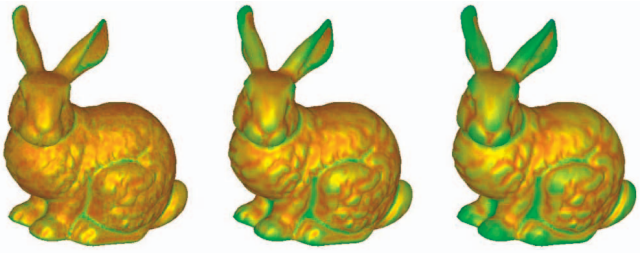


Fig. 2. Different scales of the zero moment shift on a surface choosing $\epsilon = 0.02$ on the left, 0.04 in the middle, and 0.06 on the right. The diameter of the object is scaled to 1.

First, we show the classifiers on various data sets (Section 4.1) and compare the zero and first moment classifiers with respect to robustness, detail detection, and computation time. In particular, it turns out that the combined classification is performing much better. Indeed, already, for small parameters ϵ, we obtain a robust feature detection. Let us mention that the computational cost for a single moment evaluation on a vertex of the surface $\mathcal{M}_h$ is $O(\frac{\epsilon^2}{h})$. Hence, the cost is large for a scanning width ϵ which is relatively large compared to the grid size h. As we will see, one obtains satisfying results for typical meshes $\mathcal{M}_h$ and $\epsilon \approx 3h, 4h$. Finally, we present an application of the proposed detectors for surface denoising (Section 4.2).

4.1 Classification Results

At first, we test our classification in Fig. 1 on a sequence of octahedra with an increasing noise level. In this case, the scale, i.e., the scanning width is always the same. Fig. 2 shows the detection of details on different scales choosing a different scanning width ϵ. Here, the norm of the linearly rescaled zero moment shift $n_\epsilon(\xi) = M^0_\epsilon(\xi) - x(\xi)$ is color coded from red (low) to green (high). Next, we consider the classification of human cortices. Here, the focus is on visually separating convex and concave areas of the surface, which is a difficult task due to the complicated local geometry. Potential applications are the segmentation of certain domains on the cortex, analyzing the course of diseases as, e.g., Alzheimer's disease, and, perspectively, the matching of different cortices.

Fig. 3 shows the classification of a human cortex using our zero moment classifier, on the same red to green colormap as in Fig. 2. Here, we use a slightly improved color coding. To be able to distinguish sulci from gyri ("mountains and valleys" or convex and concave areas, respectively), we, in addition, consider the sign of the scalar product $n_\epsilon(\xi) \cdot n(\xi)$, where $n(\xi)$ is an averaged oriented

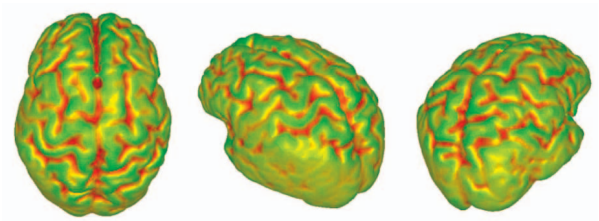


Fig. 3. Human cortex classification using the zero moment shift, different viewpoints. The scanning width is $\epsilon = 0.05$, where the diameter of the object is scaled to 1.

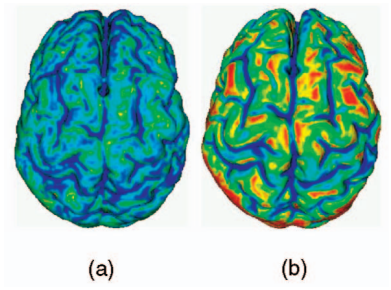


Fig. 4. Zero moment (a) and combined (b) classifiers for the human cortex in Fig. 3.

normal on the surface. The surface was generated by a marching cube algorithm and consists of approximately 130,000 triangles.

Fig. 4 compares both types of surface classifications. Here, the scanning width is $\epsilon = 3h$, where h is the average diameter of the data set triangle cells. This corresponds to a fraction of 0.015 of the object size. We notice that, by the combined zero and first moment classification, we obtain the best results. The zero moment classification delivers a significantly weaker result for the same scanning width. Although, for a larger scanning width, one is able to detect the major edges (cf. Fig. 3), the combined classifier visualizes and separates fine structures much better. Furthermore, Fig. 5 shows a comparison of the classification based on moments and based on local curvature computation as introduced in Section 2 (here, based on the local L^2 projection onto polynomial graphs). This comparison clearly demonstrates the robustness of the new moment based method with respect to noise.

In Fig. 6, subsequent examples are shown using both classifiers $\mathcal{C}^0_\epsilon$ and $\mathcal{C}^{0,1}_\epsilon$ The scanning width is $\epsilon = 4h$ in all cases. As in the previous examples, the zero moment classifier $\mathcal{C}^0_\epsilon$ is only able to detect coarser scale features. The best results are obtained by the combined zero and first moment classifier $\mathcal{C}^{0,1}_\epsilon$. In all applications listed so far, we used $\alpha = 0.1$, whereas β was set to 5 for the zero moment classifier, 100 for the first moment classifier, and 20 for the combined classifier, respectively. The computation of the classifiers took around 7 seconds for a mesh of 269,000 triangles on a Pentium 4 PC at 1.7 GHz.

4.2 Edge Preserving Surface Fairing

As an application of our local surface classification in surface processing, we consider the fairing of surfaces

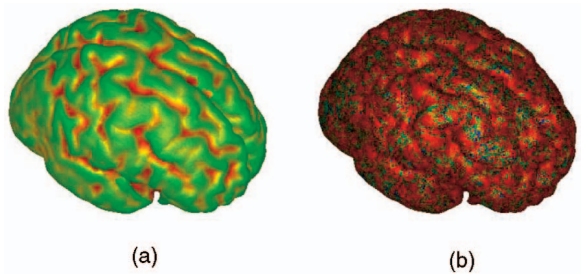


Fig. 5. Human cortex surface classified with zero moments (a) and curvature classification (b).

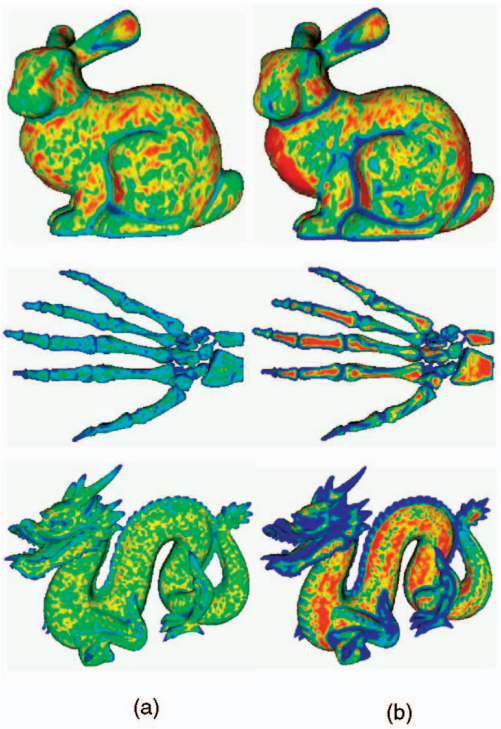


Fig. 6. Classifier applied to different surfaces. Left column: $\mathcal{C}^0_\epsilon$ and right column: $\mathcal{C}^{0,1}_\epsilon$.

based on anisotropic geometric diffusion. Our model preserves edges and in addition enables tangential smoothing along edges. Therefore—as in Section 2—we consider an anisotropic diffusion tensor $a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}$. For a suitable definition of $a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}$, we take into account the eigenvectors of the first moment $M^1_\epsilon(x)$ (see Section 3.2). Hence, we define the actual diffusion tensor on $\mathcal{T}_\xi\mathcal{M}$ in the orthogonal basis $\{w^{1,\epsilon}, w^{2,\epsilon}, n\}$, where n denotes the surface normal and $\{w^{1,\epsilon}, w^{2,\epsilon}\} \subset I\!R^3$ denote the embedded tangent vectors corresponding to the eigenvectors of the first moment. More precisely, $w^{1,\epsilon}$ corresponds to the largest eigenvalue of the first moment and $w^{2,\epsilon}$ is the orthogonal complement of $w^{1,\epsilon}$ in the tangent space. Practically, we compute the eigenvalues and eigenvectors of the three by three matrix $M^1_\epsilon(T)$ of every mesh triangle T applying Jacobi iteration.

Next, we take the eigenvector $w^{1,\epsilon}$ corresponding to the largest eigenvalue, project it on the plane of the triangle, and normalize it. Then, we compute $w^{2,\epsilon}$ as a cross product between $w^{1,\epsilon}$ and the triangle's normal. This gives a stable and simple way to determine the tangent basis $(w^{1,\epsilon}, w^{2,\epsilon})$. To illustrate this procedure, we show in Fig. 7 the eigenvector $w^{1,\epsilon}$ corresponding to the largest eigenvalue of the first moment as an arrow plot. The vector is shown only in areas where the combined zero and first moment classifier is larger than a given threshold, i.e., where features such as edges and cusps are detected. Hence, $w^{1,\epsilon}$ is aligned to the direction of the features (edges). As shown by the detailed images in Fig. 7, the computation of the feature directions is robust and reliable.

Next, we define the application of the diffusion tensor $a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}$ to a vector $z \in R^3$ by

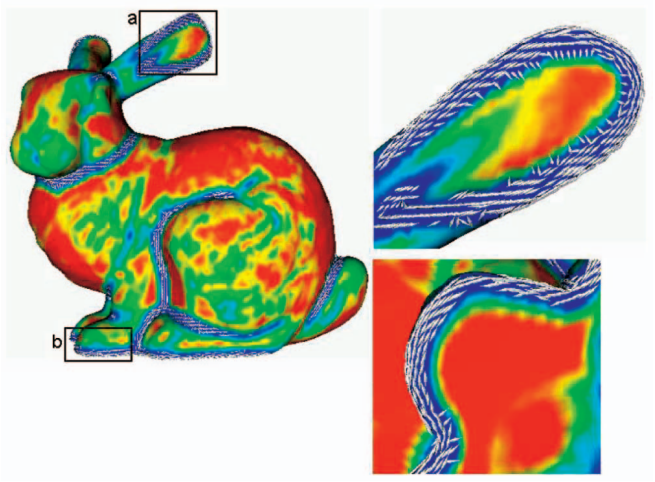


Fig. 7. In the vicinity of edges indicated by the combined classifier $\mathcal{C}_\epsilon^{0,1}$, the feature directions based on the first moment M_ϵ^1 are drawn. On the right-hand side, we zoom into the areas a and b, respectively.

$$a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}\, z := \Pi_{\mathcal{T}_\xi\mathcal{M}}\Big(G(0)(z\cdot w^{1,\epsilon})w^{1,\epsilon} + \mathcal{C}_\epsilon^{0,1}(z\cdot w^{2,\epsilon})w^{2,\epsilon} + (z\cdot n)n\Big). \tag{6}$$

Here, $\Pi_{\mathcal{T}_\xi\mathcal{M}}$ denotes the orthogonal projection onto the tangent space $\mathcal{T}_\xi\mathcal{M}$ and we identify the operator on the abstract tangent space and the endomorphism in $\mathbb{R}^3$. Furthermore, the function $G(\cdot)$ is chosen as in Section 4.1, where, in our applications, $\alpha = 0.1$ and $\beta = 20$. Now, we apply $a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}$ as diffusion tensor in the following type of parabolic evolution problem:

Given an initial compact embedded manifold $\mathcal{M}_0$ in $\mathbb{R}^3$, we compute a one parameter family of manifolds $\{\mathcal{M}(t)\}_{t\in\mathbb{R}_0^+}$ with corresponding coordinate mappings $x(t)$ which solves the system of anisotropic geometric evolution equations:

$$\partial_t x - \operatorname{div}_{\mathcal{M}(t)}(a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}\,\nabla_{\mathcal{M}(t)}x) = f \ \text{ on } \ \mathbb{R}^+\times\mathcal{M}(t), \tag{7}$$

and satisfies the initial condition

$$\mathcal{M}(0) = \mathcal{M}_0.$$

Hence, due to the anisotropy defined above, we enforce a signal enhancement in the direction of the eigenvector corresponding to the largest eigenvalue of the first moment. In the perpendicular direction on the tangent space, the amount of diffusion is determined by the combined zero and first moment classifier $\mathcal{C}_\epsilon^{0,1}$, i.e., as a function of the surface smoothness. For $f = 0$, we can rewrite the evolution problem by use of the notion of the generalized mean curvature $h_{a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}}$ (for the corresponding background, we refer to [3], [5]):

$$\partial_t x = -h_{a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}}\, n + (\operatorname{div}_{\mathcal{M}} a^\epsilon_{\mathcal{T}_\xi\mathcal{M}})(x).$$

Hence, the velocity $\partial_t x$ splits into a tangential component and a component orthogonal to the surface,

$$\Pi_{(\mathcal{T}_\xi\mathcal{M})^\perp}\,\partial_t x = -h_{a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}} n,$$
$$\Pi_{\mathcal{T}_\xi\mathcal{M}}\,\partial_t x = (\operatorname{div}_{\mathcal{M}} a^\epsilon_{\mathcal{T}_\xi\mathcal{M}})(x).$$

Fig. 8. Several time steps of the evolution problem for surface fairing using anisotropic diffusion based on the local surface classification via moments. The parameters of the function G are $\alpha = 0.1$ and $\beta = 20$.

Here, $\Pi_{(\mathcal{T}_\xi\mathcal{M})^\perp} v = (v\cdot n)n$ (with n being the surface normal), is the orthogonal projection onto the normal direction.

The tangential part $\Pi_{T_x\mathcal{M}}\,\partial_t x$ causes a tangential drift of the surface coordinates on the surface, but it does not influence the shape of the surface itself. Nevertheless, this drift property may result in degeneration of triangles in the case of discrete surfaces. To avoid this problem, we reformulate (7) by

$$\partial_t x - \Big(\operatorname{div}_{\mathcal{M}}\Big(a^\epsilon_{\mathcal{T}_\xi\mathcal{M}}, \nabla_{\mathcal{M}}x\Big)\cdot n\Big)n = 0. \tag{8}$$

In the spatially discretized form, we project the displacement of the mesh nodes onto node normals. The node normals are recomputed after each mesh smoothing step. The problem is discretized by a semi-implicit time stepping scheme (cf. the algorithm by Dziuk [10] and its generalization in [4]). Frequently, due to the robustness of our classification, it suffices to compute the classifier on the initial noisy mesh once and use it subsequently for all the deformation steps. The corresponding linear system is solved using CG-iterations. One deformation step takes about two seconds on a mesh of 269,000 triangles on a Pentium 4 PC at 1.7 GHz.

Fig. 8 shows several time steps during the edge preserving fairing of a triangular surface. The leftmost image corresponds to the original noisy surface.

5 Conclusion

In this paper, we have presented a range of local classifiers that are able to detect surface features such as edges, corners, and concave and convex smooth regions. Our main focus was to provide a stable tool that is robust even on irregular, discrete, and noisy surfaces. We have described two classifiers, based on the zero moment and a combination of the first moment eigenvalues and the zero moment. The classifiers are able to detect surface features on complex, real-world discrete surface meshes. They are simple to compute and come with a built-in scale parameter, which is the scanning width in the integration of the moments. This parameter allows us to detect only features which are above a user-specified scale. Moreover, we have proven results about the scaling behavior as well as the eigenvalues of the first moment in the smooth surfaces areas and on edges. These results serve as a quantitative basis for the use of the classifiers. For the moment calculation, we consider a block filter approach. In signal processing, it can be advantageous to use a Gaussian-type filter kernel. But, in the geometric context relevant

here, this would require a corresponding time step of mean curvature motion [4] as the geometric counterpart of Gaussian filtering. Hence, we confine ourselves to the simplest filter here, which, in particular, allows us to present a detailed qualitative and quantitative analysis.

Future work will address the use of the presented surface classifiers, especially the combined one, for devising better surface smoothing methods and for the multiscale modeling of surfaces.

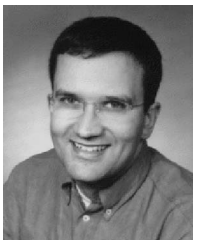
ACKNOWLEDGMENTS

The authors would like to thank Marc Tittgemeyer for valuable hints on the cortex classification.

REFERENCES

[1] L. Alvarez, F. Guichard, P.L. Lions, and J.M. Morel, "Axioms and Fundamental Equations of Image Processing," *Architectural Rational Mechanical Analysis,* vol. 123, no. 3, pp. 199-257, 1993.
[2] V. Caselles, F. Catté, T. Coll, and F. Dibos, "A Geometric Model for Active Contours in Image Processing," *Numerical Math.,* vol. 66, 1993.
[3] U. Clarenz, "Enclosure Theorems for Extremals of Elliptic Parametric Functionals," *Calculus Variations,* vol. 15, pp. 313-324, 2002.
[4] U. Clarenz, U. Diewald, and M. Rumpf, "Anisotropic Diffusion in Surface Processing," *Proc. Visualization 2000,* B. Hamann, T. Ertl, and A. Varshney, eds., pp. 397-405, 2000.
[5] U. Clarenz, G. Dziuk, and M. Rumpf, "On Generalized Mean Curvature Flow," *Geometric Analysis and Nonlinear Partial Differential Equations,* H. Karcher and S. Hildebrandt, eds., Springer, 2003.
[6] R. Deriche, "Using Canny's Criteria to Derive a Recursively Implemented Optimal Edge Detector," *Int'l J. Computer Vision,* vol. 1, pp. 167-187, 1987.
[7] M. Desbrun, M. Meyer, P. Schroeder, and A. Barr, "Implicit Fairing of Irregular Meshes Using Diffusion and Curvature Flow," *Computer Graphics (SIGGRAPH '99 Proc.),* pp. 317-324, 1999.
[8] M. Desbrun, M. Meyer, P. Schroeder, and A. Barr, "Anisotropic Feature Preserving Denoising of Height Fields and Bivariate Data," *Graphics Interface '00 Proc.,* 2000.
[9] U. Diewald, S. Morigi, and M. Rumpf, "On Geometric Evolution and Cascadic Multigrid in Subdivision," *Proc. Vision, Modeling and Visualization 2001,* T. Ertl, B. Girod, G. Greiner, H. Niemann, and H.-P. Seidel, eds., pp. 67-75, 2001.
[10] G. Dziuk, "An Algorithm for Evolutionary Surfaces," *Numerical Math.,* vol. 58, pp. 603-611, 1991.
[11] C. Gotsman, S. Gumhold, and L. Kobbelt, "Simplification and Compression of 3D Models," *Tutorials on Multiresolution in Geometric Modeling,* Springer, 2002.
[12] J. Jost, *Riemannian Geometry and Geometric Analysis.* Springer, 1998.
[13] R. Kimmel, "Intrinsic Scale Space for Images On Surfaces: The Geodesic Curvature Flow," *Graphical Models and Image Processing,* vol. 59, no. 5, pp. 365-372, 1997.
[14] L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel, "Interactive Multi-Resolution Modeling on Arbitrary Meshes," *Computer Graphics (SIGGRAPH '98 Proc.),* pp. 105-114, 1998.
[15] F.F. Leymarie and B.B. Kimia, "Computation of the Shock Scaffold for Unorganized Point Clouds in 3D," *IEEE Proc. Conf. Computer Vision and Pattern Recognition,* vol. 1, pp. 821-827, 2003.
[16] M. Meyer, M. Desbrun, P. Schröder, and A.H. Barr, "Discrete Differential-Geometry Operators for Triangulated 2-Manifolds," *Proc. VisMath Conf.,* 2002.
[17] H.P. Moreton and C.H. Séquin, "Functional Optimization for Fair Surface Design," *SIGGRAPH '92 Conf. Proc.,* pp. 167-176, 1992.
[18] S.J. Osher and J.A. Sethian, "Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," *J. Computational Physics,* vol. 79, pp. 12-49, 1988.
[19] P. Perona and J. Malik, "Scale Space and Edge Detection Using Anisotropic Diffusion," *Proc. IEEE CS Workshop Computer Vision,* 1987.
[20] U. Pinkall and K. Polthier, "Computing Discrete Minimal Surfaces and Their Conjugates," *Experimental Math.,* vol. 2, no. 1, pp. 15-35, 1993.
[21] T. Preußler and M. Rumpf, "A Level Set Method for Anisotropic Geometric Diffusion in 3D Image Processing," *SIAM J. Applied Math.,* vol. 62, no. 5, pp. 1772-1793, 2002.
[22] M. Rumpf and A. Telea, "A Continuous Skeletonization Method Based on Level Sets," *Proc. VisSym '02,* 2002.
[23] G. Taubin, "A Signal Processing Approach to Fair Surface Design," *Computer Graphics (SIGGRAPH '95 Proc.),* pp. 351-358, 1995.
[24] J. Weickert, "Foundations and Applications of Nonlinear Anisotropic Diffusion Filtering," *Z. Angew. Math. Mech.,* vol. 76, pp. 283-286, 1996.
[25] J. Weickert, *Anisotropic Diffusion in Image Processing.* Teubner, 1998.
[26] J. Wu, S. Hu, C. Tai, and J. Sun, "An Effective Feature-Preserving Mesh Simplification Scheme Based on Face Constriction," *Pacific Graphics 2001 Proc.,* pp. 12-21, 2001.

Ulrich Clarenz studied mathematics at the University of Bonn and received the Diploma in 1996 and the PhD degree in 1999. In 2000 and 2001, he worked for the Deutsche Boerse AG. Since 2001, he has held a postdoctoral research position at the University of Duisburg-Essen. His research interests are in analytical and numerical aspects of nonlinear partial differential equations and the calculus of variations.



Martin Rumpf received the Diplom degree and the PhD degree in mathematics from Bonn University in 1989 and 1992, respectively. He held a postdoctoral research position at Freiburg University. Between 1996 and 2001, he was an associate professor at Bonn University. Since 2001, he has held the chair for Numerical Mathematics and Scientific Computing at the University Duisburg-Essen. His research interests are in numerical methods for nonlinear partial differential equations, geometric evolution problems, calculus of variations, adaptive finite element methods, image and surface processing.

Alexandru Telea graduated in 1996 in computer science from the Polytechnics Institute of Bucharest, Romania, and received the PhD degree in computer science in 2000 from the Eindhoven University of Technology (TU/e), The Netherlands. Since then, he has been an assistant professor in visualization and computer graphics in the Department of Mathematics and Computer Science at TU/e. His research interests are multiscale modeling and analysis for scientific and information visualization, shape analysis and representation, and component and object-oriented software architectures.