Konrad-Zuse-Zentrum
für Informationstechnik Berlin

Takustraße 7
D-14195 Berlin-Dahlem
Germany

KLAUS HILDEBRANDT AND KONRAD POLTHIER

Anisotropic Filtering of Non-Linear Surface Features

ZIB-Report 04-25 (June 2004)

Anisotropic Filtering of Non-Linear Surface Features[‡]

Klaus Hildebrandt[*] Konrad Polthier[*]

Zuse Institute Berlin[†]

Dedicated to the 65^{th} birthday of Prof. Dr. Hermann Karcher

Abstract

A new method for noise removal of arbitrary surfaces meshes is presented which focuses on the preservation and sharpening of non-linear geometric features such as curved surface regions and feature lines. Our method uses a prescribed mean curvature flow (PMC) for simplicial surfaces which is based on three new contributions: 1. the definition and efficient calculation of a discrete shape operator and principal curvature properties on simplicial surfaces that is fully consistent with the well-known discrete mean curvature formula, 2. an anisotropic discrete mean curvature vector that combines the advantages of the mean curvature normal with the special anisotropic behaviour along feature lines of a surface, and 3. an anisotropic prescribed mean curvature flow which converges to surfaces with an estimated mean curvature distribution and with preserved non-linear features. Additionally, the PMC flow prevents boundary shrinkage at constrained and free boundary segments.

1 Introduction

Noise is an omnipresent artifact in 2d and 3d meshes due to resolution problems in mesh acquisition processes. For example, meshes extracted from image data or supplied by laser scanning devices often carry high-frequency noise in the position of the vertices. Many filtering techniques have been suggested in recent years, among them Laplace smoothing is the most prominent example. In practice, denoising is still a delicate task and left to the hands of a user who carefully chooses different filtering algorithms.

Anisotropic denoising concentrates on the preservation of important surface features like sharp edges and corners by applying direction dependent smoothing. For example, a sharp edge remains sharp when smoothing is avoided to happen across the edge.

In geometry, different notions of curvature have been established to detect and measure the bending and the geometric disturbance of a shape. One approach to denoise a shape therefore concentrates on the removal of unwanted curvature peaks while a feature preservation simultaneously tries to keep certain curvature distributions, for example, the high curvature along sharp corners. Anisotropic mean curvature flow addresses this problem by constraining the isotropic mean curvature flow to preserve features encountered in a shape.

A good knowledge of curvature is an eminent prerequisite for constrained mesh smoothing. Especially for feature constrained denoising the computation of principal curvatures on simplicial surfaces is important since it measures the individual bending of a surface in different directions. The results of this paper are based on the novel definition and explicit calculation of a shape operator and principal curvature information on a simplicial surface. These definitions rely on a smallest possible stencil for curvature calculations and are still fully consistent with the known vertex-based discrete mean curvature formulas. We incorporate these operators in new kinds of diffusion algorithms for the feature preserving denoising of meshes.

1.1 Related Work

On simplicial surfaces the definition of discrete versions of the various curvature notions has a long history. The

[*]Supported by the DFG research center "Mathematics for key technologies" (FZT 86) in Berlin.

[†]e-mail: {hildebrandt — polthier}@zib.de

[‡]This is a preprint of an article accepted for publication in Computer Graphics Forum - Eurographics, Blackwell Publishing, 2004.

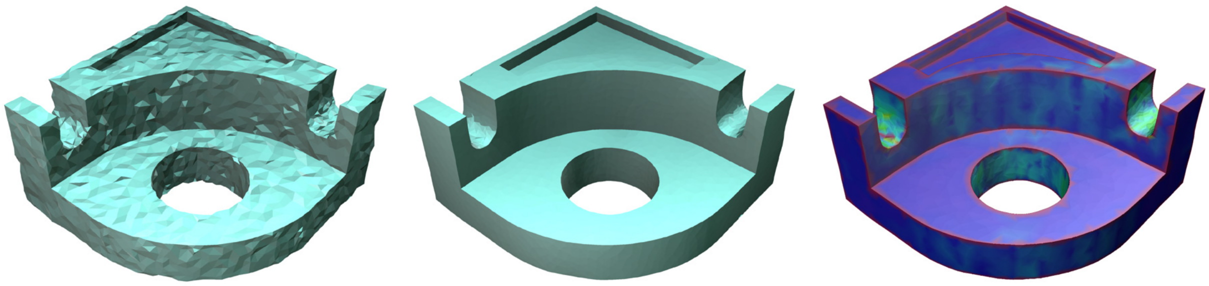
discrete Gauß curvature defined as angle defect at a vertex played a major role in the work of Alexandrov [1]. The simplicial mean curvature defined as gradient of the simplicial surface area has a simple intrinsic description as the sum of the weighted edges emanating from a vertex, and led to several algorithms for the computation of minimal and constant mean curvature surfaces, see [22] for an overview.

Different approaches have been made to calculate the principal curvature directions of a simplicial surface. A common adhoc approach uses a quadratic surface to estimate a pair of principal curvatures at the center of a triangle. Here the quadric is the unique surface which interpolates the six vertices of a triangle and its three neighbors. The principal curvatures of this quadric are evaluated at the point corresponding to the barycenter of the triangle and assigned as constant discrete principal curvatures of the inner triangle. Taubin [28] uses an approximation of a formula known from the theory of smooth surfaces to compute the directional curvature. An estimation of the shape operator at each vertex of a surface is computed based on the directional curvature of the emanating edges. Meyer et al. [16] combine the scalar-valued simplicial Gauß and mean curvatures to estimate principal curvature values with a formula known from the smooth case. They derive principal directions by a best quadratic fit of a pair of two orthogonal tangent vectors. A relation of the obtained principal curvature directions and the otherwise obtained discrete mean curvature is not obvious. Cohen-Steiner and Morvan [5] define an integrated shape operator for subsets of a simplicial surface in $\mathbb{R}^3$ using the theory of normal cycles. For a special class of approximations of a smooth surface S, namely restricted Delaunay triangulation of a vertex sample of S, they derive bounds on the error between the estimated and the smooth curvature.

The most common techniques for fairing and denoising of surfaces are based on Laplace smoothing. This can be modeled as a solution of the diffusion equation $\partial_t F = \Delta_M F$ where F is the parametrization of the surface and Δ_M is the Laplace-Beltrami operator. On surfaces the Laplace smoothing is equivalent to the mean curvature flow since the Laplace-Beltrami operator equals the mean curvature vector. Many improvements and extensions of the Laplace smoothing for surface fairing and denoising have been proposed. Taubin [29] developed a fast and simple iterative scheme to integrate the diffusion equation and designed a low pass filter by alternating the sign in the Laplace smoothing. Desbrun et al. [7] suggested to use an implicit integration scheme to allow larger time-steps and to stabilize the flow. To compensate shrinkage of the surface and to additionally avoid undesired deformations of the shape, Liu et al. [15] proposed a method that keeps the volume of each star of a vertex, and Vollmer et al. [31] suggested a method that is based on the idea to push the vertices back to their previous positions. Ohtake et al. [17] extended the Laplace smoothing by combining it with mesh regularization. Kuriyama and Tachibana [14] and Rumpf et al. [9] connected surface fairing to subdivision. In order to get smoothness at the boundary Schneider and Kobbelt [26] propose a forth order method that smoothes the surface based on mean curvature values given at the boundary.

Anisotropic smoothing methods were developed to preserve and enhance features like sharp edges or corners while denoising the surface. The main difference to isotropic schemes is the way how areas with highly different principal curvatures are processed. Usually, such areas contain significant shape information, i.e. sharp edges have one large and one vanishing principal curvature. An anisotropic scheme evolves the surface in a way that the smaller principal curvature value is reduced and the larger value is kept. This produces sharp edges. Unfortunately, the anisotropic smoothing tends to converge against linear features like straight lines and flat planes. One of the contributions of this paper is the extension of this technique to allow non-linear curved features as stable limits. Anisotropic scheme were first introduced in image processing and later extended to geometric problems, for example, by Desbrun et al. [8] to smooth high fields and by Rumpf et al. [3] for surfaces, level sets [24] and to process textures [4] on the surface as well. Bajaj and Xu [2] developed a scheme to smooth higher order functions on surfaces while fairing it. Other methods [30][27] use diffusion filters to smooth the normal field and then integrate this to get the smoother surface. Recently Fleishman et al. [10] described a method that generalizes the bilateral filtering approach known in image processing to meshes. The basic idea is to regard a neighborhood of each vertex as a distance graph over its tangent plane. Then the graph corresponds to the gray level of an image. Large values of the graph indicate surface features. The method can preserve some kinds of features but fails to reconstruct sharp edges, compare the results shown in their Fig. 6 with our Fig. 2.

Alternative methods use surface energies [6] [11] like the total curvature [32][25], a membrane energy [13] and more recently statistical measures [12] and a



(a) (b) (c)

Figure 1: Noisy mesh with curved feature lines is smoothed using the anisotropic PMC flow. (a) Noisy surface. (b) Surface after denoising. (c) The surface is colored by the absolute value of the predominating principal curvature.

Wiener filter to denoise surface meshes [18].

1.2 Contributions

The focus of our work targets three problems:

- A discrete shape operator and principal curvature directions.

We define an edge based shape operator and principal curvatures of simplicial surfaces explicitly in terms of a discrete surface. The direct calculation avoids the need of higher order interpolating surfaces, and effectively simplifies and accelerates curvature calculations. The small stencil of our operators also avoids smoothing side-effects introduced when using higher order approximations.

- An anisotropic mean curvature vector and flow.

The small stencil of our shape operator is used to develop an improved anisotropic diffusion algorithm with a better feature recognition. Our anisotropic mean curvature flow reproduces sharp features with very high quality when compared to previous approaches.

- A smoothing algorithm based on a *prescribed* mean curvature flow (PMC).

The anisotropic prescribed mean curvature flow solves the problem of shrinkage and undesired deformation of the surface for anisotropic smoothing. It additionally extends the known anisotropic smoothing techniques by allowing to correctly preserve non-linear features like the sharp circular corner of a drilled hole. Cylindrical shapes like those shown in Fig. 1 and 5 appear as stable limits of the flow.

1.3 Paper Organization

In Section 2 we derive a novel discrete shape operator for simplicial surfaces and explain its relation to the known discrete mean curvature vector. Based on the shape operator we define in Section 2.1 an anisotropic mean curvature vector and an anisotropic mean curvature flow. In Section 3 we introduce a discrete *prescribed* mean curvature flow that solves the problem of shrinkage of curved surface regions and allows curved surfaces such as cylinders to appear as stable limits of the smoothing. In Section 4 we incorporate anisotropy into the PMC flow to denoise and sharpen non-linear features like round edges which typically appear in CAD models. Section 5 summarizes the experimental results and discusses different integration schemes.

2 Discrete Shape Operator and Principal Curvatures

The shape operator determines the principal curvature values and directions on a surface. In this section we derive a discrete shape operator based on the smallest possible stencil consisting of two adjacent triangles. Especially the detection of sharp surface features requires a curvature notion based on a small stencil to avoid blurring of sharp features.

The well-known mean curvature vector $\overrightarrow{H}$ at a vertex equals the gradient of the area functional whose explicit representation

$$\overrightarrow{H}(p) = \frac{1}{2} \sum_{q \in \operatorname{link} p} (\cot \alpha_q + \cot \beta_q)(p - q) \quad (1)$$

was derived in the context of discrete minimal surfaces [19]. This vertex based mean curvature can be reformulated in terms of an edge based mean curvature vector

$$\overrightarrow{H}(e) = H_e \overrightarrow{N}_e \tag{2}$$

which is the area gradient of a non-conforming mesh [20]. If θ_e denotes the dihedral angle of the edge e and $\overrightarrow{N}_e = \frac{N_1+N_2}{\|N_1+N_2\|}$ the edge normal, then $H_e = 2\,|e|\cos\frac{\theta_e}{2}$ is the mean curvature at the edge. Following [20] both mean curvature vectors (1) and (2) are related by the equation

$$\overrightarrow{H}(p) = \frac{1}{2} \sum_{e=(p,q),q\in \mathrm{link}\, p} \overrightarrow{H}(e). \tag{3}$$

For smooth surfaces the shape operator S is a symmetric operator that applies to tangential vector fields. In the discrete case we specify $S(e)$ to be an operator in $\mathbb{R}^3$ that has the edge normal $\overrightarrow{N}_e$ in its null space. We base the operator on the following remarks. Let $\overrightarrow{e}$ denote a unit vector in direction of the edge e. Since the normal does not change along the edge e, $\overrightarrow{e}$ is in the null space of S. For all other tangential directions v the normal curvature $\langle v, Sv\rangle$ is either strictly positive, strictly negative or zero. This means that each point on an edge is parabolic or flat. As a consequence we see that S has rank ≤ 1, that $\overrightarrow{e}$ is an eigenvector with eigenvalue 0 and that $\overrightarrow{e} \times \overrightarrow{N}_e$ is the non-trivial eigenvector. The requirement $\mathrm{trace}\, S(e) = H_e$ determines the non-trivial eigenvalue.

Therefore, we define the shape operator of a piecewise linear surface M_h in $\mathbb{R}^3$ at the inner edges e of M_h by

$$S(e) = H_e(\overrightarrow{e} \times \overrightarrow{N}_e)(\overrightarrow{e} \times \overrightarrow{N}_e)^t. \tag{4}$$

At a vertex $p \in M_h$ the tangent space $T_p M_h$ is given by the two dimensional subspace orthogonal to the vertex normal. Let $\overrightarrow{t}_e$ denote the unit vector in the direction of $\overrightarrow{e} \times \overrightarrow{N}_e$ projected onto $T_p M_h$. The representation of the shape operator of M_h at a vertex p is

$$S(p) = \frac{1}{2} \sum_{e=(p,q),q\in link(p)} \omega_e H_e \overrightarrow{t}_e \overrightarrow{t}_e^{\,t}, \tag{5}$$

where $\omega_e = \langle N_p, N_e\rangle$. Note that $\mathrm{trace}\, S(p) = H_p$ is ensured by the choice of ω.

Using the theory of normal cycles Cohen-Steiner and Morvan [5] define a similar integrated curvature operator for simplicial surfaces. On the star of an edge e their operator differs from our operator only in second order of the circumradius of the triangles adjacent to e. This allows to apply their error estimates and convergence analysis to our operator as well. Additionally our operator fits well with other discrete differential operators such as the discrete mean curvature vector (1).

2.1 Anisotropic Mean Curvature Vector

```
AnisotropicSmoothing (M, λ, s,
n)
  for (steps=1... n)
    Δλ = 0
    for each edge e = (vi, vj)
      compute He, Ne
      Δλ[vi]− = (wλ(He) He) ∗ Ne
      Δλ[vj]− = (wλ(He) He) ∗ Ne
    for each triangle t = (vi, vj, vk)
      compute area_t
      areaStar[vi]+=area_t
      areaStar[vj]+=area_t
      areaStar[vk]+=area_t
    for each vertex v
      v+ = 3s/(2areaStar[v]) ∗ Δλ[v]
  return M
```

Table 1: The explicit anisotropic mean curvature flow. The parameters are: M a mesh, λ the feature detection parameter, the scaling factor s determines the step width, and n is the number of explicit smoothing steps.

In the previous section we decomposed the mean curvature vector into a sum of vectors of the form $H_e\overrightarrow{N}_e$ located at the edges (3) and showed that the term H_e measures the directional curvature of the surface in the direction orthogonal to the edge. Now we obtain the anisotropic mean curvature vector $\vec{H}_A$ at a vertex p as a weighted sum over the contributions $H_e\overrightarrow{N}_e$ at the edges incident to a vertex p:

$$\vec{H}_A(p) = \frac{1}{2} \sum_{e=pq,q\in \mathrm{link}\, p} w(H_e) H_e \overrightarrow{N}_e. \tag{6}$$

The choice of the weight function w determines the anisotropic mean curvature vector. We use the function

$$w_{\lambda,\,r}(a) = \begin{cases} 1 & for\ |a| \leq \lambda \\ \frac{\lambda^2}{r(\lambda-|a|)^2+\lambda^2} & for\ |a| > \lambda. \end{cases}.$$

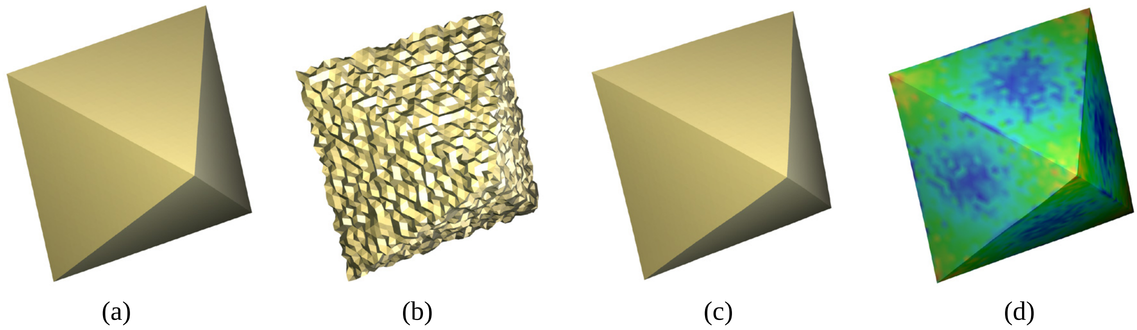


Figure 2: The anisotropic mean curvature flow preserves and sharpens linear features like edges and corners of a surface. (a) Original surface. (b) Surface with normal and tangential noise. (c) Reconstructed surface after 25 steps of anisotropic mean curvature flow. (d) Distance of each vertex of the smoothed surface to the corresponding vertex of the original mesh is indicated by a color, ranging from blue to red.

that provides a smooth transition between those areas that are smoothed and those that are kept as features. We call the parameter λ the feature detection parameter. It is handed to the user and specified for each process individually. The parameter r controls the width of the transition. In our experiments we used $r = 10$ ensuring that $w_{\lambda,10}(2\lambda) < 0.1$.

2.2 Explicit Anisotropic Mean Curvature Flow

In this section we present an explicit anisotropic mean curvature flow that combines the advantages of the mean curvature flow with the ability to preserve and sharpen linear features like edges and corners of a surface while removing noise. It can be seen as a discretization of the anisotropic geometric diffusion equation used by Rumpf et al. [3] although we solely rely on intrinsic information of our discrete shape operator and avoid the usage of any higher order interpolating surfaces.

Here we integrate the flow of the anisotropic mean curvature vector $\vec{H}_A$ with an explicit Euler method. This leads to an algorithm that is easy to understand and implement. The description of a semi-implicit integration scheme and a comparison of both methods is given in Section 5.1.

In terms of its vertices $\mathcal{P} = \{p_1, ..., p_m\}$ an explicit iteration step of the anisotropic mean curvature flow is given by

$$\mathcal{P}^{j+1} = \mathcal{P}^j - s\, M^{-1}\, \vec{H}_A(\mathcal{P}^j), \tag{7}$$

where s is the adaptive size of the integration step and M^{-1} is the inverse of the mass matrix M of the surface M_h^j. Here the mass matrix is used to convert the integrated mean curvature vector into a piecewise linear vector field. For a simplicial surface M_h with m vertices, M is the $(m \times m)$-matrix with entries:

$$M_{pq} = \begin{cases} \frac{1}{6}\text{area(star } p) & \text{if } p = q \\ \frac{1}{12}\text{area(star } e) & \text{if there is an edge } e = (p, q) \\ 0 & \text{in all other cases} \end{cases} .$$

Computing a step of the flow (7) involves solving a linear system to invert the mass matrix. A problem here is that the mass matrix can have a large condition number. An adequate solution in our case is to use a diagonalization of M with diagonal elements $M_{pp} = \frac{1}{3}area(star\ p)$ called the lumped mass matrix. Then the integration step for each vertex p is given by an explicit formula:

$$p^{j+1} = p^j - \frac{3s}{area(star\ p^j)} \vec{H}_A(p^j). \tag{8}$$

The advantage of our explicit representation of the anisotropic mean curvature vector is that the analytic machinery of the resulting algorithm reduces to less than 30 lines of code.

The smoothing process can be fine-tuned with two parameters:

- The feature detection parameter λ determines the weight function w_λ, and hence the anisotropic mean curvature vector. This provides control over what is regarded as a feature and what will be preserved during the smoothing.
- The scaling factor s determines the amount of smoothing done in a single step.

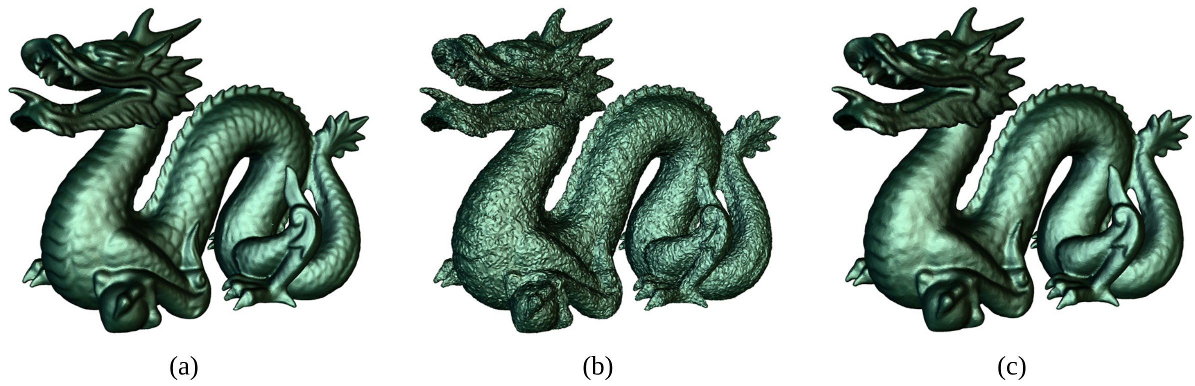


Figure 3: The prescribed mean curvature flow is used to filter the dragon corrupted with noise. The features of the surface are preserved and the shape of the features is kept. (a) The original surface. (b) The model corrupted with noise. (c) The reconstructed dragon. (Mesh from Stanford University - 3D scanning repository.)

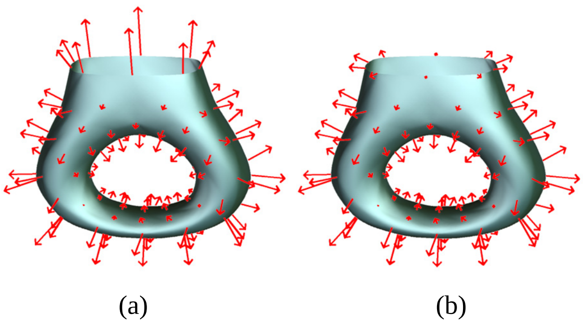


Figure 4: Avoiding boundary shrinkage. The tangential tension shown in the standard discrete Laplacian (a) is clearly avoided in the modified Laplacian (b).

2.3 Smoothing Surfaces with Boundary

A common problem of smoothing algorithms is the extension of the method to the boundary of the surface. For diffusion based methods this requires to extend the definition of the Laplacian to the boundary in a consistent way. At an inner vertex the Laplacian (1) is normal to the surface such that it is often used to define the normal of a vertex. But at the boundary that Laplacian has a strong tangential component since the outer edges are missing to compensate the surface tension. For smoothing algorithms the tension causes the problem of boundary shrinkage. To compensate for this effect Taubin [30] proposed to project the Laplacian of each boundary vertex onto a normal vector that is computed by averaging over the normals of the adjacent faces.

Computing the Laplacian as a weighted sum of edge normals instead of edges, see (2) and (3), leads to the same result at all inner vertices but differs at the boundary. The sum of the edge normals can be interpreted as a weighted sum of the face normals where the weights are determined by the edge curvatures. Consequently it avoids the tangential components and thus provides a better definition of a normal at boundary vertices. The problem of boundary shrinkage is efficiently reduced by this operator without the need for a projection or other extra treatment. Additionally this ensures that the boundary is smoothed with the same speed as the interior parts of a surface. The representation of the mean curvature vector at the boundary generalizes to the anisotropic mean curvature vector in a natural way.

3 Prescribed Mean Curvature Flow

For surfaces, the Laplacian applied to the identity map equals the area gradient at each vertex of the surface. Hence, Laplacian smoothing is equivalent to minimizing surface area. Depending on the boundary constraints the limit is therefore a minimal surface, or a degenerate situation like a singular point. For smoothing this causes the problem of shrinkage of the surface. For each region of the surface the speed of the shrinking depends on the curvature in that part, i.e. areas with high mean curvature shrink faster than others. This leads to undesired deformations of the surface. The anisotropic smoothing slows down the smoothing process in regions with high curvature, hence suppresses the shrink-

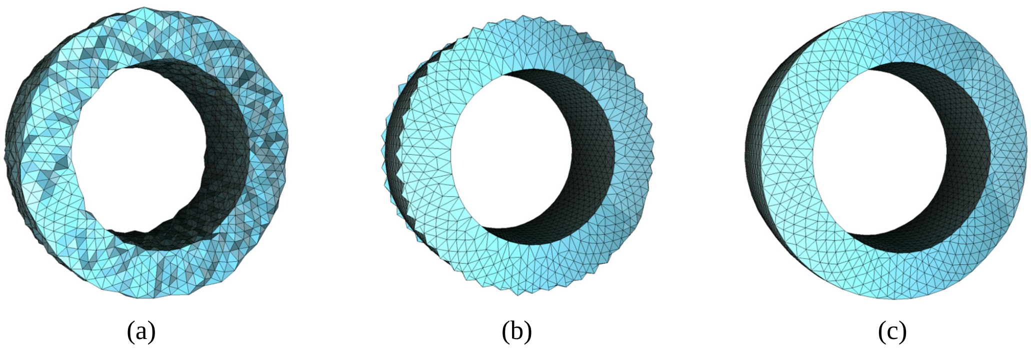


Figure 5: Starting with a noisy mesh (a) the anisotropic MC flow contracts the round feature lines and fails to recover the curved edges (b). In contrast, the anisotropic PMC flow converges to a stable limit (c).

ing in these areas. This can cause even stronger deformations of the surface or even degeneration of mesh, cf. Fig. 5. To the authors knowledge no adequate method to compensate the deformations for the anisotropic case is known.

In this section we introduce a fairing technique that during the evolution of the surface smoothes its mean curvature distribution rather than only reducing the surface area. The method preserves the features of the surface during the smoothing process and avoids the deformations described above. It is applicable to the anisotropic case, too. The algorithm is described in two steps. First we extend the mean curvature flow such that instead of converging to a surface with zero mean curvature, the new flow allows to evolve the surface towards a surface having a prescribed mean curvature. We call this flow prescribed mean curvature flow (PCM). Then instead of smoothing the surface directly, we compute its mean curvature, smooth this scalar field and use the PCM flow to evolve it towards a surface with this smoothed mean curvature. We describe the isotropic PMC flow in this section and generalize it to the anisotropic case in the next section.

The design of the PMC flow is motivated by properties of surfaces of constant mean curvature. These are known to be critical with respect to the area functional for any variation that preserves the volume and fixes the boundary. For discrete surfaces the same characterization means that

$$\nabla_p \, \text{area} = H \, \nabla_p \, \text{vol} \tag{9}$$

is valid for all interior vertices p and a constant H [21]. The volume of a surface is the orientated volume enclosed by the cone of the surface over the origin in $\mathbb{R}^3$,

$$vol \; M_h = \frac{1}{6} \sum_{T=(p,q,r)\in M_h} < p, q \times r > .$$

The gradient of $vol \; M_h$ is

$$\nabla_p vol = \frac{1}{6} \sum_{T=(p,q,r)\in M_h} q \times r.$$

We define the isotropic prescribed mean curvature flow of a simplicial surface M_h with vertices $\mathcal{P} = \{p_1, ..., p_m\}$ and a function $f(\mathcal{P})$ on the vertices of M_h by

$$\frac{\partial}{\partial t}\mathcal{P} = -M^{-1}(\overrightarrow{H}(\mathcal{P}) - f(\mathcal{P}) \cdot \nabla_p vol), \tag{10}$$

where M is the mass matrix of M_h.

An explicit step of the isotropic smoothing algorithm consists of two parts. First, compute the piecewise linear scalar mean curvature $M^{-1}H$ of the actual surface M_h and smooth $M^{-1}H$ at each vertex p by averaging over the neighbors of p. Secondly, compute a step of the PMC flow of M_h using the smoothed $M^{-1}H$ as the function f that prescribes the target curvature.

4 Denoising Non-Linear Surface Features

A main characteristic of anisotropic smoothing, in comparison to isotropic methods, is the way sharp edges of a surface are processed. Sharp edges are features characterized by a large and a smaller principal curvature

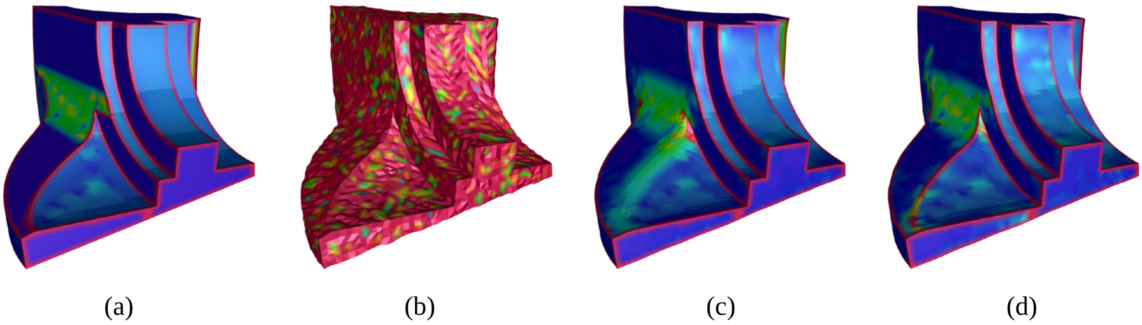


Figure 6: Comparison of the anisotropic mean curvature flow and the PMC flow on the fandisk model that has a vanishing ridge. Whereas the PMC flow preserves the ridge, the anisotropic flow flattens it. The models are colored by the absolute of the predominant principal curvature. (a) Original model. (b) Noisy model. (c) Denoised model using the anisotropic MC flow. (d) Best results with the anisotropic PMC flow. (Original mesh courtesy of H. Hoppe.)

value. The anisotropic smoothing sharpens the edges, this means that the smaller principal curvature is reduced until it vanishes. The results are sharp edges that are part of a straight line. This works fine, unless the feature itself is curved. In this section we extend the PMC flow described in the last section to the anisotropic mean curvature. This allows to denoise surfaces with sharp curved features like the curved boundary of a hole.

Analog to the isotropic case the anisotropic PMC flow is defined by

$$\frac{\partial}{\partial t}\mathcal{P} = -M^{-1}(\overrightarrow{H}_A(\mathcal{P}) - f(\mathcal{P}) \cdot \overrightarrow{V}_A(\mathcal{P})), \qquad (11)$$

where $\overrightarrow{H}_A$ is the anisotropic mean curvature vector defined in Section 2.1 and f is a function, that prescribes the anisotropic mean curvature. The term $\overrightarrow{V}_A$ is an anisotropic analog of the volume gradient. We call the vertices p with $\overrightarrow{H}_A(p) \neq \overrightarrow{H}(p)$ the feature vertices and set $\overrightarrow{V}_A(p) = \nabla_p vol$ for all non-feature vertices p. For the other vertices we set

$$\overrightarrow{V}_A(p) = \operatorname{sign}(\langle \overrightarrow{e}_{H_A}(p), \nabla_p vol\rangle)\overrightarrow{e}_{H_A}(p)$$

where $\overrightarrow{e}_{H_A}$ is the unit vector field of $\overrightarrow{H}^s_A$ and we get $\overrightarrow{H}^s_A$ by performing a simple smoothing step on $\overrightarrow{H}_A$. In our experiments we used

$$\overrightarrow{H}^s_A(p) = \frac{1}{2}(\overrightarrow{H}_A(p) + \frac{1}{\sum\limits_{q\in \mathrm{link}\, p} \omega_q} \sum_{q\in \mathrm{link}\, p} \omega_q \overrightarrow{H}_A(q))$$

$$\overrightarrow{e}_{H_A}(p) = \overrightarrow{H}^s_A(p) / \left\|\overrightarrow{H}^s_A(p)\right\|.$$

where ω_q is the sum of the vertex angles at p in the triangles adjacent to the edge $\overline{pq}$.

An explicit integration step of the PMC flow consists of two parts. First, compute $f = M^{-1}H_A$ and smooth this scalar field. Second, compute the new positions of the vertices by using the iterative formula

$$p^{j+1} = p^j - \frac{3s}{area(star(p^j))}(\overrightarrow{H}_A(p^j) - f(p^j) \cdot \overrightarrow{V}_A(p^j))$$

for each vertex p^j of M^j_h. When smoothing the anisotropic scalar mean curvature, we must take care to keep the sharp features. Analog to the isotropic case, we smooth $M^{-1}H_A(p)$ by averaging over the neighbor vertices of p. But to preserve the sharp edges, at each feature vertex p we only average over those neighbor vertices that are feature vertices as well. To avoid solving a linear equations system in each step, in our experiment we have used the term $H_A(p^j)/||\overrightarrow{V}_A(p^j)||$ instead of $M^{-1}H_A(p^j)$.

The thresholds to control the method are the same as those for the anisotropic mean curvature flow in Section 2.1, namely the feature detection parameter λ to determine what is regarded as a feature and the scaling factor s to control the magnitude of the smoothing steps. Additionally the control of the amount of smoothing done to the function $M^{-1}H_A$ that prescribes the curvature can be handed to the user.

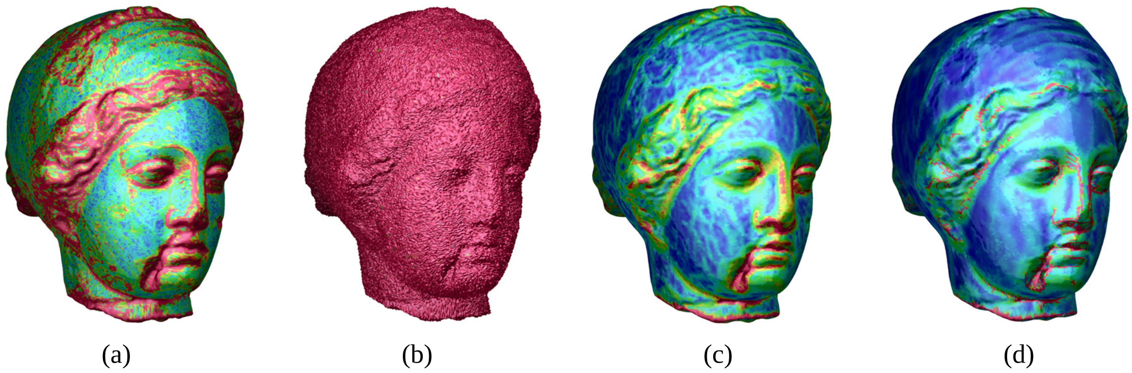


Figure 7: Application of the anisotropic mean curvature flow to the venus head corrupted with noise. The features of the surfaces are preserved, while the noise is removed. Models are colored by the absolute of the predominant principal curvature. (a) Original mesh consisting of 260k triangles. (b) Noisy head. (c) Anisotropic mean curvature flow is used to remove the noise and (d) to additionally smooth the model. (Mesh from Cyberware Incoporated.)

5 Experimental Results

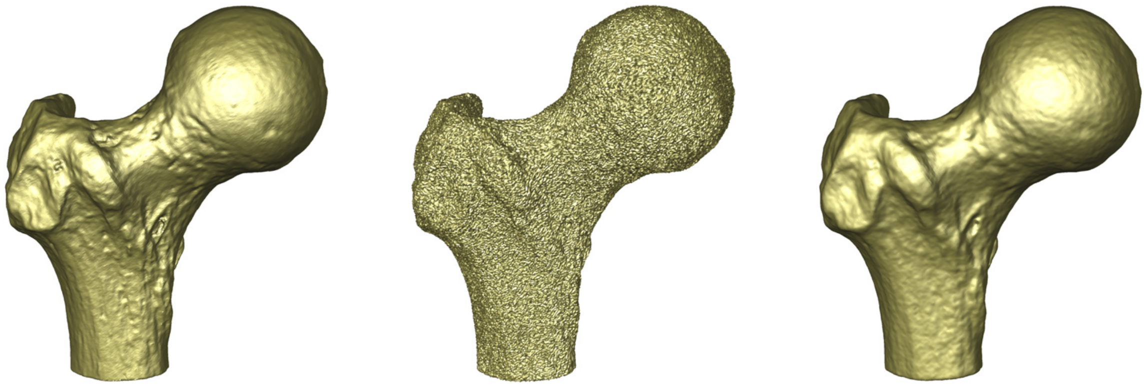
We demonstrate our results in Fig. 1-3 and 5-9. The models in Fig. 2 and 7 are smoothed with the anisotropic smoothing introduced in Section 2.1 and 2.2, and the other models with the prescribed mean curvature flow described in Section 3 and 4. A comparison of the anisotropic and the prescribed smoothing is given in Fig. 5 and 6.

Model	Fig.	#Vert.	Method
Armadillo	9	173k	Prescr.
Bearing	1	6k	Prescr.
Bone	8	137k	Prescr.
Dragon	3	125k	Prescr.
Fandisk	6	6k	both
Octahedron	2	4k	Aniso.
Ring	5	6k	both
Venus	7	130k	Aniso.

Table 2: The table lists the models used in our experiments.

Fig. 2 shows an example of the anisotropic smoothing applied to recover the surface of an octahedron, that has been corrupted with noise. Due to the explicit measurement of curvature based only on quantities of the simplicial mesh, the detection and sharpening of the features is very precise. The recovering of the edges therefore has a high quality, especially when compared to other approaches using interpolating higher order surfaces to measure curvature. An application of the anisotropic mean curvature flow to a noisy higher resolution model is shown in Fig 7.

Whereas the anisotropic MC flow can only recover straight edges, the anisotropic PMC flow is able to sharpen curved feature lines. We demonstrate this with different examples. Fig. 5 shows the surface of a ring that has been corrupted with noise. The PMC flow recovers the shape and removes the noise. The ring is a stable limit of the flow. For comparison we have processed the ring with the anisotropic MC flow, too. This flow contracts the feature lines and fails to recover the shape. While the ring surface has circular feature lines the surface shown in Fig. 1, has different types of curved feature lines, especially the curvature of some feature lines varies strongly. The prescribed mean curvature flow correctly sharpens the features. The fandisk (Fig. 6) model is a model with a vanishing and curved ridge. For comparison we tested it with both smoothing methods. The PMC flow correctly preserves the ridge while the anisotropic smoothing does not. We tested the flows on surfaces that do not have such artificial and regular feature lines but have different kinds of features, cp. Fig. 3, 9 and 8. The PMC flow proved to be very well suited to denoise the surfaces and to preserve the surface features.



(a) (b) (c)

Figure 8: The anisotropic PMC flow is used to denoise the surface of a bone. Features of the surface are preserved. (a) Original model. (b) Model corrupted with noise. (c) Reconstructed surface. (Mesh from Cyberware Incoporated.)

Model	Octahedron	Venus
#Steps	10	10
Ex. AMC flow	0.3s	13.7s
Im. AMC flow	0.9s	52.6s
Ex. PMC flow	1.8s	103.1s
Im. PMC flow	2.8s	152.4s

Table 3: Comparison of the computation time needed for 10 steps of the different flows and integration methods. Time measured using our Java implementation on a PC with a 1.6 GH Pentium 4 CPU.

5.1 Implicit Integration of the Flow

In Section 2.1 and 4 we have derived explicit integration schemes for the anisotropic MC flow (8) and the PCM flow (11), because explicit methods are simple to understand and to implement. Implicit methods stabilize the flow and allow larger integration steps, but require to set up and solve a system of equations. Desbrun et al. [7] introduced a semi implicit scheme for the mean curvature flow and Rumpf et al. [3] used a semi implicit method to integrate the anisotropic diffusion equation. In this section we describe an analog semi implicit integration scheme for the anisotropic MC flow and for the PMC flow. The anisotropic mean curvature vector $\overrightarrow{H}_A$, compare equation (6), can be represented by a matrix K_A defined by $\overrightarrow{H}_A = K_A P$ where P lists the coordinates of all vertices of the surface M_h. An implicit integration step of the anisotropic MC flow is the solution of the equation

$$(M^j + s\,K_A^j)\,P^{j+1} = M^j\,P^j, \qquad (12)$$

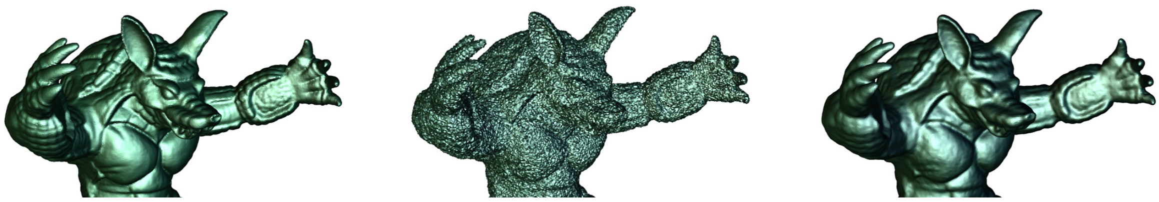
where M^j is the mass matrix of the surface M_h^j and s a scaling factor controlling the size of the step. The trick that keeps this scheme linear and is that the mass matrix and the matrix K_A are still computed on the given surface M_h^j. To solve this system of linear equations we use a preconditioned biconjugate gradient method as described in [23].

To extend this scheme to the PMC flow (11) we add the term $f\,\overrightarrow{V}_A$ that prescribes the curvature. Since the computation of this term already involves a smoothing process, it varies only little compared to $\overrightarrow{H}_A$. Thus we compute the term $f\,\overrightarrow{V}_A$ on the surface M_h^j. A step of the semi implicit scheme for the PMC flow is given by

$$(M^j + s\,K_A^j)\,P^{j+1} = M^j\,P^j + s\,f(P^j)\cdot\overrightarrow{V}_A^j. \qquad (13)$$

6 Conclusion

We presented a novel discrete shape operator whose trace is fully consistent with the well-known discrete mean curvature, and defined an anisotropic mean curvature vector. The curvature operators were used for feature preserving noise removal algorithms. Using the computation technique for constant mean curvature surfaces we modified the anisotropic mean curvature



(a) (b) (c)

Figure 9: The prescribed mean curvature flow is applied to denoise a surface with many different features. (a) The original model. (b) The mesh corrupted with noise. (c) The reconstructed surface. (Mesh from Stanford University - 3D scanning repository.)

flow such that it converges to a surface with prescribed (anisotropic) mean curvature. This allows to sharpen non-linear features such as cylindrical holes.

References

[1] A. D. Aleksandrov and V. A. Zalgaller. *Intrinsic Geometry of Surfaces*, volume 15 of *Translation of Mathematical Monographs*. AMS, 1967.

[2] C. Bajaj and G. Xu. Anisotropic diffusion on surfaces and functions on surfaces. *ACM Transactions on Graphics 22,* pages 4–32, 2003.

[3] U. Clarenz, U. Diewald, and M. Rumpf. Anisotropic geometric diffusion in surface processing. *Proc. of IEEE Visualization*, pages 397–405, 2000.

[4] U. Clarenz, U. Diewald, and M. Rumpf. A multiscale fairing method for textured surfaces. In Hans-Christian Hege and Konrad Polthier, editors, *Visualization and Mathematics III*, pages 245–260. Springer-Verlag, Heidelberg, 2003.

[5] D. Cohen-Steiner and J.-M. Morvan. Restricted Delaunay triangulations and normal cycles. *Proc. 19th Annu. ACM Sympos. Comput. Geom.*, pages 237–246, 2003.

[6] T. DeRose, T. Duchamp, H. Hoppe, J. McDonald, and W. Stuetzle. Mesh optimization. *Proc. of ACM SIGGRAPH*, pages 19–26, 1993.

[7] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. *Proc. of ACM SIGGRAPH*, pages 317–324, 1999.

[8] M. Desbrun, M. Meyer, P. Schröder, and A.H. Barr. Anisotropic feature-preserving denoising of height fields and bivariate data. *Graphics Interface*, pages 145–152, 2000.

[9] U. Diewald, S. Morigi, and M. Rumpf. A cascadic geometric filtering approach to subdivision. *Computer Aided Design*, 19:675–694, 2002.

[10] S. Fleishman, I. Drori, and D. Cohen-Or. Bilateral mesh denoising. *Proc. of ACM SIGGRAPH*, pages 950–953, 2003.

[11] M.H. Gross and A. Hubeli. Fairing of non-manifolds for visualization. *Proc. of IEEE Visualization*, pages 407–414, 2000.

[12] T. R. Jones, F. Durand, and M. Desbrun. Non-iterative, feature-preserving mesh smoothing. *Proc. of ACM SIGGRAPH*, 2003.

[13] L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel. Interactive multi-resolution modeling on arbitrary meshes. *Proc. of ACM SIGGRAPH*, pages 105–114, 1998.

[14] S. Kuriyama and K. Tachibana. Polyhedral surface modeling with a diffusion system. *Proc. of Eurographics*, pages C39–C46, 1997.

[15] X. Liu, H. Bao, H.-Y. Shum, and Q. Peng. A novel volume constrained smoothing method for meshes. *Graphical Models, volume 64*, pages 169–182, 2002.

[16] Mark Meyer, Mathieu Desbrun, Peter Schröder, and Alan H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. In Hans-Christian Hege and Konrad Polthier, editors, *Visualization and Mathematics III*, pages 35–57. Springer-Verlag, Heidelberg, 2003.

[17] Y. Ohtake, A. Belyaev, and I. Bogaevski. Mesh regularization and adaptive smoothing. *Computer Aided Design*, 2001.

[18] J. Peng, V. Strela, and D. Zorin. A simple algorithm for surface denoising. *Proc. of IEEE Visualization*, 2001.

[19] U. Pinkall and K. Polthier. Computing discrete minimal surfaces and their conjugates. *Experim. Math.*, 2(1):15–36, 1993.

[20] K. Polthier. Polyhedral surfaces of constant mean curvature. *Habilitationsschrift Technische Universität Berlin*, 2002.

[21] K. Polthier and W. Rossmann. Index of discrete constant mean curvature surfaces. *J. Reine und Angew. Math.*, pages 47–77, 2002.

[22] Konrad Polthier. Computational aspects of discrete minimal surfaces. In D. Hoffman et al., editor, *Global Theory of Minimal Surfaces*. AMS, 2004.

[23] W. Press, S. Teukolsky, W. Vetterling, and Brian Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992.

[24] M. Rumpf and T. Preußer. A level set method for anisotropic geometric diffusion in 3D image processing. *SIAM J. Appl. Math.*, 62(5):1772–1793, 2002.

[25] R. Schneider and L. Kobbelt. Generating fair meshes with G1 boundary conditions. *Proc. of ACM SIGGRAPH*, pages 247–256, 2000.

[26] R. Schneider and L. Kobbelt. Geometric fairing of irregular meshes for free-form surface design. *Computer Aided Geometric Design*, 18(4):359–379, 2001.

[27] T. Tasdizen, R. Whitaker, P. Burchard, and S. Osher. Geometric surface smoothing via anisotropic diffusion of normals. *Proc. of IEEE Visualization*, pages 125–132, 2002.

[28] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. *Proc. of fifth international conference on computer vision*, pages 902–907, 1995.

[29] G. Taubin. A signal processing approach to fair surface design. *Proc. of ACM SIGGRAPH*, 1995.

[30] G. Taubin. Linear anisotropic mesh filtering. *IBM Research Report RC2213*, 2001.

[31] J. Vollmer, R. Mencl, and H. Müller. Improved Laplacian smoothing of noisy surface meshes. *Proc. of Eurographics*, pages 131–138, 1999.

[32] W. Welch and A. Witkin. Free-form shape design using triangulated surfaces. *Proc. of Geometric Modeling and Processing*, pages 251–261, 1994.