# Anisotropic Geometric Diffusion in Surface Processing

U. Clarenz, U. Diewald, M. Rumpf [*]
Institute for Applied Mathematics
University of Bonn, Germany

## Abstract

A new multiscale method in surface processing is presented here which combines the image processing methodology based on nonlinear diffusion equations and the theory of geometric evolution problems. Its aim is to smooth discretized surfaces while simultaneously enhancing geometric features such as edges and corners. This is obtained by an anisotropic curvature evolution, where time is the multiscale parameter. Here, the diffusion tensor depends on the shape operator of the evolving surface.

A spatial finite element discretization on arbitrary unstructured triangular meshes and a semi-implicit finite difference discretization in time are the building blocks of the easy to code algorithm presented here. The systems of linear equations in each timestep are solved by appropriate, preconditioned iterative solvers. Different applications underline the efficiency and flexibility of the presented type of surface processing tool.

*Keywords:* Image Processing, Geometric Modeling, Numerical Analysis

## 1 INTRODUCTION

The processing of detailed triangulated surfaces is an important topic in computer graphics [5, 12, 13, 19]. Nowadays, various such surfaces are delivered from different measurement techniques [4] or derived from two- or three dimensional data sets [14]. Recent laser scanning technology for example enables very fine triangulation of real world surfaces and sculptures. Also from medical image generation methods such as CT and MRI devices or 3D ultrasound certain surfaces of interest can be extracted - frequently in triangulated form - at a high resolution for further post processing and analysis. These surfaces are often characterized by interesting features, such as edges and corners. On the other hand, they are typically disturbed by some noise, which is often due to local measurement errors.

The aim of this paper is to discuss a method which allows the *smoothing of discrete surfaces* and thus permits a drastic reduction of high frequency perturbations. Additionally it is able to *retain and even enhance important features such as edges and corners* on the surface. Figure 1 (cf. also color-page) shows the performance of this method and compares it to the smoothing by mean curvature flow, the appropriate geometric "Gaussian" smoothing filter.

The core of the method is a geometric formulation of scale space evolution problems for surfaces. These techniques were originally developed for image processing purposes. Thus the method not only delivers a single resulting surface, but a complete scale of surfaces in time. For increasing time, we obtain successively smoother surfaces with continuously sharpened edges.

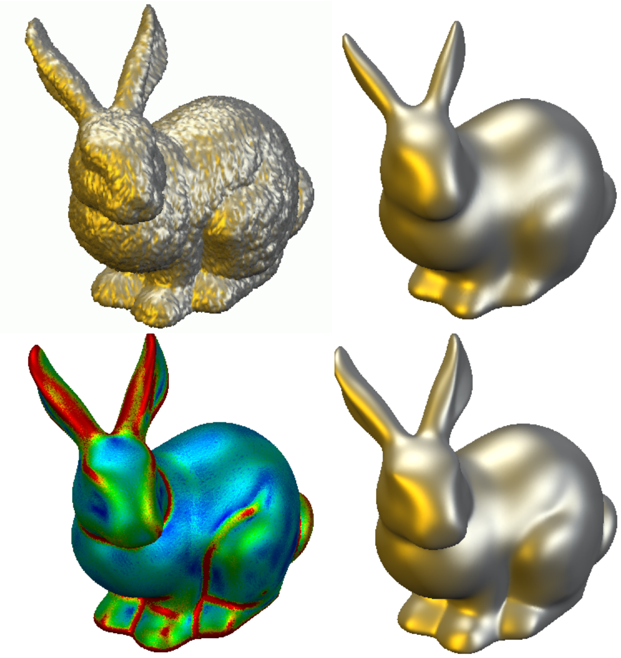

Figure 1: *A noisy initial surface (top left) is evolved by discrete mean curvature flow (top right) and by the new anisotropic diffusion method (bottom right). Furthermore for the latter surface the dominant principal curvature - on which the diffusion tensor depends - is color coded (bottom left). The snapshots are taken at the same timesteps.*

First, we derive a continuous model, which leads to a nonlinear system of parabolic partial differential equations for the coordinate mapping of the evolving surface. An anisotropic diffusion tensor depending on the shape operator and thus on the principal curvatures and principal directions of curvature, is sensitive to the identification of the important surface features and decreases the diffusivity in certain directions in close vicinity to edges or corners.

Two parameters are at the disposal of the user to influence the performance of the method:

- A threshold value $\lambda$ for principal curvatures which are assumed to indicate an edge and thus require local sharpening and
- a filter width $\epsilon$ which controls the noise reduction on the actual surface before evaluating principal curvatures.

Especially the latter is essential to make the proposed method robust and mathematically well-posed. The method presented in this paper starts with the description of a continuous model, which has many nice qualitative properties. Then in a second step we seek a robust and efficient discretization. Hence, we derive an appropriate finite element method with respect to a formulation of the continuous problem in variational form.

The paper is organized as follows.

[*] Institute for Applied Mathematics, University of Bonn, Wegelerstraße 6, 53115 Bonn, Germany, [*clarenz* | *diewald* | *rumpf*]@*iam.uni-bonn.de*

First, in Section 2 we will discuss the background work on surface fairing by geometric smoothing and on image processing on planar images. In the following Section 3 we introduce necessary mathematical notation and discuss the basic type of geometric evolution problems. Then in Section 4 we present the continuous model of anisotropic diffusion, which afterwards, in Section 5, is discretized by finite elements. The definition of a shape operator on triangulated surfaces is given in Section 6. Finally, Section 7 is concerned with the concrete implementation and in Section 8 we draw conclusions.

## 2 REVIEW OF RELATED WORK

In physics, diffusion is known as a process that equilibrates spatial variations in concentration. If we consider some initial noisy concentration or image intensity $\rho_0$ on a domain $\Omega \subset \mathbb{R}^2$ and seek solutions of the linear heat equation

$$\partial_t \rho - \Delta \rho = 0 \tag{1}$$

with initial data $\rho_0$ and natural boundary conditions on $\partial\Omega$, we obtain a scale of successively smoothed concentrations $\{\rho(t)\}_{t \in \mathbb{R}^+}$. For $\Omega = \mathbb{R}^2$ the solution of this parabolic problem coincides with the filtering of the initial data using a Gaussian filter $G_\sigma(x) = (2\pi\sigma^2)^{-1} e^{-x^2/(2\sigma^2)}$ of width or standard deviation $\sigma$, i. e. $\rho(\sigma^2/2) = G_\sigma * \rho_0$. If we discretize (1) and use an explicit Euler scheme we have to compute a sequence $\{\rho^n\}_{n=0,\cdots}$ with

$$\rho^{n+1} = (\mathrm{Id} + \tau \Delta_h)\rho^n$$

where $\tau$ is the timestep, $\Delta_h$ an approximation of the Laplacian and $\rho^0 = \rho$. Concerning the smoothing of disturbed surface geometries one may ask for analogues strategies. The counterpart of the Euclidian Laplacian $\Delta$ on smooth surfaces is the Laplace Beltrami operator $\Delta_{\mathcal{M}}$ [6, 3]. Thus, one obtains the geometric diffusion $\partial_t x = \Delta_{\mathcal{M}(t)} x$ for the coordinates $x$ on the corresponding family of surfaces $\mathcal{M}(t)$. On triangulated surfaces as they frequently appear in computer graphics applications, several authors introduced appropriate discretized operators. Taubin [19] discussed this and related approaches in the context of generalized frequencies on meshes and Kobbelt [12] used interpolation schemes. Explicit time discretizations are known to have strong timestep restrictions to ensure stability [20]. Thus, many iterations are required to obtain appropriate results. Kobbelt et al. [13] introduced multilevel strategies in the context of multiresolutional editing to improve the efficiency of these methods. Guskov et al. [8] discussed relaxation schemes with weights depending on the local geometry.

Rescently Desbrun et al. [5] considered an implicit discretization of geometric diffusion to obtain strongly stable numerical smoothing schemes. Furthermore they improved the consistency of the discrete operator on arbitrary meshes significantly. The problem of tangential coordinate shifts on the surface could be avoided, which is a drawback of some explicit methods concerning the geometric positioning of an accompanying texture. The mathematical reason for this shifting problem in geometric diffusion is that the Laplace Beltrami operator depends on the metric (cf. Section 3), thus the metric of the discrete surface should be kept fixed during a single explicit or implicit smoothing iteration.

From differential geometry [6] we know that the mean-curvature vector $HN$ equals the Laplace Beltrami operator applied to the identity Id on a surface $\mathcal{M}$:

$$H(x)N(x) = -\Delta_{\mathcal{M}} x. \tag{2}$$

Thus geometric diffusion is equivalent to mean curvature motion ($MCM$)

$$\partial_t x = -H(x)N(x) \,, \tag{3}$$

where $H(x)$ is the corresponding mean curvature (here defined as the sum of the two principal curvatures), and $N(x)$ is the normal on the surface at point $x$. Already in '91 Dziuk [7] presented a semi implicit finite element scheme for $MCM$ on triangulated surface.

The mean curvature motion model is known as the natural local surface area decreasing flow. I.e., we obtain for the area $\mathrm{Ar}(\omega(t))$ of a subset $\omega(t)$ of a smooth surface $\mathcal{M}$ undergoing the $MCM$ evolution (cf. [9])

$$\frac{d}{dt}\mathrm{Ar}(\omega(t)) = -\int\limits_{\omega(t)} H^2 \,\mathrm{d}x \,. \tag{4}$$

This is one indication for the strong regularizing effect of $MCM$.

Unfortunately $MCM$ not only decreases the geometric noise due to unprecise measurement but also smoothes out geometric features such as edges and corners on the surface. Hence, we seek models which improve a simple high pass filtering.
In image processing Perona and Malik [16] proposed a nonlinear diffusion method, which modifies the diffusion coefficient at edges. Edges are indicated by steep intensity gradients. For a given initial image $\rho_0$ they considered the evolution problem

$$\partial_t \rho - \mathrm{div}(G(\|\nabla\rho\|)\nabla\rho) = 0 \,.$$

For increasing time $t$ - the scale parameter - the original image at the initial time is now successfully smoothed and image patterns are coarsened. But simultaneously edges are enhanced if one chooses a diffusion coefficient $G(.)$ which suppresses diffusion in areas of high gradients . A suitable choice for $g$ is

$$G(s) = \left(1 + \frac{s^2}{\lambda^2}\right)^{-1} \tag{5}$$

for a positive constant $\lambda$. Thus edges are classified by $\lambda$. I. e. sharpening by backward diffusion is invoked whenever $\|\nabla\rho\|_{\geq}\lambda$ whereas the image is smoothed by forward diffusion for $\|\nabla\rho\|_{<}\lambda$. Kawohl and Kutev [11] gave a detailed analysis of the diffusion types in this method. Unfortunately the above original Perona and Malik model is still ill-posed because there is a true backward diffusion in areas of large gradients. Catté et al. [2] proposed a regularization method where the diffusion coefficient is no longer evaluated on the exact intensity gradient. Instead they suggested to consider the gradient evaluation on a prefiltered image, i.e., they consider the equation

$$\partial_t \rho - \mathrm{div}(G(\|\nabla\rho_\epsilon\|)\nabla\rho) = 0 \tag{6}$$

where $\rho_\epsilon = G_\epsilon * \rho$ with a suitable local convolution kernel $G_\epsilon$. For instance we again take into accout the Gaussian filter kernel.

This model turns out to be well-posed, edges are still enhanced. The evolution and the prefiltering avoid the detection and pronouncing of artificial edges, which are due to the initial noise.

Weickert [22] improved this method taking into account anisotropic diffusion, where the Perona Malik type diffusion is concentrated in one direction, for instance the gradient direction of a prefiltered image. This leads to an additional tangential smoothing along edges and amplifies intensity correlations along lines. Preußer and Rumpf [18] took up this idea for the construction of streamline type patterns in flow fields.

Concerning the numerical implementation Weickert proposed finite difference schemes [21] and Kačur and Mikula [10] suggested a semi-implicit finite element implementation for the isotropic model by Catté et al. Large stencils have to be considered in case of the implementation of anisotropic diffusion by finite differences. This is a crucial shortcoming of such methods, especially if we consider a geometric counterpart on discrete surfaces. Preußer and Rumpf

[17] discussed adaptive finite element methods in 2D and 3D image processing by anisotropic nonlinear diffusion.

In the axiomatic work by Alvarez et al. [1], general nonlinear diffusion problems were introduced. More precisely they derive parabolic equations with elliptic terms which are based on the curvature of isolines or isosurfaces in images. Sethian and Malladi [15] presented a numerical level set method which also considers the curvature of level lines.

## 3 NOTATION AND GEOMETRIC SETTING

Before we develop our model of nonlinear geometric surface processing, let us first briefly review the basic notation of manifolds, differential calculus and geometric diffusion. For a detailed introduction to geometry and differential calculus we refer to [6] and [3, Chapter 1]. For the sake of simplicity we assume our surfaces to be compact embedded manifolds without boundaries. Thus we consider a smooth manifold $\mathcal{M}$, which we suppose to be embedded in $\mathbb{R}^3$. By $\{(x^\alpha, \Omega^\alpha)\}_\alpha$ we denote a countable atlas of $\mathcal{M}$, where $\Omega^\alpha \subset \mathbb{R}^2$ is an open reference domain and

$$x^\alpha : \Omega^\alpha \to \mathcal{M}\,;\; \xi^\alpha \mapsto x^\alpha(\xi^\alpha)$$

is the corresponding coordinate map. For each point $x$ on $\mathcal{M}$ a tangent space $\mathcal{T}_x\mathcal{M}$ is spanned by the basis $\{\frac{\partial}{\partial \xi_1^\alpha}, \frac{\partial}{\partial \xi_2^\alpha}\}$. We regard tangent vectors as linear functionals on $C^\infty(\mathcal{M})$, i. e. for $f \in C^\infty(\mathcal{M})$ we define $\frac{\partial}{\partial \xi_i^\alpha}(x) f := \frac{\partial f(x^\alpha)}{\partial \xi_i^\alpha}(\xi)$ where $x = x^\alpha(\xi)$.

Due to the embedding in $\mathbb{R}^3$ we identify $\frac{\partial}{\partial \xi_i^\alpha}$ with the tangent vector $\frac{\partial x^\alpha}{\partial \xi_i^\alpha}$. By $\mathcal{TM}$ we denote the tangent bundle. Integration on $\mathcal{M}$ requires the definition of a metric

$$g(\cdot,\cdot) : \mathcal{T}_x\mathcal{M} \times \mathcal{T}_x\mathcal{M} \to \mathbb{R}\,,$$

where $g = (g_{ij})_{ij}$ is supposed to be a quadratic positive definite form. In our embedded case we obtain

$$g_{ij} = g\Big(\frac{\partial}{\partial \xi_i^\alpha}, \frac{\partial}{\partial \xi_j^\alpha}\Big) = \frac{\partial x^\alpha}{\partial \xi_i^\alpha} \cdot \frac{\partial x^\alpha}{\partial \xi_j^\alpha}\,, \tag{7}$$

where $\cdot$ indicates the scalar product in $\mathbb{R}^3$. The inverse of $g$ is denoted by $g^{-1} = (g^{ij})_{ij}$.

Now we can define the integration of a function $f$ on $\mathcal{M}$. Let $\{\eta_\alpha\}_\alpha$ be a finite partition of unity on $\mathcal{M}$ with $\operatorname{supp}\eta_\alpha \subset \Omega^\alpha$, then

$$\int_{\mathcal{M}} f \,\mathrm{d}x := \sum_\alpha \int_{\Omega^\alpha} \eta_\alpha f(x^\alpha)\sqrt{\det g}\,\mathrm{d}\xi^\alpha\,. \tag{8}$$

Note that this definition is independent of the choice of the partition of unity. Integrating either a product of two functions $f$, $g$ on $\mathcal{M}$ or the product of two vector fields $v$, $w$ on $\mathcal{TM}$ we obtain the following scalar products on $C^0(\mathcal{M})$ and $C^0(\mathcal{TM})$ respectively:

$$(f,g)_{\mathcal{M}} := \int_{\mathcal{M}} fg \,\mathrm{d}x\,, \quad (v,w)_{\mathcal{TM}} := \int_{\mathcal{M}} g(v,w)\,\mathrm{d}x\,.$$

In the following we use the sloppy notation dropping the index $\alpha$. Next we proceed considering the fundamental differential operators on $\mathcal{M}$. Suppose $f \in C^1(\mathcal{M})$ then the total differential $df$ is a linear functional on $\mathcal{TM}$ ($df \in \mathcal{TM}'$), i. e. $\langle \frac{\partial}{\partial \xi_i}, df\rangle := \frac{\partial}{\partial \xi_i}(\xi)(f) = \frac{\partial (f\circ x)}{\partial \xi_i}(\xi)$. The gradient $\nabla_{\mathcal{M}} f$ of $f$ is defined as the representation of $df$ with respect to the metric $g$: $g\left(\nabla_{\mathcal{M}} f, \frac{\partial}{\partial \xi_i}\right) = \left\langle \frac{\partial}{\partial \xi_i}, df\right\rangle$. We obtain in coordinates

$$\nabla_{\mathcal{M}} f = \sum_{i,j} g^{ij}\,\frac{\partial (f\circ x)}{\partial \xi_j}\,\frac{\partial}{\partial \xi_i}\,. \tag{9}$$

Furthermore, we define the divergence $\operatorname{div}_{\mathcal{M}} v$ for a vector field $v \in \mathcal{TM}$ as the dual operator of the gradient by

$$\int_{\mathcal{M}} \operatorname{div}_{\mathcal{M}} v\,\phi\,\mathrm{d}x := -\int_{\mathcal{M}} g(v, \nabla_{\mathcal{M}}\phi)\,\mathrm{d}x$$

for all $\phi \in C_0^\infty(\mathcal{M})$.

Once we have introduced the gradient of a function on $\mathcal{M}$, we directly obtain the Dirichlet form on $\mathcal{M}$ by

$$(\nabla_{\mathcal{M}} u, \nabla_{\mathcal{M}} v)_{\mathcal{TM}} := \int_{\mathcal{M}} g(\nabla_{\mathcal{M}} u, \nabla_{\mathcal{M}} v)\,\mathrm{d}x\,.$$

Furthermore, the Laplace Beltrami operator $\Delta_{\mathcal{M}}$ applied to any function $u \in C^2(\mathcal{M})$ is given by the duality

$$(-\Delta_{\mathcal{M}} u, \phi)_{\mathcal{M}} := (\nabla_{\mathcal{M}} u, \nabla_{\mathcal{M}}\phi)_{\mathcal{TM}} \tag{10}$$

for all $\phi \in C^\infty(\mathcal{M})$. A simple computation leads to the following representation of $\Delta_{\mathcal{M}} u$ in local coordinates :

$$\Delta_{\mathcal{M}} u = \sum_{i,j} \frac{1}{\sqrt{\det g}}\frac{\partial}{\partial \xi_i}\left(g^{ij}\sqrt{\det g}\,\frac{\partial u}{\partial \xi_j}\right)\,.$$

In the anisotropic diffusion method to be presented in this paper we will make intensive use of some fundamental curvature quantities. We are now introducing the corresponding notation. Let us assume that $\mathcal{M}$ is orientable; then we have a well defined normal $N : \mathcal{M} \to S^2 \subset \mathbb{R}^3$ on $\mathcal{M}$. The second fundamental form $I\!I : \mathcal{T}_x\mathcal{M} \times \mathcal{T}_x\mathcal{M} \to \mathbb{R}$ is locally given by

$$I\!I(v,w) := \sum_{i,j} v^i w^j\, N_{,i}\cdot x_{,j}\,,$$

where the Gauss map $N : \Omega \to S^2$ is a representation of the normal with respect to a coordinate map $x : \Omega \to \mathcal{M}$ and $v = v^1\frac{\partial}{\partial \xi_1} + v^2\frac{\partial}{\partial \xi_2}$, $w = w^1\frac{\partial}{\partial \xi_1} + w^2\frac{\partial}{\partial \xi_2}$, are two tangent vectors in $\mathcal{T}_x\mathcal{M}$. We point out that $I\!I$ is well defined, i. e. it is invariant under reparametrization. Furthermore, $I\!I$ is a symmetric bilinear form and therefore there is a symmetric endomorphism $S : \mathcal{T}_x\mathcal{M} \to \mathcal{T}_x\mathcal{M}$ with

$$g(Sv, w) = I\!I(v,w)\,;\; v,w \in \mathcal{T}_x\mathcal{M}\,.$$

The endomorphism $S$ is called shape operator. Its eigenvalues are the principal curvatures of $\mathcal{M}$ at the point $x$ and the eigenvectors are the principal directions of curvature. Now, one can define notions of curvature as the *mean-curvature* $H$ and the *Gaussian curvature* $K$ by

$$H := \operatorname{tr} S\,, \quad K := \det\, S\,. \tag{11}$$

(Note that in our notation the mean-curvature $H$ is only the sum of the principal curvatures.) With the Laplace Beltrami operator at hand we can finally define a geometric diffusion problem in analogy to the linear diffusion problem in Euclidian space. We look for a solution $u : \mathbb{R}_0^+ \times \mathcal{M} \to \mathbb{R}$ of the parabolic equation $\partial_t u(t,x) - \Delta_{\mathcal{M}} u(t,x) = 0$ on $\mathbb{R}_0^+ \times \mathcal{M}$ for given initial data $u(0,\cdot) = u_0$.

Here, $u_0$ is some function on $\mathcal{M}$. Furthermore, we can consider a diffusion of the manifold geometry itself (cf. Section 2). I. e., we seek an one parameter family of embedded manifolds $\{\mathcal{M}(t)\}_{t\geq 0}$ and corresponding parametrizations $x(t)$, such that

$$\begin{aligned} \partial_t x(t) - \Delta_{\mathcal{M}(t)} x(t) &= 0\,, \\ \mathcal{M}(0) &= \mathcal{M}_0\,. \end{aligned} \tag{12}$$

For the sake of simplicity we define $MCM(\mathcal{M}_0, t) := \mathcal{M}(t)$, where $\mathcal{M}(t)$ is the solution surface at time $t$. Thus $MCM(\mathcal{M}, \sigma^2/2)$ can be regarded as the application of a "geometric " Gaussian filter of width $\sigma$ to $\mathcal{M}$.

By integration by parts, we obtain

$$(\partial_t x, \theta)_{\mathcal{M}(t)} + (\nabla_{\mathcal{M}(t)} x, \nabla_{\mathcal{M}(t)} \theta)_{\mathcal{T}\mathcal{M}(t)} = 0\,. \tag{13}$$

This is the corresponding weak variational formulation which holds for all test functions $\theta \in C^\infty(\mathcal{M}(t))$. The fundamental observation is that this geometric diffusion on the coordinate mapping itself coincides with the motion by mean curvature ($MCM$); in fact for any manifold $\mathcal{M}$ we have $\Delta_{\mathcal{M}} x = -H(x)\,N(x)$ as already stated above in Section 2.

# 4 ANISOTROPIC GEOMETRIC DIFFUSION

We are now prepared to discuss the concept of anisotropic geometric diffusion as a powerful multiscale method in surface processing. The aim is to appropriately carry over approved methodology from scale space theory in image processing. Let us first summarize the building blocks of the new method:

• We consider a noisy initial surface $\mathcal{M}_0$ to be smoothed and replace the linear diffusion from the Euclidian case of flat images by an appropriate diffusion of the surface geometry itself. The natural geometric diffusion process is curvature flow, which leads to an already nonlinear parabolic system of equations with initial data $\mathcal{M}_0$. Thereby a family of surfaces $\{\mathcal{M}(t)\}_{t\in\mathbb{R}_0^+}$ is generated, where the time $t$ serves as the scale parameter.

• In addition to the smoothing of the surface our aim is to maintain or even enhance sharp edges on the surface. Gradients of a coordinate mapping are not intrinsic objects on manifolds. The canonical quantity is the curvature tensor, in the case of embedded surfaces represented by the shape operator $S$. An edge is indicated by a sufficiently large eigenvalue. Hence we consider a diffusion tensor depending on $S$, which enables us to decrease diffusion significantly at edges indicated by $S$. Furthermore we will introduce a threshold parameter $\lambda$ as in (5) for the identification of edges.

• The evaluation of the shape operator on a noisy surface might be misleading with respect to the original but unknown surface and its edges. Thus we prefilter the current surface $\mathcal{M}(t)$ before we evaluate the shape operator. The straightforward "geometric Gaussian" filter is a short timestep of mean curvature flow. Hence, we compute a shape operator $S_\epsilon$ on the resulting prefiltered surface $\mathcal{M}_\epsilon(t)$, where $\epsilon$ is the "geometric Gaussian" filter. Let us emphasize that this choice also leads to a mathematically well-posed parabolic problem. Hence we avoid ill-posed backward diffusion in our model.

• With an appropriately chosen scalar diffusion coefficient $a_\epsilon$ depending on the eigenvalues $\kappa_{\epsilon,1}, \kappa_{\epsilon,2}$ of $S_\epsilon$, which are the principal curvatures of $\mathcal{M}_\epsilon(t)$, it is already possible to smooth in approximately flat surface areas and to enhance edges. Along these edges the surfaces $\mathcal{M}(t)$ still retains its noisy structure from $\mathcal{M}_0$ (cf. color-page, Fig. 6).We incorporate anisotropic diffusion now based on a proper diffusion tensor $a_\epsilon$ which enables tangential smoothing along edges. Thereby, the tangential edge direction on the tangent space $\mathcal{T}_x\mathcal{M}(t)$ is indicated by the principal direction of curvature corresponding to the subdominant principal curvature. The second, perpendicular direction is considered to be the actual sharpening direction. Color-page, Figure 6 clearly outlines the advantage of an anisotropic diffusion tensor.

• The resulting method leads to spatial displacement and the volume enclosed by $\mathcal{M}(t)$ is changed in the evolution. Selecting either a retrieving force towards the initial surface $\mathcal{M}_0$ or a constant forcing which leads to volume preservation, we can further improve our multiscale method.

We end up with the following type of parabolic 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 solve the anisotropic geometric evolution equation generalizing the system (12):

$$\partial_t x - \mathrm{div}_{\mathcal{M}(t)}(a_\epsilon\,\nabla_{\mathcal{M}(t)} x) = f \quad\text{on}\quad \mathbb{R}^+ \times \mathcal{M}(t), \tag{14}$$

and satisfy the initial condition $\mathcal{M}(0) = \mathcal{M}_0$.

Here, for every point $x$ on $\mathcal{M}(t)$ the diffusion tensor $a_\epsilon$ is supposed to be a symmetric, positive definite, linear mapping on the tangent space $\mathcal{T}_x\mathcal{M}$:

$$a_\epsilon(x) : \mathcal{T}_x\mathcal{M} \to \mathcal{T}_x\mathcal{M}\ .$$

Furthermore, $f$ represents the forcing on the right-hand side that maintains certain geometric properties of $\mathcal{M}(t)$.

Let us point out that an anisotropic timestep $\tau$ corresponds to a nonlinear $\delta$-filtering, where $\tau = \delta^2/2$. The corresponding variational formulation is given by

$$(\partial_t x, \theta)_{\mathcal{M}(t)} + (a_\epsilon\,\nabla_{\mathcal{M}(t)} x, \nabla_{\mathcal{M}(t)} \theta)_{\mathcal{T}\mathcal{M}(t)} = (f, \theta)_{\mathcal{T}\mathcal{M}(t)}\,, \tag{15}$$

for all $\theta \in C^\infty(\mathcal{M}(t))$. We can express the above equation not only in the variational form but also in coordinates. This formulation is as follows (cf. equation (10)):

$$\partial_t x - \sum_{i,j,k,l} \frac{1}{\sqrt{\det\,g}} \frac{\partial}{\partial\xi_i}\left(\sqrt{\det\,g}\;\; g^{ij} a_{\epsilon,jk} g^{kl} \frac{\partial x}{\partial \xi_l}\right) = f$$

on $\mathbb{R}^+ \times \mathcal{M}(t)$, where $a_{\epsilon,jk} = g\left(a_\epsilon \frac{\partial}{\partial\xi_j}, \frac{\partial}{\partial\xi_k}\right)$. We will nevertheless focus on the variational formulation - especially when we implement a suitable finite element algorithm.

In the simplest model we consider a scalar diffusion coefficient and set

$$a_\epsilon = G\left(\sqrt{\kappa_{\epsilon,1}^2 + \kappa_{\epsilon,2}^2}\right)\mathrm{Id},$$

for a function $G$ similar to the one which was introduced for the nonlinear diffusion on planar images (see (5)). We slightly modify the original definition of $G$ for the following reason. We already know that the fundamental $MCM$ evolves the surface only in normal direction (cf. equations (12) and (2) ). This is no longer true for an edge enhancing evolution as proposed here. It is a concentration effect in the coordinate mapping accompanied by some tangential displacement which allows the actual sharpening at an edge. Nevertheless we want to restrict any necessary tangential displacement in close vicinity to an edge and we intend to prescribe the standard $MCM$ evolution outside this area. Thus we define a generalization of (5)

$$G(s) = \begin{cases} 1 & ;\ |s| \leq \Theta\lambda \\ \left(1 + \frac{(|s|-\Theta\lambda)^2}{(1-\Theta)^2\lambda^2}\right)^{-1} & ;\ |s| > \Theta\lambda \end{cases}\ .$$

Here $\lambda$ serves as a threshold value for the identification of edges. If one of the principal curvatures is larger than $\lambda$, then we suppose there is an edge which we want to be sharpened. In the model without prefiltering this shows up in the steepening behaviour of backward diffusion for $|s| > \lambda$. Below a fraction $\Theta$ of this threshold $\lambda$ we set fast $MCM$ type smoothing without tangential displacement and in between $\Theta\lambda$ and $\lambda$ some suitable transition is prescribed. In our applications we always consider $\Theta = 0.5$.

As already announced, an improved model integrates tangential smoothing along edges into the multiscale approach. Therefore we consider a true diffusion tensor $a_\epsilon$ which is no longer restricted to multiples of the identity and which leads in such a way to a scalar diffusion coefficient. We introduce the prefiltering by mean-curvature flow as follows: $\mathcal{M}_\epsilon(t) := MCM(\mathcal{M}(t), \epsilon^2/2)$. At any point $x$ on the prefiltered surface $\mathcal{M}_\epsilon(t)$ the shape operator $S_\epsilon$ is a symmetric endomorphism $\mathcal{T}_x\mathcal{M}_\epsilon \to \mathcal{T}_x\mathcal{M}_\epsilon$. Thus there is an orthonormal basis $\{w^1, w^2\}$ of $\mathcal{T}_x\mathcal{M}_\epsilon$ such that $S_\epsilon$ is represented by

$$S_\epsilon = \left( \begin{array}{cc} \kappa_{\epsilon,1} & 0 \\ 0 & \kappa_{\epsilon,2} \end{array} \right) .$$

Now we consider a diffusion tensor in equation (14) which is defined with respect to the above orthonormal basis by

$$a_\epsilon = a(S_\epsilon) = \left( \begin{array}{cc} G(\kappa_{\epsilon,1}) & 0 \\ 0 & G(\kappa_{\epsilon,2}) \end{array} \right) \tag{16}$$

with the function $G$ from above. Hence, in a principal direction of curvature with curvature larger than $\lambda$ we enforce a signal enhancement. If the second principal curvature is smaller than $\lambda$ we regard the first direction as an orthogonal direction of an important edge on the surface which has to be sharpened. Simultaneously, in the other direction - the tangent direction along the edge - we allow smoothing. At corners both principal curvatures are large, thus sharpening takes place in both directions. Here, we again expect tangential shifting only if one of the principal curvatures is larger then $\Theta\lambda$.

In the simple case of $a_\epsilon = \mu\,\mathrm{Id}$, where $\mu$ is a smooth function, we can determine the tangential part of $\mathrm{div}_{\mathcal{M}(t)}(a_\epsilon\,\nabla_{\mathcal{M}(t)}x)$ explicitly. It is simply given by the differential $d\mu(x)$. Thus tangential displacement is just caused by the change of $\mu$.

With respect to a concrete implementation let us remark that we can interprete $a_\epsilon$ also as a mapping on the product of the embedded tangent space and the one dimensional space spanned by the normal. In order to do this we decompose a vector $z \in \mathbb{R}^3$ in the orthogonal basis $\{v^1, v^2, N\}$

$$z = (z \cdot v^1)v^1 + (z \cdot v^2)v^2 + (z \cdot N)N$$

where $v^1$, $v^2$ denote the embedded tangent vectors corresponding to $w^1$, $w^2$ (after projection and renormalization). Then using the definition of $a_\epsilon$ in equation (16), we have

$$a_\epsilon\, z = G(\kappa_{\epsilon,1})(z \cdot v^1)v^1 + G(\kappa_{\epsilon,2})(z \cdot v^2)v^2 + (z \cdot N)N\ . \tag{17}$$

Motivated by our definition of the diffusion tensor we define as a generalization of the mean curvature (see (11))

$$H_{a_\epsilon} := \mathrm{tr}(a_\epsilon \circ S)$$

as a generalization of the mean curvature (see (11)). This we will call $a_\epsilon$-mean curvature. Using this notion, we can express the changing-rate of the area $\mathrm{Ar}(\mathcal{M}(t))$ and the volume $\mathrm{Vol}(\mathcal{M}(t))$ enclosed by the compact surface $\mathcal{M}(t)$. Here $\mathcal{M}(t)$ is assumed to be the solution of the homogenous evolution problem $\partial_t x - \mathrm{div}_{\mathcal{M}(t)}(a_\epsilon\,\nabla_{\mathcal{M}(t)}x) = 0$:

$$\begin{array}{rcl} \dfrac{d}{dt}[\mathrm{Ar}(\mathcal{M}(t))]_{t=t_0} & = & -\displaystyle\int_{\mathcal{M}(t_0)} H\,H_{a_\epsilon}\ \mathrm{d}x\ , \\ \dfrac{d}{dt}[\mathrm{Vol}(\mathcal{M}(t))]_{t=t_0} & = & -\displaystyle\int_{\mathcal{M}(t_0)} H_{a_\epsilon}\ \mathrm{d}x\ . \end{array}$$

The first equation expresses one aspect of smoothing in our model. There is a significant regularization, indicated by the area minimization in areas which are expected to be flat with respect to the prefiltering. The second equation leads to a formulation of a flow which keeps the volume enclosed by the evolution surfaces $\mathcal{M}(t)$ constant (cf. Huisken [9] for the $MCM$ case). To achieve this we have to select a right-hand side which compensates the overall change in volume by a constant forcing in normal direction, i. e. we consider in equation (14)

$$f(t,x) = h(t)\,N(x)\ , \tag{18}$$

where $h(t) := \frac{1}{\mathrm{Ar}(\mathcal{M}(t))} \int\limits_{\mathcal{M}(t)} H_{a_\epsilon}\ \mathrm{d}x$. Alternatively, we can select a simple retrieving force

$$f(t,x) = C\,(x_0 - x(t)) \tag{19}$$

where $x_0$ is the original point location on the initial surface $\mathcal{M}_0$.

# 5 FINITE ELEMENT DISCRETIZATION

Up to now we have considered surfaces $\mathcal{M}$ which have been continuous manifolds. Concerning the implementation of the proposed multiscale method we now discretize our model. We use a finite element discretization to avoid large stencils as they would appear in our problem using e.g. finite difference schemes. Thus, we have to choose a spatial discretization and some timestepping scheme. To clarify the notation we will always denote discrete quantities with upper case letters to distinguish them from the corresponding continuous quantities in lower case letters. In the application surfaces are typically represented by triangular meshes. Hence, we suppose our meshes to be triangulations as well. Let us denote such a discrete surface $\mathcal{M}_h$. We are interested in a family of discrete successively smoothed and sharpened surfaces starting from some initially given noisy surface $\mathcal{M}_{h,0}$. We suppose them all to be equivalent with respect to a unique topological triangulation

$$\mathcal{T} = \{T_i \,|\, i \in I\}\ ,$$

where $I$ is some index set. For the sake of convenience, we identify a discrete surface and its triangulation. Here the subscript $h$ indicates the grid size, which we regard as a piecewise constant function on the current triangulation. Its value on a triangle is the length of the longest edge. On $\mathcal{T}$ and therefore also on $\mathcal{M}_h$ we define the space of piecewise linear continuous functions

$$V^h = \{\Phi \in C^0(\mathcal{M}_h) \,|\, \Phi|_{T_i} \in \mathcal{P}_1\, \forall i \in I\}\ ,$$

where $\mathcal{P}_1$ is the space of linear polynomials. The identity $\mathrm{Id}(\mathcal{M}_h)$ on the triangulation $\mathcal{M}_h$, which coincides with the pointwise coordinate vector $X$ can be regarded as a function in $(V^h)^3$. Here, we consider every reference map from a single reference triangle $\hat{T} \subset \mathbb{R}^2$ onto some $T_i$ on $\mathcal{M}_h$ as a coordinate map. Integration over $\mathcal{M}_h$ is defined in analogy to the continuous case (see definition (8)), now summing over local contributions on the triangles of the mesh. The metric and the gradients of functions on $\mathcal{M}_h$ are evaluated accordingly on each triangle $T$ with respect to the reference triangle $\hat{T}$.

Now we are able to formulate our discrete problem. Discretizing first only in space we obtain a variational formulation (cf. for the continuous case (14)) of an evolution problem for a family $\{\mathcal{M}_h(t)\}$ of discrete surfaces with coordinate maps $X(t)$. In analogy to (15) we obtain

$$\begin{array}{l} (\partial_t X(t), \Theta)^h_{\mathcal{M}_h(t)} + \\ (A_\epsilon \nabla_{\mathcal{M}_h(t)} X(t), \nabla_{\mathcal{M}_h(t)}\Theta)_{\mathcal{T}\mathcal{M}_h(t)} = (f(t), \Theta)^h_{\mathcal{M}_h(t)} \end{array}$$

for all test functions $\Theta \in (V^h)^3$ and $\mathcal{M}_h(0) = \mathcal{M}_{h,0}$. Here, we use the lumped mass scalar product $(\cdot,\cdot)^h_{\mathcal{M}_h}$, which is the geometric counterpart of the lumped masses for standard parabolic problems on domains in $\mathbb{R}^2$. It replaces the scalar product from the variational formulation of our problem in the continuous case and is defined by

$$(U,V)^h_{\mathcal{M}_h} := \sum_{T\in\mathcal{M}_h} \int_T \mathcal{I}_h(U\,V)\,\mathrm{d}x$$

for two discrete functions $U, V \in V^h$ (cf. [20]). Here, $\mathcal{I}_h : C^0(\mathcal{M}^n_h) \to V^h$ denotes the nodal projection operator. As an immediate consequence the mass matrix is diagonal. This simplifies the resulting scheme significantly. Our discrete anisotropic diffusion tensor $A_\epsilon$, to be defined later, is supposed to be an endomorphism on the discrete tangent space $\mathcal{T}\mathcal{M}_h(t)$ approximating $a_\epsilon$ from the continuous case.

We end up with a system of ordinary differential equations for the three coordinates of all triangulation nodes.

Next, we have to discretize in time, which includes the choice of some timestepping scheme and the decision which terms to be handled implicitly and which explicitly. Here we proceed in analogy to the approach presented by Dziuk [7] for the discretization of mean curvature motion. We expect $X^n$ to be an approximation of $X(n\tau)$ where $\tau$ is the selected timestep. Hence, the time derivative can be discretized applying a backward Euler scheme. Thus we obtain:

*Find a sequence of discrete coordinate maps* $\{X^n_h\}_{n=0,\cdots}$ *which defines a family of triangular surfaces* $\{\mathcal{M}^n_h\}_{n=0,\cdots}$ *such that*

$$\begin{aligned}&(\frac{X^{n+1}-X^n}{\tau},\Theta)^h_{\mathcal{M}^n_h} + \\ &(A^n_\epsilon \nabla_{\mathcal{M}^n_h} X^{n+1}, \nabla_{\mathcal{M}^n_h}\Theta)_{\mathcal{T}\mathcal{M}^n_h} = (F^n,\Theta)^h_{\mathcal{M}^n_h}.\end{aligned}$$

*for all discrete test functions* $\Theta \in (V^h)^3$.

In what follows we explain in detail the above notation and discuss at which timestep to evaluate functions, metric and diffusion tensor. As the governing metric we always consider the one at the old timestep, here indicated by the subscript $\mathcal{M}^n_h$, $\mathcal{T}\mathcal{M}^n_h$ respectively. Thus, also the gradients $\nabla_{\mathcal{M}^n_h}$ are considered with respect to the metric on $\mathcal{T}\mathcal{M}^n_h$. Furthermore, the diffusion tensor $A_\epsilon$ is evaluated explicitly at time $t_n$, which we indicate by an upper index $n$. Concerning the right hand side, we evaluate $f$ at the old time $t_n$ and define $F^n = f(t_n)$.

Finally, in each step of the discrete evolution we have to solve a single system of linear equations. In terms of nodal vectors, which we indicate by a bar on top of the corresponding discrete function we can rewrite the scheme and get

$$(M^n + \tau L^n(A^n_\epsilon))\bar{X}^{n+1} = M^n\bar{X}^n + \tau M^n \bar{F}^n \qquad (20)$$

for the new vertex positions $\bar{X}^{n+1}$ at time $t_{n+1} = (n+1)\tau$. Here, we assume the lumped mass matrix

$$M^n = \Big((\Phi_i,\Phi_j)^h_{\mathcal{M}^n_h}\Big)_{ij}$$

and the nonlinear stiffness matrix

$$L^n(A^n_\epsilon) = \Big((A^n_\epsilon \nabla_{\mathcal{M}^n_h}\Phi_i, \nabla_{\mathcal{M}^n_h}\Phi_j)_{\mathcal{T}\mathcal{M}^n_h}\Big)_{ij}$$

to be applied simultaneously to each of the three coordinate components. If in addition a nonvanishing right-hand side is considered, we have to evaluate the vector $\bar{F}^n$ representing the right-hand side on each node $X$, i. e. $(\bar{F}^n)_i = F^n(X^n_i)$. Here, we either choose in case of the retrieving force $F^n(X) = C(X^0 - X)$, which has to be evaluated only at the nodes of the current triangulation due to the selected lumped mass integration formula, or the constant force $F^n(X) = \frac{1}{|\mathcal{M}^n_h|}\int_{\mathcal{M}^n_h} \mathrm{tr}(A^n\,S^n_\epsilon)\,\mathrm{d}x\,N(X)$ for an interpolated normal $N(X)$, which corresponds to the area preserving evolution in the continuous model (compare the forces in (18) and (19)).

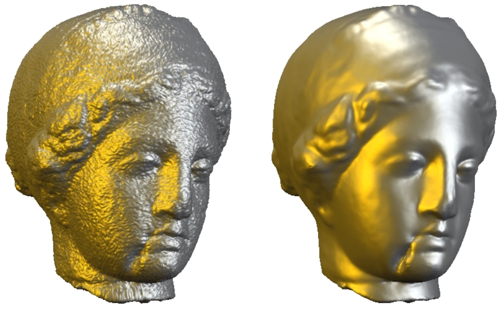

Figure 2: *The noisy initial image and a timestep (t=0.0002) from the evolution of a venus head dataset (the diameter is scaled to* 1*).*

The diffusion tensor $A^n_\epsilon$ is supposed to be piecewise constant on the triangles of our mesh. As in the continuous model the evaluation of the diffusion tensor has to be based on a regularized, prefiltered surface. Here we consider a single timestep of the discrete mean curvature evolution as a geometric appropriate regularization, i. e. we compute

$$\bar{X}^n_\epsilon = (\mathrm{Id} + \frac{\epsilon^2}{2}M^n L^n(\mathrm{Id}))^{-1}\bar{X}^n \qquad (21)$$

where $L^n(\mathrm{Id})$ is the above stiffness matrix for the isotropic diffusion tensor Id. Then the corresponding coordinate map $X^n_\epsilon$ defines a discrete surface $\mathcal{M}^n_{h,\epsilon}$. Thus, we have filtered the probably noisy initial coordinates with a "geometric" and discrete Gaussian filter of width $\epsilon$ before we identify edges to be enhanced by our actual discrete evolution. Hence, high frequency noise is suppressed and we obtain well-posed discrete problems whose asymptotic behaviour is independent of the grid size. Let us emphasize that we apply this geometric filter only to evaluate the diffusion tensor and not as an evolution step of the surface itself. The required solver for this smoothing problem is already available by a slight modification of our original scheme for a single timestep.

In the continuous model a suitable construction of a diffusion tensor, which incorporates edge sharpening and tangential smoothing along edges, involves the principal curvatures and curvature directions deduced from the shape operator. Now in the discrete case, we are interested in some discrete counterpart. At first, triangulated surfaces have no canonical curvature tensor. For every triangle $T$ the curvature evaluation is based on local $L^2$-projections of the triangulated and regularized surface $\mathcal{M}^n_{h,\epsilon}$ onto graphs of quadratic polynomials over the tangent space. In our case the embedded tangent space coincides with the plane containing $T$. For these polynomials the corresponding shape operator $S^n_\epsilon$ can be computed explicitly. For details we refer to Section 6. Finally, we evaluate the required diffusion tensor $A^n_\epsilon$ on every triangle $T$ by (see (16))

$$A^n_\epsilon|_T = a(S^n_\epsilon)\ ,$$

where we take up our original definition (cf. Section 4) for the continuous problem and apply it now to the numerically approximated shape operator $S^n_\epsilon$.

Color-page, Figure 7 shows results from the semi-implicit algorithm for a venus head data set. The corresponding initial data is

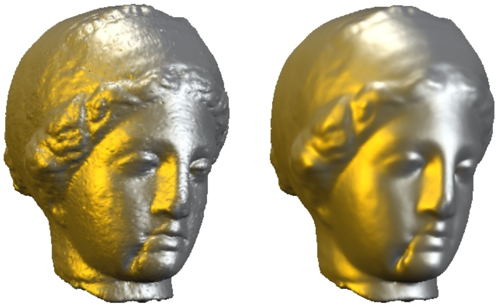

Figure 3: *The discrete solutions at time* $t = 0.00005$ *are calculated for different values of the prefilter width (*$\epsilon = 0.005$ *(left) and* $\epsilon = 0.08$ *(right)).*

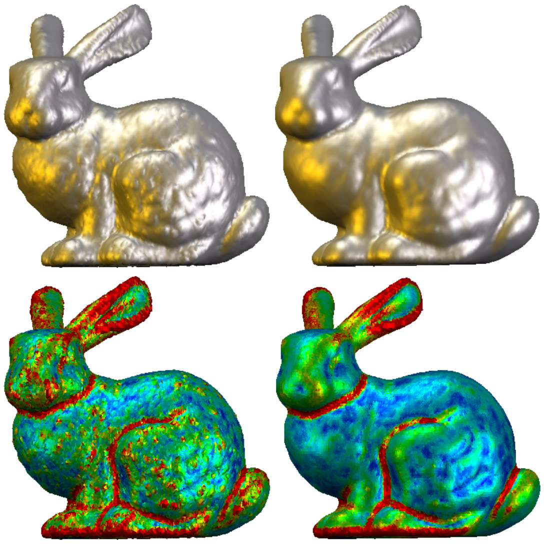

Figure 4: *For* $\lambda = 10$ *(top left) and* $\lambda = 30$ *(top right) the discrete solutions are shows at time* $t = 0.0001$ *(the dominant principal curvature is depicted in color,* $\lambda = 10$ *(bottom left) and* $\lambda = 30$ *(bottom right)).*

displayed in Fig. 2. Figure 3 gives a comparison of the evolution results at time $t = 0.0004$ for different prefilter width $\epsilon$. Finally, we compare in Fig. 4 the dependance of the solution on the parameter $\lambda$. For smaller values of $\lambda$ more and more feature edges are enhanced. Here we consider the data set from Fig. 1 (cf. also color-page), with initial diameter 1 of the object.

## 6 CURVATURE ON A DISCRETE SURFACE

Let $T$ be an element of a triangulation $\mathcal{M}_h$ with vertices $P^1, P^2, P^3$ and barycenter $C_T = \frac{1}{3}(P^1 + P^2 + P^3)$. The normal of $T$ is denoted by $N$. First we assume a normalized situation such that $C_T = 0$ and $N = (0, 0, 1)$, i.e. $P^1, P^2, P^3 \in \{z = 0\}$, where points in $\mathbb{R}^3$ are denoted by $(x, y, z)$. This can be obtained by a rotation with some matrix $Q \in SO(3)$ and a translation.

In the following we will assume that $\omega_T = \{T' \in \mathcal{M}_h \,|\, T' \cap T \neq \emptyset\}$ is given as a graph over the plane $\{z = 0\}$. The image of the projection of $\omega_T$ on $\{z = 0\}$ is denoted by $\hat\omega_T$. Then the graph $\omega_T$ is represented by a piecewise linear function $\varphi : \hat\omega_T \to \mathbb{R}$. Now we compute the $L^2(\hat\omega_T)$-projection of $\varphi$ onto the subspace $\mathcal{Q}$ of the space of quadratic polynomials $\mathcal{P}_2$ spanned by $\{x^2, xy, y^2\}$. To this end, let $p(x, y) = \alpha x^2 + \beta xy + \gamma y^2$ be this $\mathcal{P}_2$-function, characterized by $\Xi = (\alpha, \beta, \gamma) \in \mathbb{R}^3$ and let $\varphi_1 = x^2, \varphi_2 = xy, \varphi_3 = y^2$ be the canonical basis of $\mathcal{Q}$. Then $\Xi$ is given by the linear equation

$$\sum_j \left( \int_{\hat\omega_T} \varphi_i \varphi_j \, dxdy \right) \Xi_j = \int_{\hat\omega_T} \varphi \varphi_i \, dxdy,\; i = 1, 2, 3\,.$$

The discrete principal curvatures $\kappa_1, \kappa_2$ and the principal directions of curvature of $\mathcal{M}_h$ on $T$ will be defined as the corresponding quantities for the smooth surface $(x, y, p(x, y))$ in the origin $0$.

The matrix representing the shape operator $S(0)$ in the basis $\{\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\}$ is given by:

$$\begin{pmatrix} 2\alpha & \beta \\ \beta & 2\gamma \end{pmatrix}.$$

For the mean-curvature $H$ and the Gaussian curvature $K$ in the point $0$ we obtain

$$\begin{aligned} H &= 2\ (\alpha + \gamma) \\ K &= 4\alpha\gamma - \beta^2\,. \end{aligned}$$

This enables us to compute $\kappa_1, \kappa_2$ as follows:

$$\kappa_{1,2} = \frac{H}{2} \pm \sqrt{\frac{H^2}{4} - K}.$$

Color-page, Figure 7 shows the numerically calculated dominant principal curvature for the timesteps of an anisotropic evolution.

Then we easily obtain the embedded principal directions of curvature $V^1, V^2$. If $\alpha = \gamma$ and $\beta = 0$, one can choose as $\{V^1, V^2\}$ any orthonormal basis of the tangential space. Otherwise let $W^1 = (W^1_1, W^1_2),\ W^2 = (W^2_1, W^2_2)$ be non-zero solutions of

$$\begin{aligned} \left(\alpha - \gamma - \sqrt{H^2/4 - K}\right)\ W^1_1 + \beta\, W^1_2 &= 0 \\ \left(\alpha - \gamma + \sqrt{H^2/4 - K}\right)\ W^2_2 + \beta\, W^2_1 &= 0\,. \end{aligned}$$

In a last step we normalize the vectors $W^1$, $W^2$ and compute the embedded principal directions of curvature $V^1$, $V^2$ by a push forward onto the embedded tangent space applying the inverse of the rotation $Q$ from our normalization step above

$$V^1 = \frac{Q^{-1}}{\sqrt{(W^1_1)^2 + (W^1_2)^2}} \begin{pmatrix} W^1_1 \\ W^1_2 \\ 0 \end{pmatrix}, \qquad (22)$$

$$V^2 = \frac{Q^{-1}}{\sqrt{(W^2_1)^2 + (W^2_2)^2}} \begin{pmatrix} W^1_1 \\ W^1_2 \\ 0 \end{pmatrix}.$$

Note that $V^1$, $V^2$ are the discrete analogues of $v^1$, $v^2$ in Section 4. In our considerations we obtain a regularization by prefiltering the corresponding surface. We would like to point out that choosing larger stencils instead of $\omega_T$ is a further possibility for a regularization.

# 7 ALGORITHM AND IMPLEMENTATION

Based on our description of the method of anisotropic diffusion in surface processing let us now focus on the concrete implementation of the corresponding scheme for a single timestep of the discrete evolution problem. Due to the considered semi-implicit approach the algorithm is easy to code following the standard finite element procedure. Here we will give a detailed explanation, especially to outline how the solution of the discrete problem breaks up into elementary and mainly local calculation steps.

Each timestep consists of the assembly of the matrices, the solution of the prefiltering problem, the evaluation of the diffusion tensor and finally the solution of the actual diffusion problem:

```
timestep($\mathcal{M}^n$) {
  compute the prefiltered surface $\mathcal{M}^n_{h,\epsilon}$ solving
    $(M^n + \frac{\epsilon^2}{2} L^n(\mathrm{Id}))\bar{X}^n_\epsilon = M^n \bar{X}^n$;
  calculate $S^n_\epsilon$ and $A^n_\epsilon$;
  compute $\mathcal{M}^{n+1}_h$ solving for the nodal coordinates
    $(M^n + \tau L^n(A^n_\epsilon))\bar{X}^{n+1} = M^n \bar{X}^n$;
}
```

The assembly of each matrix $M^n$, $L^n(A^n_\epsilon)$ or $L^n(\mathrm{Id})$ - here denoted by $B$ - is based on the standard Finite Element procedure. It consists of an initialization $B = 0$ followed by a traversal of all surface triangles $T$. On each $T$ with nodes $P^0, P^1, P^2$, a corresponding local matrix $(b^T_{ij})_{ij}$ is computed first. It corresponds to all pairings of local nodal basis functions. Then the matrix entries are added to the matching entries in the global matrix $B$, i. e. for every pair $i, j$ we update $B_{\alpha(i),\alpha(j)} = B_{\alpha(i),\alpha(j)} + b^T_{ij}$. Here $\alpha(i)$ is defined as the global index of the node with local index $i$. Thus we can now focus on the computation of the local mass matrix $m^T$ and the local stiffness matrices $l^T(A)$ and $l^T(\mathrm{Id})$ respectively. Due to the applied lumped mass integration we immediately verify

$$m^T_{ij} = \frac{1}{3}\delta_{ij}|T|$$

where $|T|$ is the area of the triangle $T$ and $\delta_{ij}$ the usual Kronecker symbol.

Next, let us consider for every triangle $T$ the reference triangle $\hat{T} \subset \mathbb{R}^2$ with independent variables $\xi_1, \xi_2$ and nodes $\xi^0 = (0,0)$, $\xi^1 = (1,0)$, and $\xi^2 = (0,1)$. Then an affine coordinate map $X$ maps $\hat{T}$ onto $T$ and its nodes $\xi^i$ onto the corresponding nodes $P^i$ of $T$ on the discrete surface in $\mathbb{R}^3$. Hence the corresponding metric tensor is as in the continuous case (7) given by

$$g_{ij} = \frac{\partial X}{\partial \xi_i} \cdot \frac{\partial X}{\partial \xi_j} \ ,$$

where $\frac{\partial X_k}{\partial \xi_i} = P^i_k - P^0_k$. Now we are able to evaluate the gradients of the linear basis functions $\Phi^l$ corresponding to the nodes $P^l$ in the embedded tangent space spanned by $P^1 - P^0$ and $P^2 - P^0$, i. e. (cf. equation (9))

$$\nabla_{\mathcal{M}^n_h}\Phi^l = \sum_{i,j} g^{ij}\frac{\partial \Phi^l}{\partial \xi_j}(P^i - P^0) \ ,$$

where we refer to the derivatives of $\Phi^l$ with respect to the reference coordinates $\xi$:

$$\begin{pmatrix} \frac{\partial \Phi^l}{\partial \xi_1} \\ \frac{\partial \Phi^l}{\partial \xi_2} \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \ .$$

Thus for the linear stiffness matrix required in the presmoothing step we obtain $l^T_{ij} = |T|\nabla_{\mathcal{M}^n_h}\Phi^i \cdot \nabla_{\mathcal{M}^n_h}\Phi^j$. If our actual diffusion tensor $A_\epsilon$ on $T$ diagonalizes with respect to an orthogonal basis $\{V^1, V^2\}$ with entries $g_1, g_2$ on the diagonal in the corresponding representation (see definition (16)), for the local nonlinear stiffness matrix we finally obtain

$$\begin{aligned} l^T_{ij}(A) = \quad & |T| \quad [g_1(\nabla_{\mathcal{M}^n_h}\Phi^i \cdot V^1)(\nabla_{\mathcal{M}^n_h}\Phi^j \cdot V^1) + \\ & g_2(\nabla_{\mathcal{M}^n_h}\Phi^i \cdot V^2)(\nabla_{\mathcal{M}^n_h}\Phi^j \cdot V^2)] \ . \end{aligned}$$

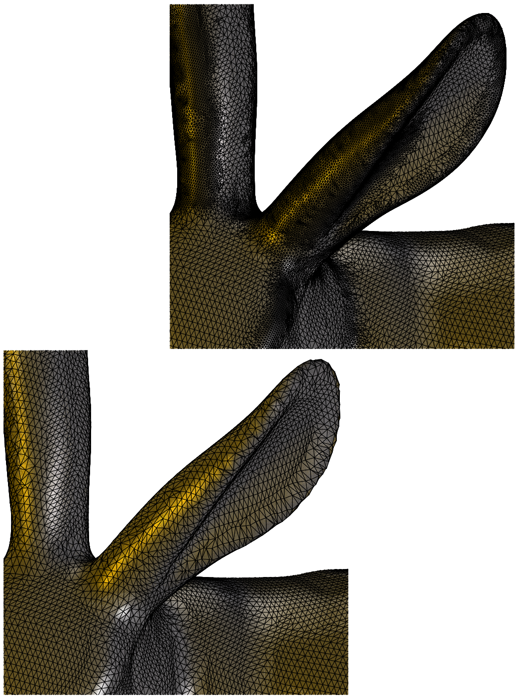

Figure 5: *The resulting surface mesh (top) from an adaptive algorithm is compared with the corresponding grid of an evolution with fixed triangulation (bottom). The areas of high curvature are clearly visible as refinement regions.*

## Grid Adaptivity

Due to the possible tangential displacement in the vicinity of edges the results of the evolution can be significantly improved taking into account adaptive grid refinement depending on the dominant principal curvature (cf. Fig. 5). Here, we apply a Delauney type refinement.

## Iterative Solver

The resulting systems of linear equations, which arise in each timestep of the discrete anisotropic curvature evolution are solved either by a *preconditioned conjugate gradient method* or by an *algebraic multigrid method*. For small timesteps and moderately fine meshes the conjugate gradient method converges in several iterations in case of a diagonal preconditioning. For instance for the venus data set and timestep $5 \cdot 10^{-5}$ the relative residual in $l^2$ norm drops below $10^{-12}$ after 4 iterations.

If we consider larger timesteps and fine grids the condition of the matrix $O(1 + \tau h^{-2})$ becomes large. Thus the number of required iterations increases significantly. Unfortunatly standard multigrid strategies are not available on arbitrary meshes. But applying an algebraic multigrid solver $O(1)$ iterations are required to solve the system independent of the grid size and the timestep. In the application we obtain a reduction of the residual in each timestep by a factor of $0.5$.

## 8 COMPARISON AND CONCLUSIONS

We have presented a novel multiscale technique for surface fairing. It is able to successively smooth noisy initial surfaces while simultaneously enhancing edges and corners on the surface. The evolution time is the scale parameter.

The method is based on an anisotropic curvature evolution problem. The corresponding nonlinear partial differential equations have been discretized by finite elements in space and a semi implicit backward Euler scheme in time. The user controls the surface evolution mainly by two parameters which have an intuitive meaning. A regularization parameter $\epsilon$ has to be chosen to filter out high frequency noise before the diffusion coefficient is evaluated. Here a suitable choice in the application is $\epsilon = Ch$ with $C \in [1, 4]$. Furthermore, $\lambda$ can be regarded as a user given threshold for edge detecting, with the meaning that a principal curvature larger than $\lambda$ indicates an edge which is to be preserved by the fairing scheme.

Previous work on surface fairing already involves the idea of curvature motion. Taubin [19] and Kobbelt [12] considered an umbrella operator, which is a “spring force type” implementation of the Laplace Beltrami operator. The shortcoming of tangential shifts in their work is mainly due to the successive local change of the metric in the interation scheme itself. Furthermore, due to the explicit character of the scheme, timestep limitations show up. Desburn et al. [5] avoided both shortcomings by considering an implicit scheme which holds the metric fixed and is unconditionally stable. Sharp edges on the surface are rapidly smoothed by all previous approaches because of the high local curvature which leads to fast smoothing in these regions. Our method is able to detect such edges and their direction and incorporates appropriate direction dependent smoothing only. Concerning multigrid methods for the smoothing, Kobbelt et al. [13] discuss a V cycle type smoothing with straightforward prolongation and restriction, where we propose a true algebraic multigrid which involves appropriate matrix dependent prolongations and restrictions.

Interesting future research directions are

• the combination of the presented multiscale method with multiresolutional techniques, which should appropriately reflect the continuous coarsening in the evolution,

• further investigations on surface modeling concerning suitable choices of the diffusion tensor and the forcing on the right hand side of the parabolic system, and

• the simultaneous processing of the geometry and the texture if such an additional texture is given on the surface.

**Acknowledgement**

The authors thank Alexander Schweitzer from Bonn for helping with the test of the algebraic multigrid and offering his code.

## References

[1] L. Alvarez, F. Guichard, P. L. Lions, and J. M. Morel. Axioms and fundamental equations of image processing. *Arch. Ration. Mech. Anal.*, 123:199–257, 1993.

[2] F. Catté, P. L. Lions, J. M. Morel, and T. Coll. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, 29:182–193, 1992.

[3] I. Chavel. *Eigenvalues in Riemannian Geometry*. Academic Press, 1984.

[4] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Computer Graphics (SIGGRAPH '96 Proceedings)*, pages 303–312, 1996.

[5] M. Desbrun, M. Meyer, P. Schroeder, and A. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Computer Graphics (SIGGRAPH '99 Proceedings)*, pages 317–324, 1999.

[6] M. P. do Carmo. *Riemannian Geometry*. Birkhäuser, Boston–Basel–Berlin, 1993.

[7] G. Dziuk. An algorithm for evolutionary surfaces. *Numer. Math.*, 58:603–611, 1991.

[8] I. Guskov, W. Sweldens, and P. Schroeder. Multiresolution signal processing for meshes. In *Computer Graphics (SIGGRAPH '99 Proceedings)*, 1999.

[9] G. Huisken. The volume preserving mean curvature flow. *J. Reine Angew. Math.*, 382:35–48, 1987.

[10] J. Kačur and K. Mikula. Solution of nonlinear diffusion appearing in image smoothing and edge detection. *Appl. Numer. Math.*, 17:47–59, 1995.

[11] Kawohl, B. and Kutev, N. Maximum and comparison principle for one-dimensional anisotropic diffusion. *Math. Ann.*, 311 (1):107–123, 1998.

[12] L. Kobbelt. Discrete fairing. In *Proceedings of the 7th IMA Conference on the Mathematics of Surfaces*, pages 101–131, 1997.

[13] L. Kobbelt, S. Campagna, J. Vorsatz, and H.-P. Seidel. Interactive multi-resolution modeling on arbitrary meshes. In *Computer Graphics (SIGGRAPH '98 Proceedings)*, pages 105–114, 1998.

[14] W. Lorensen and H. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163–169, 1987.

[15] R. Malladi and J. A. Sethian. Image processing via level set curvature flow. *Proc. Natl. Acad. Sci. USA*, 92:7046–7050, 1995.

[16] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. In *IEEE Computer Society Workshop on Computer Vision*, 1987.

[17] T. Preußer and M. Rumpf. An adaptive finite element method for large scale image processing. In *Scale-Space Theories in Computer Vision*, pages 232–234, 1999.

[18] T. Preußer and M. Rumpf. Anisotropic nonlinear diffusion in flow visualization. In *Proceedings Visualization 1999*, 1999.

[19] G. Taubin. A signal processing approach to fair surface design. In *Computer Graphics (SIGGRAPH '95 Proceedings)*, pages 351–358, 1995.

[20] V. Thomée. *Galerkin - Finite Element Methods for Parabolic Problems*. Springer, 1984.

[21] J. Weickert. *Anisotropic diffusion in image processing*. Teubner, 1998.

[22] Weickert, J. Foundations and applications of nonlinear anisotropic diffusion filtering. *Z. Angew. Math. Mech.*, 76:283–286, 1996.