

Principal curvatures from the integral invariant viewpoint

Helmut Pottmann* and Johannes Wallner

Vienna University of Technology

Yong-Liang Yang and Yu-Kun Lai and Shi-Min Hu

Tsinghua University, Beijing

Abstract

The extraction of curvature information for surfaces is a basic problem of Geometry Processing. Recently an integral invariant solution of this problem was presented, which is based on principal component analysis of local neighbourhoods defined by kernel balls of various sizes. It is not only robust to noise, but also adjusts to the level of detail required. In the present paper we show an asymptotic analysis of the moments of inertia and the principal directions which are used in this approach. We also address implementation and, briefly, robustness issues and applications.

Key words: integral invariants, principal curvatures, robustness

1 Introduction

The role of differential geometry in the investigation of curves and surfaces is a very important one. The computation of local invariants such as curvatures is frequently required for geometry processing applications. Also the global understanding of shapes benefits from differential entities. Examples are principal curvature lines and crest lines. References include Alliez et al. (2003), Cazals and Pouget (2005), Hildebrandt et al. (2005), Ohtake et al. (2004), and Yoshizawa et al. (2005).

Unfortunately, differentiation is sensitive to noise. Thus, the use of differential invariants requires data smoothing and de-noising prior to computation. This

* corresponding author



Fig. 1. Feature extraction on multiple scales using principal component analysis on ball neighbourhoods of different radii. Darker regions are classified as features on all scales, lighter shaded regions correspond to features extracted at only one or two scales. The images from left: Ravines – Original data — Ridges.

may be done in a global way via appropriate geometric flows, as in Bajaj and Xu (2003), Clarenz et al. (2004b), and Osher and Fedkiw (2002). Local methods, using smooth approximations of the data in an appropriate neighbourhood, are presented by Cazals and Pouget (2003), Goldfeather and Interrante (2004), Ohtake et al. (2004), Taubin (1995), and Tong and Tang (2005). In both cases, preserving features which may not be considered as noise is not an easy task and requires especially adapted algorithms. A related problem is to suppress details smaller than a given threshold size.

Classical differential geometry cannot be used directly for such important objects as triangle meshes. This led to the development of *discrete differential geometry*, a highly interesting and practically useful area which gained increasing attention over the past years and offers precise statements on the given geometry and elegant extensions of the classical theory (for an introduction with a focus on Computer Graphics, see Desbrun et al. (2005)) It is possible to extract curvatures at multiple scales using concepts of discrete differential geometry: one can use a multiresolution mesh representation, which requires a change of the underlying geometry. Some discrete operators do not need this complication and work simply on larger neighbourhoods. Handling noisy data is possible (cf. Hildebrandt and Polthier (2004)), but is neither the main intent nor strength of discrete differential geometry.

Recently we have presented an approach to the problem of robust and multiscale computation of principal curvatures via *integral invariants* obtained by integration over local neighbourhoods (Yang et al. (2006)), which is an approach initiated by Manay et al. (2004) and Clarenz et al. (2004b,a).

Our method, which is based on principal component analysis of local neighbourhoods defined via kernel balls, in a natural way leads to multiscale curvature extraction. The kernel radius serves as an approximate threshold size which distinguishes noise from features. The use of integration rather than differentiation has a smoothing effect.

We do not aim at presenting yet another method for curvature estimation, but rather investigate a new tool which takes a more *global view*. It computes integrated curvature-like quantities over the chosen neighbourhoods. Only for very small kernel radii the invariants computed in this way are closely related to curvatures. Their behavior on larger scales is different, but consistent and useful for applications.

1.1 Prior work on integral invariants, principal curves and feature extraction

While principal component analysis has been used to obtain shape characteristics for centuries (see e.g. Taubin and Cooper (1992) for applications in Computer Vision), local integral invariants are a rather new topic in geometric computing. Manay et al. (2004) investigate integral invariants for curves in the plane and show their superior performance on noisy data, especially for the reliable retrieval of shapes from geometric databases. A special case of an integral invariant, defined for 2D curves or 3D surfaces, has been used by Connolly (1986) for molecular shape analysis. In the case of a surface Φ , defined as boundary of a domain D , Connolly considers the surface area $A_s^r(\mathbf{p})$ of the spherical patch neighbourhood $N_s^r := D \cap S_r(\mathbf{p})$ (see Fig. 2); here \mathbf{p} is a point of Φ and $S_r(\mathbf{p})$ denotes the sphere of radius r centered at \mathbf{p} . Equation (1) which describes the relation of the Connolly function to mean curvature H of Φ in a precise way, has been derived by Cazals et al. (2003). Gelfand et al. (2005) use the volume $V_b^r(\mathbf{p})$ of the ball neighbourhood $N_b^r(\mathbf{p}) := D \cap B_r(\mathbf{p})$ to obtain a geometry descriptor useful for finding correspondences in matching problems. Both A_s^r and V_b^r turn out to be related to mean curvature (cf. Hulin and Troyanov (2003); Cazals et al. (2003); Gelfand et al. (2005)):

$$V_b^r = \frac{2\pi}{3}r^3 - \frac{\pi H}{4}r^4 + O(r^5), \quad A_s^r = 2\pi r^2 - \pi H r^3 + O(r^4). \quad (1)$$

Most closely related to the present work are the papers of Clarenz et al. (2004a,b), where integral invariants are employed for feature detection. The authors consider the surface patch neighbourhood $N_p^r(\mathbf{p}) = \Phi \cap B_r(\mathbf{p})$ and

perform PCA on this patch. They show that the distance between \mathbf{p} and the barycenter of N_p^r is related to mean curvature, and discuss some scaling properties of the eigenvalues coming up in PCA. The eigenvalues resulting from PCA on N_p^r have also been used by Pauly et al. (2003) for multi-scale feature extraction on point-sampled surfaces.

Globally defined features computable from principal directions are the principal curvature lines. These curves proved to be useful for the detection of special shapes, and also for such diverse applications as non-isotropic remeshing (by Alliez et al. (2003)), or line-art rendering of smooth surfaces (see Hertzmann and Zorin (2000)). Global features whose definition relies on even higher order differential invariants like derivatives of principal curvatures also carry important shape information: For instance, the curves known as feature lines and crest lines received a lot of attention in geometry processing (see Cazals and Pouget (2005), Clarenz et al. (2004b), Hildebrandt et al. (2005), Ohtake et al. (2004), Pauly et al. (2003), and Yoshizawa et al. (2005)).

1.2 Contributions and overview

The main contributions of our paper are:

- Section 2 presents a thorough study of principal component analysis of four types of neighbourhoods (see Fig. 2) which can be defined by means of a kernel ball or kernel sphere. The precise relation of the quantities obtained by PCA to the curvatures of the underlying surface is obtained by an asymptotic analysis as the kernel radius tends to zero.
- Integral invariants lead to the definition of principal curvatures at a given resolution level r which are consistent with the classical theory ($r \rightarrow 0$). This is discussed in Sec. 3. The multiscale behaviour of these modified principal curvatures and comparison of this method with others is the topic of Yang et al. (2006), as is the computation of principal curves at a given scale and multi-scale feature extraction. We only briefly discuss these applications here.
- In Sec. 4 we discuss implementation issues. We use FFT and hierarchical refinement of sphere triangulations.

2 Principal component analysis of local neighbourhoods

Principal component analysis of a point set A requires the computation of its volume $V(A) = \int_A d\mathbf{x}$, its barycenter $\mathbf{s}(A) = \frac{1}{V(A)} \int_A \mathbf{x} d\mathbf{x}$, and its covariance

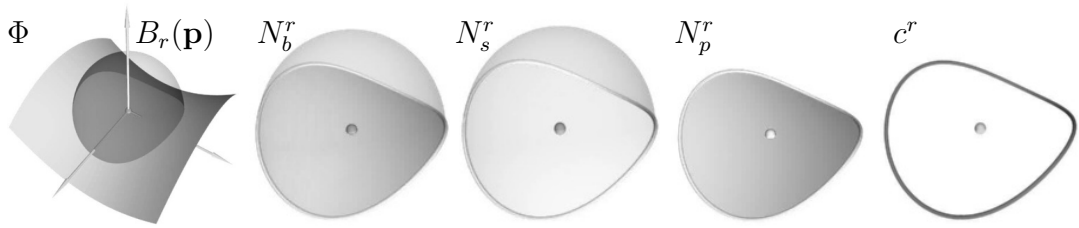


Fig. 2. From left to right: Kernel ball $B_r(\mathbf{p})$ and surface Φ ; ball neighbourhood N_b^r ; sphere neighbourhood N_s^r ; surface patch neighbourhood N_p^r ; spherical intersection curve c^r .

matrix

$$J(A) = \int_A (\mathbf{x} - \mathbf{s}) \cdot (\mathbf{x} - \mathbf{s})^T d\mathbf{x} = \int_A \mathbf{x}\mathbf{x}^T d\mathbf{x} - V(A)\mathbf{s}\mathbf{s}^T. \quad (2)$$

We use column vector notation, so $\mathbf{x}\mathbf{x}^T$ is a 3 by 3 matrix of rank 1. If A is contained in a smooth (rectifiable) surface, then $d\mathbf{x}$ means the surface area element. If A is contained in a smooth (rectifiable) curve, then $d\mathbf{x}$ is the arc length element.

In our case A is a local neighbourhood of a point defined by a kernel of radius r , and we are interested in the first terms of the Taylor expansion with respect to r of $V(A)$, $\mathbf{s}(A)$, $J(A)$, and the eigenvalues and eigenvectors of $J(A)$.

2.1 The principal frame and the simplification of volume integrals

For theoretical investigations it is convenient to work in a coordinate system associated with a point on the surface. We repeat well known results here – they can be found e.g. in do Carmo (1976). For any given point of a C^2 surface there is the so-called principal frame, with respect to which the surface is realized as the graph of the function

$$z(x, y) = \frac{1}{2}(\kappa_1 x^2 + \kappa_2 y^2) + O(\rho^3) \quad (\rho^2 = x^2 + y^2). \quad (3)$$

Here κ_1 and κ_2 denote the principal curvatures of the surface in the point of interest, which is located at the origin of the principal frame. We use the notation $\rho = \sqrt{x^2 + y^2}$ also later in the text. Further, we let $\mathbf{x} = (x, y, z)$. Consider now the general volume integral

$$I(f) = \int_A f(\mathbf{x})d\mathbf{x}, \text{ where } A = \{\mathbf{x} \mid \mathbf{x}^T \mathbf{x} \leq r, z \leq z(x, y)\}. \quad (4)$$

The next theorem shows how to approximately compute it. We use the notation B_r^+ for the half ball $x^2 + y^2 + z^2 = \rho^2 + z^2 \leq r^2, z \geq 0$.

Theorem 1 *If the function f is of magnitude $O(\rho^k z^l)$, then its integral $I(f)$ over the domain $x^2 + y^2 + z^2 \leq r$, $z \leq z(x, y)$ can be expressed as*

$$I(f) = \tilde{I}(f) + O(r^{k+2l+5}), \text{ where}$$

$$\tilde{I}(f) = \int_{B_r^+} f(\mathbf{x}) d\mathbf{x} - \int_{x^2+y^2 \leq r^2} \left(\int_{z=0}^{\frac{1}{2}(\kappa_1 x^2 + \kappa_2 y^2)} f(\mathbf{x}) dz \right) dx dy. \quad (5)$$

Proof. First, the integral $I(f)$ is approximated by

$$I(f) \approx \int_{B_r^+} f(\mathbf{x}) d\mathbf{x} - \int_{x^2+y^2 \leq r^2} \left(\int_{z=0}^{z=z(x,y)} f(\mathbf{x}) dz \right) dx dy. \quad (6)$$

The error we make is the volume integral of f over that part \tilde{A} of the original domain which lies outside the sphere $x^2 + y^2 + z^2 = r^2$, but still inside the cylinder $x^2 + y^2 = r^2$. The integral $\int_{\tilde{A}} f$ is bounded by $\text{Vol}(\tilde{A}) \cdot \max_{\tilde{A}} |f|$. The set \tilde{A} has circumference $\approx 2\pi r$ and in view of (3) its extent in the z direction is $O(r^2)$. It has been shown by Hulin and Troyanov (2003) that the radial thickness of \tilde{A} is of magnitude $O(r^3)$, so $\text{Vol}(\tilde{A}) \sim r \cdot r^2 \cdot r^3 = r^6$. Equation (3) implies that within the domain \tilde{A} , z is of magnitude r^2 . With $f \sim \rho^k \cdot z^l$, we now have an upper bound on the integral which is of magnitude $r^6 \cdot r^k (r^2)^l$.

We further approximate the integral by neglecting the term $O(r^3)$ in the expression $z(x, y)$, i.e., by replacing the surface by its osculating paraboloid. Again, we assume that $f = O(\rho^k z^l)$. The error is given by the integral of f over the layer which is bounded by the graph of the function $z(x, y)$ given by (3) and the graph of the same function without the $O(r^3)$ term. I.e., the layer has a thickness of magnitude $O(r^3)$. As before, $z(x, y)$ is of magnitude $O(\rho^2)$. The integral consequently has the form $\int_{\phi=0}^{2\pi} \int_{\rho=0}^r O(\rho^3) \cdot O(\rho^k (\rho^2)^l) \cdot \rho d\rho d\phi = O(r^{k+2l+5})$.

To sum up, the two modifications of the integral $I(\tilde{f})$ lead to errors of magnitude r^{2k+l+6} and r^{2k+l+5} , respectively, so we have shown that $I(f) - \tilde{I}(f) = O(r^{2k+l+5})$. \square

2.2 Principal component analysis of the ball neighbourhood

We consider the surface Φ as the boundary of the domain D and perform PCA on the ball neighbourhood $N_b^r := B_r(\mathbf{p}) \cap D$. The volume V^r of N_b^r is the integral of the constant function 1, which is of magnitude $O(\rho^0 z^0)$. We use (5) to approximate the volume and get precisely the known result of Equation (1), with $H = (\kappa_1 + \kappa_2)/2$. For the barycenter \mathbf{s}_b , we have to compute $\int x d\mathbf{x}$, $\int y d\mathbf{x}$, and $\int z d\mathbf{x}$. The functions x, y are of magnitude $O(\rho^1 z^0)$, whereas the

z coordinate function is of magnitude $O(\rho^0 z^1)$. Consequently,

$$\mathbf{s}_b = \frac{1}{V_b^r} (\tilde{I}(\mathbf{x}) + O(r^6)) = \left(\frac{2\pi}{3} r^3 - \frac{\pi H}{4} r^4 + O(r^5) \right)^{-1} \left(\begin{bmatrix} 0 \\ 0 \\ \pi r^4/4 \end{bmatrix} + O(r^6) \right)$$

(in this computation, many terms evaluate to zero because of symmetry). Division of power series yields

$$\mathbf{s}_b = \left[0, \quad 0, \quad \frac{3}{8}r + \frac{9H}{64}r^2 \right]^T + O(r^3). \quad (7)$$

Now we consider the covariance matrix. The following table shows the order $O(\rho^k z^l)$ of magnitude of the functions we are going to integrate:

f	x^2	y^2	z^2	xy	xz	yz
(k, l)	$(2, 0)$	$(2, 0)$	$(0, 2)$	$(2, 0)$	$(1, 1)$	$(1, 1)$

We approximate the integrals over those functions according to (5): In any case the error is at most $O(r^7)$, while $\tilde{I}(xy) = \tilde{I}(xz) = \tilde{I}(yz) = 0$ because of symmetry. Thus, the covariance matrix reads

$$\begin{aligned} J^r &= \int \mathbf{x}\mathbf{x}^T d\mathbf{x} - V_b^r \mathbf{s}_b \mathbf{s}_b^T = \text{diag} \left(\tilde{I}(x^2), \tilde{I}(y^2), \tilde{I}(z^2) \right) + O(r^7) \\ &\quad - \text{diag} \left(0, 0, \frac{\pi}{4} r^4 \left(\frac{3}{8}r + \frac{9H}{64}r^2 \right) \right) + O(r^9). \end{aligned} \quad (8)$$

Note that the product $V_b^r \mathbf{s}_b$ is already known from above. This leads to the following result:

Theorem 2 *The principal moments of inertia of the ball neighbourhood (i.e., the eigenvalues of the covariance matrix of $N_b^r(\mathbf{p})$) have the Taylor expansion*

$$\begin{aligned} M_{bi}^r &= \tilde{M}_{bi}^r + O(r^7) \quad (i = 1, 2, 3), \quad \text{where} \\ \tilde{M}_{b1}^r &= \frac{2\pi}{15} r^5 - \frac{\pi}{48} (3\kappa_1 + \kappa_2) r^6, \quad \tilde{M}_{b2}^r = \frac{2\pi}{15} r^5 - \frac{\pi}{48} (\kappa_1 + 3\kappa_2) r^6, \\ \tilde{M}_{b3}^r &= \frac{19\pi}{480} r^5 - \frac{9\pi}{512} (\kappa_1 + \kappa_2) r^6. \end{aligned} \quad (9)$$

$$(10)$$

Here κ_1 and κ_2 are the principal curvatures at the point \mathbf{p} .

Proof. For symmetric matrices J^r and \tilde{J}^r in general, the difference of eigenvalues is bounded by $\|J^r - \tilde{J}^r\|$, where $\|\cdot\|$ is a matrix norm. When computed with respect to the principal frame associated with the point \mathbf{p} , the covariance matrix J^r differs from a diagonal matrix \tilde{J}^r only by an error of magnitude $O(r^7)$, as shown by (8). It follows that the principal moments M_{bi} coincide with the

diagonal elements of \tilde{J}^r , up to $O(r^7)$. The precise values of these diagonal elements are found by computing the integrals $\tilde{I}(x^2), \dots$ according to (5), which is elementary but tedious. From (8) we see that $M_{b1} = \tilde{I}(x^2) + O(r^7)$ and $M_{b2} = \tilde{I}(y^2) + O(r^7)$. Interestingly

$$\tilde{I}(z^2) = \frac{2}{15}\pi r^5 \quad (11)$$

does not contain any curvature information. In combination with the other terms in (8), we get M_{b3} . \square

Theorem 3 *We consider the situation of Theorem 2. As $r \rightarrow 0$, the eigenvectors \mathbf{e}_1^r and \mathbf{e}_2^r of the covariance matrix which correspond to the moments M_{b1}^r and M_{b2}^r converge to the principal directions $\mathbf{e}_1, \mathbf{e}_2$ associated with κ_1 and κ_2 , provided $\kappa_1 \neq \kappa_2$. The eigenvector \mathbf{e}_3^r converges towards the surface normal vector \mathbf{n} . For \mathbf{e}_1^r and \mathbf{e}_2^r , this convergence is linear, whereas for \mathbf{e}_3^r it is quadratic:*

$$\angle(\mathbf{e}_1^r, \mathbf{e}_1), \angle(\mathbf{e}_2^r, \mathbf{e}_2) \sim \frac{r}{\kappa_1 - \kappa_2}, \quad \angle(\mathbf{e}_3^r, \mathbf{n}) \sim r^2. \quad (12)$$

Proof. The statement about convergence of eigenvectors follows from the ‘‘sine theta’’ theorems for perturbation of eigenvectors of Hermitean matrices due to Davis and Kahan (1970). If λ_i is an eigenvalue of the matrix J , and ε is the minimum distance from the remaining eigenvalues, then a perturbation of size h causes a change in the eigenvector \mathbf{e}_i of magnitude h/ε , provided $h \ll \varepsilon$. We apply this result to $J = \text{diag}(\tilde{M}_{b1}, \tilde{M}_{b2}, \tilde{M}_{b3})$, with \tilde{M}_{bi} from Equations (9) and (10). Then $h = O(r^7)$. In the case $i = 1, 2$, we have $\varepsilon \sim r^6(\kappa_1 - \kappa_2)$, and in the case $i = 3$, we have $\varepsilon \sim r^5$. \square

The moments M_{b1}^r, M_{b2}^r do not feature the principal curvatures in their dominant terms, but their difference does:

$$M_{b2}^r - M_{b1}^r = \frac{\pi}{24}(\kappa_1 - \kappa_2)r^6 + O(r^7). \quad (13)$$

2.3 Principal component analysis of the sphere neighbourhood

In this section we consider principal component analysis of the sphere neighbourhood $N_s^r = S_r^2(\mathbf{p}) \cap D$ in a way analogous to the previous section which dealt with the ball neighbourhood.

Theorem 4 *The barycenter \mathbf{s}_s of the sphere neighbourhood $N_s^r(\mathbf{p})$ is expressed in the principal frame as*

$$\mathbf{s}_s = \left[0, 0, \frac{1}{2}r + \frac{\kappa_1 + \kappa_2}{8}r^2 + \right]^T + O(r^3). \quad (14)$$

The principal moments of inertia of the sphere neighbourhood (i.e., eigenvalues of the covariance matrix) satisfy $M_{s_i}^r = \widetilde{M}_{s_i}^r + O(r^6)$, where

$$\begin{aligned}\widetilde{M}_{s_1}^r &= \frac{2\pi}{3}r^4 - \frac{\pi}{8}(3\kappa_1 + \kappa_2)r^5, & \widetilde{M}_{s_2}^r &= \frac{2\pi}{3}r^4 - \frac{\pi}{8}(\kappa_1 + 3\kappa_2)r^5, \\ \widetilde{M}_{s_3}^r &= \frac{\pi}{6}r^4 - \frac{\pi}{8}(\kappa_1 + \kappa_2)r^5.\end{aligned}\quad (15)$$

Proof. Any integral $I(f)$ over the ball neighbourhood can be written as an iterated integral:

$$I^r(f) = \int_{D \cap B_r(\mathbf{p})} f(\mathbf{x}) d\mathbf{x}, \quad I^\rho(f) = \int_{S_\rho(\mathbf{p})} f(\mathbf{x}) d\mathbf{x} \implies I^r(f) = \int_{\rho=0}^r I^\rho(f) d\rho.$$

This implies the differential relation

$$I^r(f) = \frac{d}{dr} I^r(f). \quad (17)$$

Thus we can compute the surface area A_s^r , the integral $I'(\mathbf{x})$ used for the barycenter, and the matrix $I'(\mathbf{x}\mathbf{x}^T)$ used for the covariance matrix, simply by differentiating V_b^r , $I(\mathbf{x})$, and $I(\mathbf{x}\mathbf{x}^T)$, respectively. We get the known expression for the surface area A_s^r given by (1), and further

$$A_s \mathbf{s}_s = \begin{bmatrix} 0 \\ 0 \\ \pi r^3 \end{bmatrix} + O(r^5), \quad \int_{S_r(\mathbf{p})} \mathbf{x}\mathbf{x}^T d\mathbf{x} = \begin{bmatrix} d\widetilde{M}_{b_1}/dr & 0 & 0 \\ 0 & d\widetilde{M}_{b_2}/dr & 0 \\ 0 & 0 & \frac{2}{3}\pi r^4 \end{bmatrix} + O(r^6).$$

For the lower right corner of the matrix $I'(\mathbf{x}\mathbf{x}^T)$, we have differentiated (11). By dividing $I'(\mathbf{x})$ by the Taylor series of the surface area A_s , we get (14). Now that the barycenter is known, we are able to compute the covariance matrix: $J^r = I'(\mathbf{x}\mathbf{x}^T) - A_s \mathbf{s}_s \mathbf{s}_s^T = \text{diag}(\widetilde{M}_{s_1}, \widetilde{M}_{s_2}, \widetilde{M}_{s_3}) + O(r^6)$. This completes the proof. \square

2.4 Principal component analysis of the spherical intersection curve

We now consider the intersection curve $c^r(\mathbf{p}) = S^r(\mathbf{p}) \cap \Phi$, which is defined as the intersection of a kernel sphere of radius r with the smooth surface Φ . Principal component analysis of the spherical intersection curve is included here for the sake of completeness. It is not as robust against perturbations as PCA of the ball or sphere neighbourhoods $N_b^r(\mathbf{p})$ and $N_s^r(\mathbf{p})$. The analysis consists mainly of computations not all of which are repeated here. We first give a cylinder coordinate representation $(\phi, \rho(\phi), z(\phi))$ of the intersection curve, where $x = \rho \cos \phi$ and $y = \rho \sin \phi$.

The intersection of the plane $x : y = \cos \phi : \sin \phi$ with the surface of (3) results in the curve $z(\rho, \phi) = \frac{1}{2}\kappa_n(\phi)^2\rho^2 + O(\rho^3)$, where the expression

$$\kappa_n(\phi) = \kappa_1 \cos^2 \phi + \kappa_2 \sin^2 \phi \quad (18)$$

has an interpretation as the normal curvature of the surface Φ associated with the direction $x : y = \cos \phi : \sin \phi$ (cf. do Carmo (1976)). Intersection of that curve $z = z(\rho, \phi)$, $\phi = \text{const}$, with the sphere $\rho^2 + z^2 = r^2$ requires some elementary computations and leads to the following parametrization of the intersection curve:

$$c^r : \quad \rho(\phi) = r - \frac{1}{8}\kappa_n(\phi)^2 r^3 + O(r^4), \quad z(\phi) = \kappa_n(\phi)^2 r^2 + O(r^3). \quad (19)$$

The arc length differential in cylinder coordinates reads $(ds/d\phi)^2 = \rho^2 + \rho_\phi^2 + z_\phi^2$. Here the subscript ϕ indicates differentiation. We get $s_\phi^2 = r^2 + \frac{1}{4}(\kappa_{n,\phi}^2 - \kappa_n^2) + O(r^5)$. Taking the square root via the binomial series yields $s_\phi = r + \frac{1}{8}(\kappa_{n,\phi}^2 - \kappa_n^2)r^3 + O(r^4)$. When we expand this expression and integrate, we can compute the arc length of c^r :

$$L_c^r = \oint ds = \int_{\phi=0}^{2\pi} \frac{ds}{d\phi} d\phi = 2\pi r + \frac{\pi}{32}(\kappa_1^2 - 10\kappa_1\kappa_2 + \kappa_2^2)r^3 + O(r^4). \quad (20)$$

Barycenter and principal components are subsequently found by integration. We have $\mathbf{s}_c = \frac{1}{L_c} \int \mathbf{x}(\phi)(ds/d\phi)d\phi$, and the covariance matrix is given by $J(c) = \frac{1}{L_c} \int \mathbf{x}(\phi)\mathbf{x}(\phi)^T(ds/d\phi)d\phi$. After the substitution $x(\phi) = \rho(\phi) \cos \phi$ and $y(\phi) = \rho(\phi) \sin \phi$, these integrations are elementary and will not be carried out in detail. The results are similar to the ball and sphere neighbourhood case:

Theorem 5 *The barycenter \mathbf{s}_c of the spherical intersection curve $c^r(\mathbf{p})$ is expressed in the principal frame as $\mathbf{s}_s = [0, 0, \frac{\kappa_1 + \kappa_2}{4}r^2]^T + O(r^3)$. The principal moments of inertia of the curve $c^r(\mathbf{p})$ satisfy*

$$\begin{aligned} M_{c1}^r &= \pi r^3 + \frac{\pi}{64}((\kappa_1 - \kappa_2)^2 - 12\kappa_2(\kappa_1 + \kappa_2))r^5 + O(r^6), \\ M_{c3}^r &= \frac{\pi}{16}(\kappa_1 - \kappa_2)^2 r^5 + O(r^6). \end{aligned}$$

The formula for M_{c2} is the one for M_{c1} with indices 1 and 2 exchanged.

2.5 PCA of the surface patch neighbourhood

We now discuss principal component analysis of the surface patch neighbourhood N_p^r , thereby providing more precise estimates than those given by Clarenz et al. (2004b). While the surface patch neighbourhood is of equal geometric interest as the ball and sphere neighbourhoods, it has been shown by

the numerical experiments of Yang et al. (2006) that it is not as robust against noise. This is only to be expected, since a variation of the surface causes a variation in the surface area which can be bounded only if we know bounds on the derivatives of the surface.

Like in the case of the spherical intersection curve c^r , the mathematics consists almost entirely of computations, which are not repeated here. We again refer to the cylinder coordinate representation $x = \rho \cos \phi$, $y = \rho \sin \phi$, $z = z(\rho \cos \phi, \rho \sin \phi)$, of the surface (3). The surface area element is given by $dA = (1 + (dz/dx)^2 + (dz/dy)^2)^{1/2} dx dy = (\rho + \frac{1}{2}\rho^3(\kappa_1^2 \cos^2 \phi + \kappa_2^2 \sin^2 \phi) + O(\rho^5)) d\rho d\phi$. The surface area is computed as the integral $\int_{\phi=0}^{2\pi} \int_{\rho=0}^{\rho(\phi)} dA$, where $\rho(\phi)$ is taken from (19). The result is

$$A_p^r = \pi r^2 + \frac{\pi}{32}(\kappa_1 - \kappa_2)^2 r^4 + O(r^5). \quad (21)$$

The remaining computations of barycenter and principal components are similar to the curve case:

Theorem 6 *The barycenter \mathbf{s}_p of the spherical path neighbourhood $N_p^r(\mathbf{p}) = B_r(\mathbf{p}) \cap \Phi$ is expressed in the principal frame as $\mathbf{s}_p = [0, 0, \frac{\kappa_1 + \kappa_2}{8} r^2]^T + O(r^3)$. The principal moments of inertia of the patch neighbourhood (i.e., eigenvalues of the covariance matrix) satisfy*

$$\begin{aligned} M_{p1}^r &= \frac{\pi}{4} r^4 + \frac{\pi}{192}(\kappa_2^2 - 3\kappa_1^2 - 6\kappa_1\kappa_2)r^6 + O(r^7), \\ M_{p3}^r &= \frac{\pi}{192}(3\kappa_1^3 + 3\kappa_2^2 - 2\kappa_1\kappa_2)r^6 + O(r^7). \end{aligned}$$

The formula for M_{p2} is the one for M_{p1} with indices 1 and 2 exchanged.

3 Geometry descriptors from integral invariants

For many geometry processing tasks it is important to compute numerical geometry descriptors which act as a low pass filter, i.e., ignore details which are smaller than a certain threshold size. One way to define such descriptors is described below.

3.1 Principal curvatures at a given scale

The relationship between principal moments of inertia with the principal curvatures leads to the definition of *principal curvatures at scale r* , where r is the kernel ball radius. This is done by ignoring the $O(r^7)$ -terms in (9) and solving

for κ_1, κ_2 . We get

$$\kappa_{b1}^r := \frac{6}{\pi r^6}(M_{b2}^r - 3M_{b1}^r) + \frac{8}{5r}, \quad \kappa_{b2}^r := \frac{6}{\pi r^6}(M_{b1}^r - 3M_{b2}^r) + \frac{8}{5r}. \quad (22)$$

Theorem 4 likewise makes it possible to define principal curvatures κ_{s1} and κ_{s2} , using the moments M_{s1} and M_{s2} . And of course also the moments computed from the curve c^r or the surface patch N_p^r can be used as well. The arithmetic mean $(\kappa_{b1}^r + \kappa_{b2}^r)/2$ is a possible definition of the mean curvature at scale r , but also other formulae (e.g., the ones involving barycenters, volumes, or areas) can be used to define a mean curvature at scale r . Any of these different definitions of principal curvatures or mean curvature at scale r yields a family of geometry descriptors which approximates curvatures as the kernel radius tends to zero.

An important property of principal curvatures at a given scale is that features of the surface under investigation which are smaller than the kernel radius are considered as noise and are in general smoothed away. When we use principal curvatures at various different scales r_1, r_2, \dots in order to recognize features on a surface, it may happen that some features are detected only for one r_i , whereas others (the *persistent* ones) are detected for several radii r_j . This helps to distinguish between unimportant and important features of given data. Examples are given by Yang et al. (2006). One instance of such a computation is shown by Fig. 1.

3.2 Principal curves on multiple scales

PCA on local neighbourhoods provides a way to obtain principal directions $\mathbf{e}_1^r, \mathbf{e}_2^r$ on different scales. These directions need not be tangential to the given surface at \mathbf{p} , since they are computed in a global way from the chosen neighbourhood. In fact, they should not follow details which are small compared to r . In order to obtain vector fields which are tangent to the given surface, \mathbf{e}_1^r and \mathbf{e}_2^r are projected onto the surface. Note that the projected directions need not be orthogonal anymore. However, this is not important and actually an advantage when we now integrate these two vector fields to obtain *principal curves at the chosen scale r* . By doing the projection not orthogonally to the surface, but via the eigenvector \mathbf{e}_3 , principal curves become less sensitive to local surface deviations.

Applications of principal curves at a given resolution r include remeshing for the purpose of constructing a quad-dominant mesh with planar faces approximating a given surface, as presented by Liu et al. (2006).

Due to the robustness of these principal curves, it seems feasible to use such curves for detecting special shapes such as pipe surfaces, rotational surfaces,

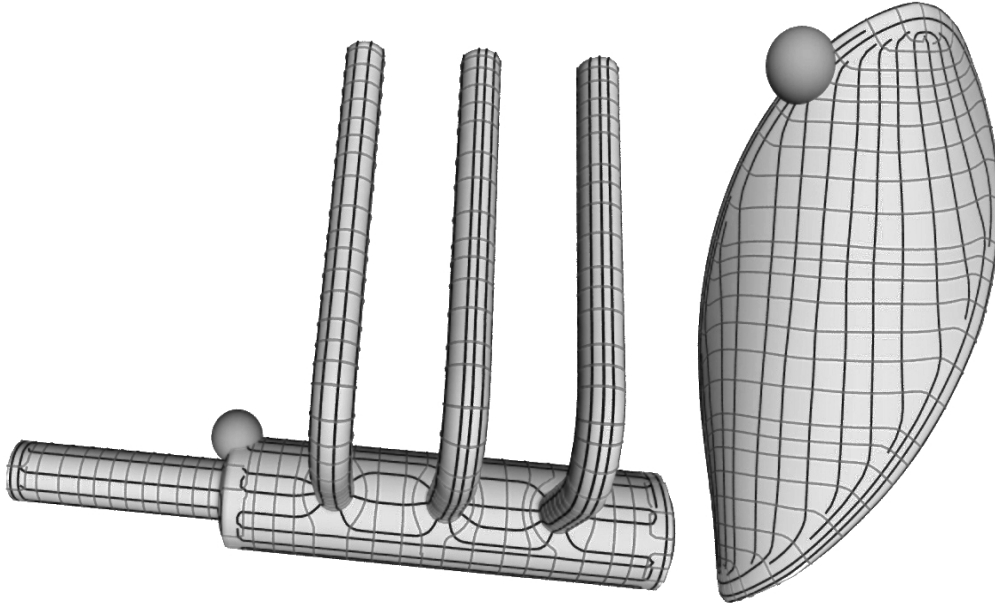


Fig. 3. Surface interrogation with principal curves at a certain scale r , computed for models which contain developable surfaces. Left: pipe surfaces which contain cylindrical parts. Right: D-form. One kernel ball is depicted in each case.

canal surfaces and developable surfaces (see Fig. 3). Non-planar developable surfaces can be characterized by the fact that exactly one principal curvature vanishes; they have one family of straight principal curvature lines. Recognizing and especially reconstructing developable surfaces from measurement data is not an easy task (see the discussion by Peternell (2004)). Classical principal curvature lines can hardly be employed for computations where robustness is essential. However it turns out that principal curvatures at a larger scale r are useful for such purposes. This is illustrated by Fig. 3: Apart from smoothing effects near sharp edges, we can nicely observe one family of nearly straight principal curves. They are good candidates for rulings in an approximating surface. The ‘D-form’ model of Fig. 3, right, is a triangle mesh generated from a laser scan of a paper model. For more facts on D-forms, the interested reader is referred to pp. 317, 401, and 418 of the monograph by Pottmann and Wallner (2001).

3.3 Comparison with other methods, robustness, and multiscale behaviour

Yang et al. (2006) compare the principal curvatures obtained via the ball and sphere neighbourhood with other methods:

- Principal components of the patch neighbourhood according to Clarenz et al. (2004b) and Pauly et al. (2003)
- The method of normal cycles of Cohen-Steiner and Morvan (2003);
- A fitting method (osculating jets by Cazals and Pouget (2003)). It turns

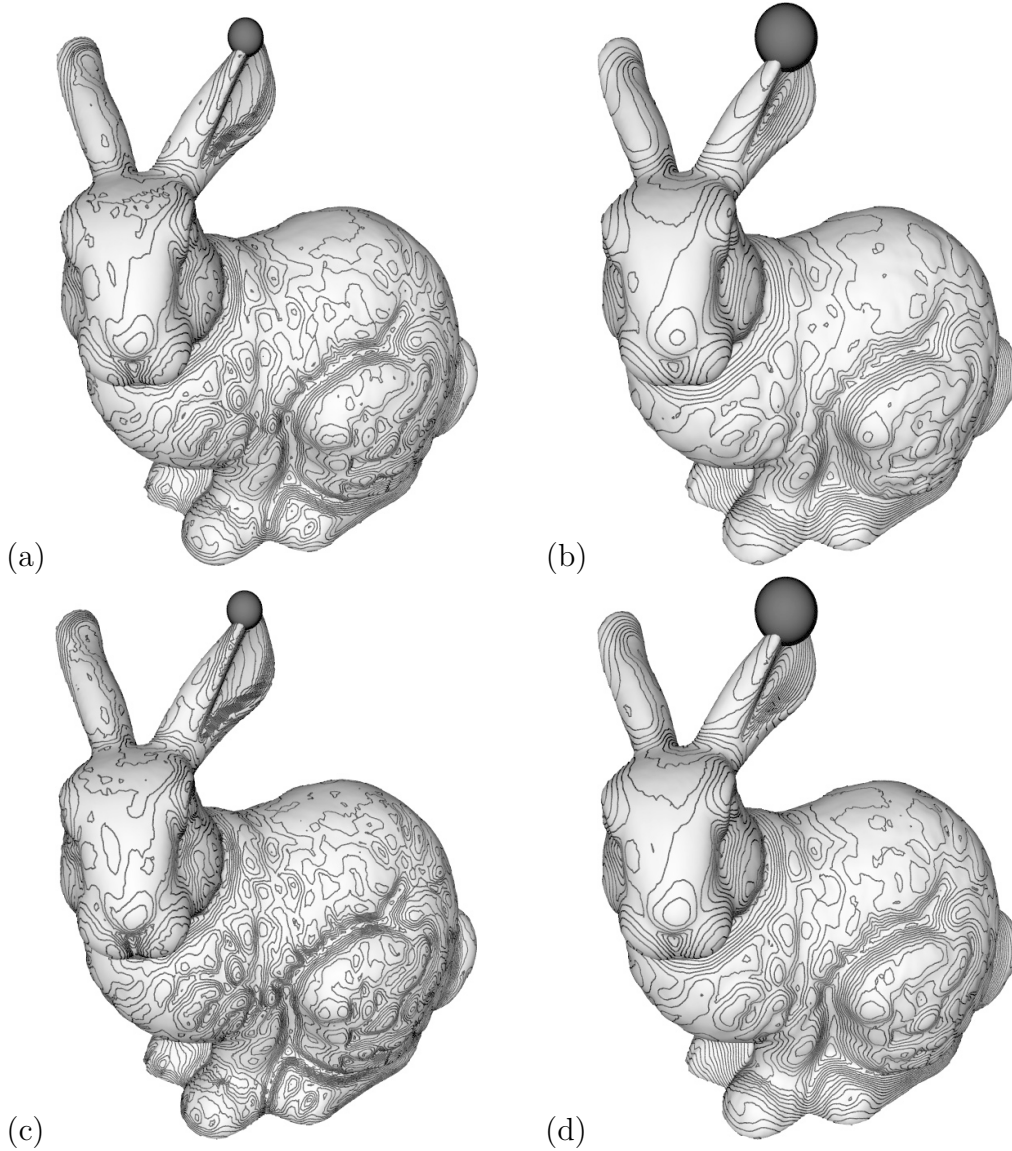


Fig. 4. Level sets of curvatures computed for the Stanford bunny dataset. (a) Sphere neighbourhood, small kernel radius; (b) Sphere neighbourhood, large kernel radius; (c) Ball neighbourhood, small kernel radius; (d) Ball neighbourhood, large kernel radius.

out that PCA of the patch neighbourhood is quite sensitive to noise, normal cycles less so. PCA of the ball and sphere neighbourhoods compare favourably with these three methods. The multiscale behaviour of these different methods is good for low noise levels, while local fitting methods apparently have defects at coarse scales.

The multiscale behaviour of integral invariants defined via the ball and sphere neighbourhoods is illustrated in Fig. 4, which shows curvatures computed in various ways.

4 Implementation

4.1 Ball neighbourhood computations via FFT

Integral invariants defined via the ball neighbourhood can easily be computed by convolution, which means that FFT can be employed. In more detail, this is done as follows: In order to do PCA on the ball neighbourhood $N^r(\mathbf{p}) = B^r(\mathbf{p}) \cap D$, we compute the integrals $I(1)$, $I(\mathbf{x})$, $I(\mathbf{x}\mathbf{x}^T)$. It does not matter which coordinate system we use for computing the moments of inertia of $N^r(\mathbf{p})$, as long as we consistently use the same coordinate system while we work with the point \mathbf{p} . We translate the world coordinate system such that the origin varies, and we indicate the origin used for computation by a subscript, i.e., we will write $I_{\mathbf{p}}(\mathbf{x})$ and so on.

We use the notation $1_D(\mathbf{x})$ for the indicator function of the domain D , which is defined by $1_D(\mathbf{x}) = 1$ if $\mathbf{x} \in D$ and $1_D(\mathbf{x}) = 0$ otherwise. Likewise, we define $1_B(\mathbf{x})$ to be the indicator function of the ball with center \mathbf{o} and radius r . Apparently, the definitions of $I(1)$, $I(\mathbf{x})$, and $I(\mathbf{x}\mathbf{x}^T)$ can be rewritten as

$$\begin{aligned} I_{\mathbf{p}}(1) &= \int_{\mathbb{R}^3} 1_D(\mathbf{x})1_B(\mathbf{x})d\mathbf{x}, \\ I_{\mathbf{p}}(\mathbf{x}) &= \int_{\mathbb{R}^3} 1_D(\mathbf{x})(-(\mathbf{p} - \mathbf{x}))1_B(\mathbf{p} - \mathbf{x})d\mathbf{x}, \\ I_{\mathbf{p}}(\mathbf{x}\mathbf{x}^T) &= \int_{\mathbb{R}^3} 1_D(\mathbf{x})(\mathbf{p} - \mathbf{x})(\mathbf{p} - \mathbf{x})^T 1_B(\mathbf{p} - \mathbf{x}). \end{aligned}$$

Note that we write $\mathbf{p} - \mathbf{x}$ instead of $\mathbf{x} - \mathbf{p}$, which is justified by the symmetries present. We use convolution notation, i.e., $(f * g)(\mathbf{p}) = \int f(\mathbf{x})g(\mathbf{p} - \mathbf{x})d\mathbf{x}$. Then the formulas above read

$$\begin{aligned} I_{\mathbf{p}}(1) &= (1_D * 1_B)(\mathbf{p}) \\ I_{\mathbf{p}}(\mathbf{x}) &= (1_D * (1_B \cdot f))(\mathbf{p}), \quad \text{where } f(\mathbf{x}) = -\mathbf{x}, \\ I_{\mathbf{p}}(\mathbf{x}\mathbf{x}^T) &= (1_D * (1_B \cdot g))(\mathbf{p}), \quad \text{where } g(\mathbf{x}) = \mathbf{x}\mathbf{x}^T. \end{aligned}$$

This means that we may compute all necessary information by convolving the indicator function 1_D with the 10 different functions $1_B(x, y, z)$, $-x \cdot 1_B(x, y, z)$, $-y \cdot 1_B(x, y, z)$, $-z \cdot 1_B(x, y, z)$, $x^2 \cdot 1_B(x, y, z)$, $y^2 \cdot 1_B(x, y, z)$, $z^2 \cdot 1_B(x, y, z)$, $xy \cdot 1_B(x, y, z)$, $xz \cdot 1_B(x, y, z)$, $yz \cdot 1_B(x, y, z)$.

For the actual computation, the indicator function 1_D is represented as an occupancy voxel grid, as discussed by Gelfand et al. (2005). Computing convolutions via FFT means that we compute the result of convolution for all points in a cube-shaped domain at once. In order to optimize computation time, we need to cover the boundary of D (i.e., the surface Φ to be analyzed) by suitable collection of cubes, so as not to compute too many values irrelevant

for curvature information at the boundary of the domain D .

4.2 Sphere neighbourhood computations

Here we use a more geometric method, which is based on an almost uniform multilevel discretization of the sphere $S_r(\mathbf{p})$. By discrete integration over the mesh faces, we can compute any integral invariant which is defined via the sphere neighbourhood.

To accelerate the computation of the sphere neighbourhood $D \cap S_r(\mathbf{p})$, we use a 4-level hierarchical representation of the sphere. It has a highly regular triangulation with 218 faces at the coarsest level. The next finer level is obtained by splitting each triangle into 4 sub-triangles, where new vertices get projected onto the sphere. During refinement, we maintain a list of points inside D . A specified face F at a given level is put into this list, if all 3 vertices are in D . Otherwise, we split the face into 4 sub-faces and judge them one by one in the next level. If we reach the finest discretization level, we use the barycenter of the face to judge inside-outside information. After a hierarchical traversal, we get a list of inside faces which may belong to different discretized sphere

Model	Number of triangles	time for PCA ball	time for PCA sphere	Figure number
pillow	24576	23.3 s	8.1 s	*2 and *3
dragon	209227	83.9 s	39.1 s	*4 and *5
bunny	69451	50.1 s	15.1 s	4

Table 1

Computation times for principal component analysis. A star before a figure number refers to figures in the paper Yang et al. (2006).

Model	No. of triangles	type of PCA	time for PCA	time for curves	Figure number
horse	114560	ball	46.7s	56.6s	*6, left
pipe	30560	ball	39.7s	5.5s	3, left
D-form	16384	sphere	4.8s	2.8s	3, right

Table 2

Computation times for principal curves. A star before a figure number refers to figures in the paper Yang et al. (2006).

levels. PCA is based on the union of these inside faces.

Table 1 gives some computation times of PCA on ball and sphere neighbourhood, respectively. We have done the experiments on a 2.8 GHz PC with 2GB RAM. Table 2 shows the computation times for principal curves.

5 Conclusion

Principal component analysis of local neighbourhoods defined by a ball kernel or spherical kernel allows to define principal directions and principal curvatures on a given scale. We have shown an asymptotic analysis of the principal moments and directions of inertia, and their relation to principal curvatures. Geometry descriptors derived in this way are robust against noise and exhibit multiscale behaviour suitable for applications like feature extraction and computation of principal curves.

Acknowledgements

This research was supported by grants No. 16002-N05 and S9207 of the Austrian Science Fund (FWF).

References

- Alliez, P., Cohen-Steiner, D., Devillers, O., Lévy, B., Desbrun, M., 2003. Anisotropic polygonal remeshing. *ACM Trans. Graphics* 22 (3), 485–493.
- Bajaj, C., Xu, G., 2003. Anisotropic diffusion on surfaces and functions on surfaces. *ACM Trans. Graphics* 22 (1), 4–32.
- Cazals, F., Chazal, F., Lewiner, T., 2003. Molecular shape analysis based upon the Morse-Smale complex and the Connolly function. In: *Proc. Symp. Comp. Geometry*. pp. 351–360.
- Cazals, F., Pouget, M., 2003. Estimating differential quantities using polynomial fitting of osculating jets. In: Kobbelt, L., Schröder, P., Hoppe, H. (Eds.), *SGP '03: Proceedings of the symposium on Geometry processing*. Eurographics Association, pp. 177–178.
- Cazals, F., Pouget, M., 2005. Topology driven algorithms for ridge extraction on meshes. *Tech. Rep. 5526*, INRIA.
URL <http://www.inria.fr/rrrt/rr-5526.html>
- Clarenz, U., Rumpf, M., Schweitzer, M. A., Telea, A., 2004a. Feature sensitive multiscale editing on surfaces. *Visual Computer* 20 (5), 329–343.

- Clarenz, U., Rumpf, M., Telea, A., 2004b. Robust feature detection and local classification for surfaces based on moment analysis. *IEEE Transactions on Visualization and Computer Graphics* 10 (5), 516–524.
- Cohen-Steiner, D., Morvan, J.-M., 2003. Restricted delaunay triangulations and normal cycle. In: *SCG '03: Proc. 19th symposium on Computational geometry*. ACM Press, pp. 312–321.
- Connolly, M., 1986. Measurement of protein surface shape by solid angles. *J. Mol. Graphics* 4 (1), 3–6.
- Davis, C., Kahan, W. M., 1970. The rotation of eigenvectors by a perturbation. III. *SIAM J. Numer. Anal.* 7, 1–46.
- Desbrun, M., Grinspun, E., Schröder, P., 2005. *Discrete differential geometry: An applied introduction*. SIGGRAPH Course Notes.
- do Carmo, M., 1976. *Differential Geometry of Curves and Surfaces*. Prentice-Hall.
- Gelfand, N., Mitra, N. J., Guibas, L. J., Pottmann, H., 2005. Robust global registration. In: Desbrun, M., Pottmann, H. (Eds.), *Symposium Geometry Processing*. Eurographics Association, pp. 197–206.
- Goldfeather, J., Interrante, V., 2004. A novel cubic-order algorithm for approximating principal direction vectors. *ACM Transactions on Graphics* 23 (1), 45–63.
- Hertzmann, A., Zorin, D., 2000. Illustrating smooth surfaces. In: *SIGGRAPH '00*. ACM Press/Addison-Wesley Publishing Co., pp. 517–526.
- Hildebrandt, K., Polthier, K., 2004. Anisotropic filtering of non-linear surface features. *Computer Graphics Forum* 23 (3), 391–400, *Proc. Eurographics*.
- Hildebrandt, K., Polthier, K., Wardetzky, M., 2005. Smooth feature lines on surface meshes. In: Desbrun, M., Pottmann, H. (Eds.), *SGP 2005: Third Eurographics Symposium on Geometry Processing*. Eurographics Association, pp. 85–90.
- Hulin, D., Troyanov, M., 2003. Mean curvature and asymptotic volume of small balls. *Amer. Math. Monthly* 110, 947–950.
- Liu, Y., Pottmann, H., Wallner, J., Yang, Y., Wang, W., 2006. Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graphics* 25 (3), 681–689.
- Manay, S., Hong, B.-W., Yezzi, A. J., Soatto, S., 2004. Integral invariant signatures. In: *Proceedings of ECCV 2004*. Vol. 3024 of *Lecture notes in Computer Science*. Springer, pp. 87–99.
- Ohtake, Y., Belyaev, A., Seidel, H.-P., 2004. Ridge-valley lines on meshes via implicit surface fitting. *ACM Trans. Graphics* 23 (3), 609–612.
- Osher, S., Fedkiw, R., 2002. *The Level Set Method and Dynamic Implicit Surfaces*. Springer.
- Pauly, M., Keiser, R., Gross, M., 2003. Multi-scale feature extraction on point-sampled geometry. *Computer Graphics Forum* 22 (3), 281–289.
- Peternell, M., 2004. Developable surface fitting to point clouds. *Computer Aided Geometric Design* 21, 785–803.
- Pottmann, H., Wallner, J., 2001. *Computational Line Geometry*. Springer.

- Taubin, G., 1995. Estimating the tensor of curvature of a surface from a polyhedral approximation. In: ICCV '95: Proceedings of the Fifth International Conference on Computer Vision. IEEE Computer Society, pp. 902–907.
- Taubin, G., Cooper, D. B., 1992. Object recognition based on moment (or algebraic) invariants. In: Mundy, J. L., Zisserman, A. (Eds.), Geometric invariance in Computer Vision. MIT Press, pp. 375–397.
- Tong, W.-S., Tang, C.-K., 2005. Robust estimation of adaptive tensors of curvature by tensor voting. *IEEE Pattern Analysis Machine Intelligence* 27 (3), 434–449.
- Yang, Y.-L., Lai, Y.-K., Hu, S.-M., Pottmann, H., 2006. Robust principal curvatures on multiple scales. In: Polthier, K., Sheffer, A. (Eds.), SGP 2006: 4th Eurographics Symposium on Geometry processing. Eurographics Association, pp. 223–226.
- Yoshizawa, S., Belyaev, A., Seidel, H.-P., 2005. Fast and robust detection of crest lines on meshes. In: SPM '05: Proceedings of the 2005 ACM symposium on Solid and physical modeling. ACM Press, pp. 227–232.