

# Constraint-based fairing of surface meshes

Klaus Hildebrandt Konrad Polthier

Freie Universität Berlin

---

## Abstract

We propose a constraint-based method for the fairing of surface meshes. The main feature of our approach is that the resulting smoothed surface remains within a prescribed distance to the input mesh. For example, specifying the maximum distance in the order of the measuring precision of a laser scanner allows noise to be removed while preserving the accuracy of the scan.

The approach is modeled as an optimization problem where a fairness measure is minimized subject to constraints that control the spatial deviation of the surface. The problem is efficiently solved by an active set Newton method.

---

## 1. Introduction

The instant availability of high-quality digital models of 3D surfaces becomes an essential prerequisite in many research areas, industrial modeling, e-commerce, medical treatment planning, archeology, and restoration, just to name a few. Acquisition technologies such as laser scans for 3D surface models or CT, MRI and other devices for volumetric shapes can measure data with high accuracy. Nevertheless the accuracy is often lost in the mesh creation pipeline. A crucial step in this pipeline is the removal of geometric noise contained in the positions of the measured points. Many techniques to effectively remove the noise have been proposed, but these may spoil the accuracy of the data.

Our new method provides the guaranty that after noise removal the surface still lies within the accuracy of the measured data. This approach is designed for applications where accuracy is crucial; for example, technical or medical applications as well as digitalization of cultural heritage.

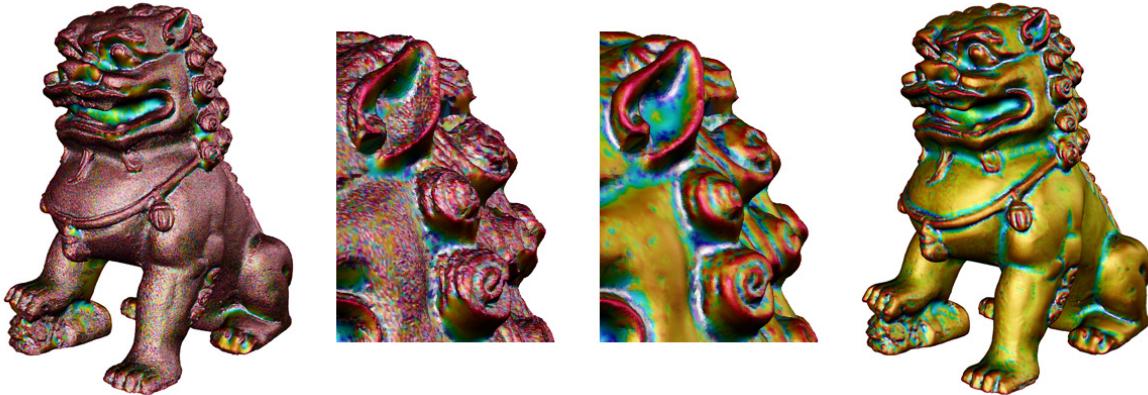
**Measuring fairness.** Fairness energies are an attempt to establish quantitative measures for fairness of a shape. Finding a commonly-accepted measure of fairness is a delicate task, due to the inherent subjectivity of rating the appearance of a geometry as well as the specific demands of applications. Nevertheless, one can agree on some general criteria: a fairness energy should be independent of the parametrization of the surface, invariant under rigid motions and scaling, and spheres should be among the minimizers of the energy.

Different measures of fairness have been proposed. These can be classified by the order of the highest derivative of the

surface needed to evaluate the energy. A measure of first order is the area of the surface. Since area is not invariant under scaling, Delingette [Del01] proposed using the isoperimetric ratio  $A^3/V^2$  as a scale invariant first-order fairness measure for closed surfaces. Here  $A$  denotes the area and  $V$  is the volume enclosed by the surface. Second-order measures relate to curvature, such as integrals of squares of curvature terms. Prominent examples are the bending energy  $\int H^2 dA$ , the total curvature  $\int (\kappa_1^2 + \kappa_2^2) dA$ , and the Willmore energy  $\int (\kappa_1 - \kappa_2)^2 dA$ . Here  $\kappa_1$  and  $\kappa_2$  denote the principal curvatures and  $H = \kappa_1 + \kappa_2$  the mean curvature. An example of a third-order measure is the curvature variation energy proposed by Moreton and Sequin [MS92]. The Euler Lagrange equation of this energy is of sixth order where minimization becomes a delicate task; especially on meshes.

**Improving fairness.** Evolution methods are a technique to improve the fairness of a surface. Such evolving surfaces monotonically decrease a fairness measure and are described as the solution of a (usually non-linear) parabolic partial differential equation. The  $L^2$ -gradient flow of the area functional evolves each point of the surface along its normal direction with a velocity equal to the mean curvature and is therefore called the mean curvature flow. The underlying flow equation is of diffusion type  $\partial_t X = \Delta X$ , where  $X$  is a family of immersions of the surface and  $\Delta$  is the Laplace-Beltrami operator of  $X$ . Smoothing algorithms based on this equation are often referred to as Laplace smoothing.

Taubin [Tau95] applied this approach to surface smoothing using a linearization of the flow equation, which keeps the Laplace operator unchanged during the evolution. Based



**Figure 1:** An original noisy scan of a Chinese lion with 1.3 m triangles and height about 10 cm (left). The smoothed mesh (right) stays within a 0.1 mm distance from the initial mesh. The surfaces are colored by mean curvature, with color range from white (negative curvature) to red (positive curvature).

on this linear equation, he constructed a low-pass filter for meshes in analogy to filters used in signal processing. Desbrun et al. [DMSB99] applied the mean curvature flow to mesh smoothing and used the more faithful *cotan* discretization of the Laplace-Beltrami operator. Furthermore, they employed an implicit scheme for the time integration to overcome the step-size restrictions of explicit schemes.

**Feature preserving schemes.** A side effect of Laplace smoothing is that it quickly smoothes out geometric features such as sharp corners. Anisotropic diffusion schemes [CDR00, TWBO02, BX03, HP04] preserve or even enhance sharp corners by suppressing diffusion in directions of high curvature. Anisotropic geometric diffusion derives from the Perona Malik filter [PM87] in image processing. A related technique - the bilateral filter [TM98] - has been transferred to meshes by Fleishman et al. [FDCO03] and Jones et al. [JDD03]. Recently, an extension employing non-local means has been presented by Yoshizawa et al. [YBS06]. Comparable results can be achieved by methods based on Wiener filtering [PSZ01, GP01, Ale02].

**Second order fairness measures.** While minimizers of the area functional are determined by values at the boundary, second order fairness measures such as bending energy or Willmore energy allow for  $C^1$  boundary conditions; i.e. positions and normals can be prescribed. This makes second-order functionals attractive for the construction of fair spline surfaces in geometric design [WN01]. By specifying consistent data at the boundaries of the individual patches, globally  $G^1$  surfaces can be constructed. Kobbelt and Schneider [SK00] construct fair meshes with  $G^1$  boundary conditions as the solution of a fourth-order differential equation.

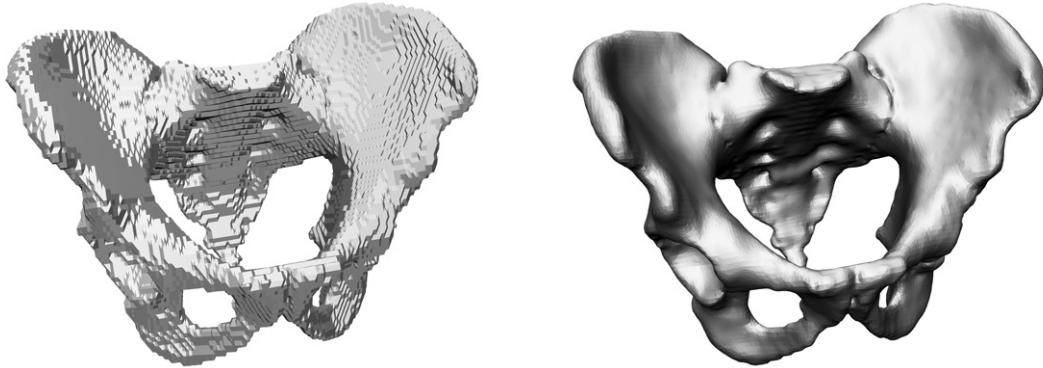
For meshes, stable discretizations of the Willmore energy and its fourth-order gradient flow have been developed in

recent years. Yoshizawa and Belyaev [YB02] describe a direct discretization of the flow equation using the cotan formula to discretize terms related to mean curvature and the angle defect for Gauss curvature. Clarenz et al. [CDD\*04] present a finite element discretization using linear Lagrange elements. The basis of their scheme is a decomposition of the weak formulation of the flow equation into a coupled system of second-order equations in which the Gauss curvature term, which is difficult to discretize, is removed. Bobenko [Bob05] constructs a discrete Willmore functional that preserves the Möbius group, the symmetry group of the continuous energy. Bobenko and Schröder [BS05] present an implicit scheme to integrate the gradient flow of this energy.

The bending energy plays an essential role in the simulation of cloth. Common edge-based discretization schemes [BW98, BMF03, GHDS03, WBH\*07] measure mean curvature at the edges of the mesh [CSM03, HP04]. Such a discrete energy is given as a sum of contributions from stencils that consist of two triangles sharing an interior edge. The small stencil reduces the complexity of the expressions for the energy and its gradient, which in turn accelerates the evaluation and simplifies the implementation.

**Constrained fairing.** To assess the quality of the output of a smoothing method, additionally to fairness, the deviation of the result from the initial surface (e.g. measured data) is an important criterion. Belyaev and Othake [BO03] use integrals of the squared distance and the squared deviation of normals to compare the results produced by different smoothing methods.

Least-square meshes proposed by Sorkine et al. [SCO04, NISA06] minimize a weighted sum of two quadratic energies, the biharmonic energy and a weighted sum of the squared distances of each vertex to its initial position. Ex-



**Figure 2:** Removing marching cubes artifacts from a human pelvis (~28 cm width) model extracted from computed tomography data. The maximum deviation of the smoothed mesh (right) from the initial surface is 1 mm.

tending this approach Volodine et al. [VVR06] present a scheme to compute minimizers of the biharmonic energy under the constraint that the sum of the squared distances is less than a prescribed value. Since for larger meshes the computation of the exact solution of this problem is too costly, they describe a scheme to approximate the solution. Liu et al. [LBH<sup>\*</sup>01] propose a different type of constraint: the positions of the barycenters of all triangles are fixed. This is very restrictive; for example, if noise is added to a planar mesh, the vertices and the barycenters move out of the plane. Since the barycenters keep their position during smoothing, the plane cannot be recovered.

Gibson [Gib98] describes a technique to reduce aliasing effects in isosurfaces extracted from binary volume data. In the first step an approximate isosurface, which has all vertices located at points of the volume grid, is extracted. Then an energy is minimized under the constraint, so that each vertex of the surface mesh remains within the eight voxels adjacent to the initial position of the vertex. Since the energy - a spring energy with zero rest length - is of first order, sharp bendings appear where the surface touches the constraints. Gibson notes that smoother surfaces could be produced using an energy that involves curvature, thus it is of second order. Nielson et al. [NGH<sup>\*</sup>03] present a related technique using a volume grid to define constraints as well. The scheme can process meshes, but requires the surface to surround a volume. First, a binary field is constructed that has value one at grid points inside the volume surrounded by the mesh and zero outside. Next, the isosurface corresponding to the value 0.5 is extracted. Each vertex of this surface is located at the midpoint of an edge of the volume grid. Then an energy is minimized with the constraint that each vertex remains on the corresponding edge of the volume grid. The energy is the sum over integrals of the squared curvature of curves that are generated by intersection of the mesh with planes. The sum runs over all coordinate planes that contain grid points.

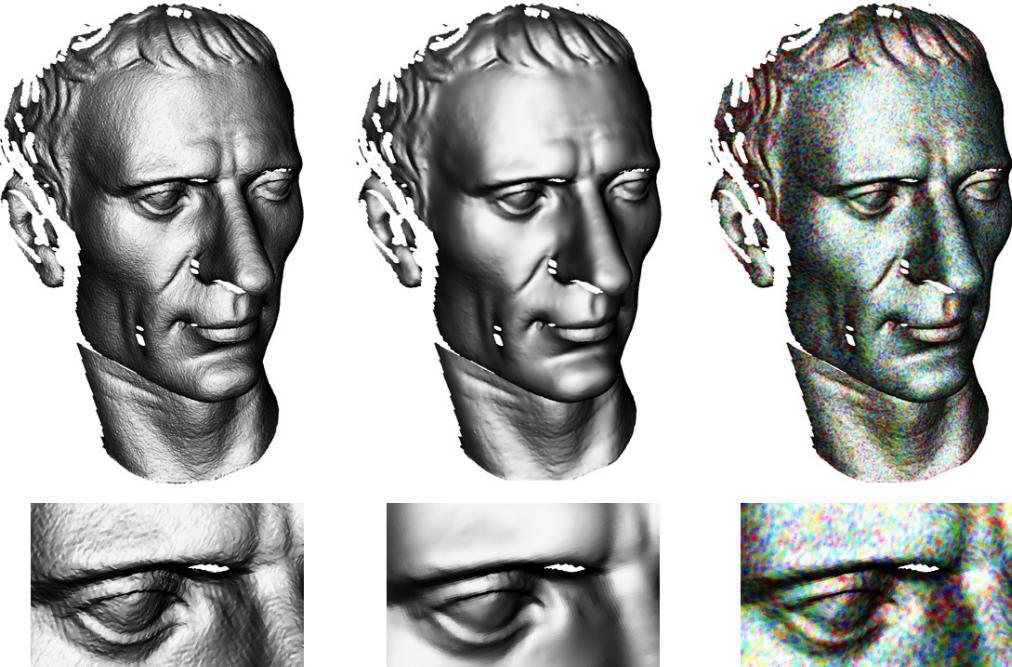
### 1.1. Contributions

We present a new constrained-based denoising and fairing algorithm which delivers high-quality fairing and a guaranty that all points of the smoothed surface remain within a prescribed distance to their initial positions. Our method is based on a novel combination of box constraints with a discrete bending energy and an efficient optimization procedure.

The scheme is modeled as a constrained non-linear optimization problem that is solved by an active set Newton method with gradient projection. We present a robust and efficient approximation scheme for the second-order information required by the Newton method. The computational cost of the method is comparable to that of implicit integration schemes for geometric flows.

Applications of our scheme include denoising of scan data with the guaranty that no point of the smoothed surface deviates further than the accuracy of the scanning device from its measured position, as well as removal of aliasing and ter-racing artifacts of isosurfaces extracted from volumetric data assuring that the surface remains within the domain consisting of the voxels that contain the initial surface and their 1-neighbors. This scheme yields a new level of control over surface smoothing that can be useful in the process of converting measured data into high-accuracy models for technical or medical applications as well as for the digitization of cultural heritage.

A property of common evolution based smoothing methods is that if they are not stopped at the right time, the model is oversmoothed or even degenerates. In contrast, additional iterations of the presented method can only improve the result. Furthermore, there is only one parameter to control the method, the maximum deviation tolerance, which for noise removal relates to the accuracy of the data. We believe that these are important steps towards an automatization of the smoothing process.



**Figure 3:** Noise removal from a range image. The maximum tolerance is 0.1 mm and the height of the object is 30 cm. The right image shows the smoothed surface colored by the distance of the initial (noisy) to the smoothed surface.

## 2. Discrete Fairness Energy

The fairness energy used in the optimization problem must be at least of second order, otherwise the surface would develop sharp bendings or staircasing artifacts at the constraints. Three examples of second-order energies, the bending energy, the Willmore energy, and the total curvature are closely related and differ only by a multiple of the integral of the Gauss curvature over the surface. By the Gauss-Bonnet theorem, this integral is constant for compact closed surfaces since it depends only on the topology of the surface. For compact surfaces with boundary the integral of the Gauss curvature is invariant under variations that fix positions and normals at the boundary. Hence in both cases the three energies have the same minimizers.

Either of the three energies is a good candidate for our purposes. An alternative second-order fairness measure would be the integral of the squared Gauss curvature. But by Gauss' Theorema Egregium this energy only depends on the metric of the surface, which rules out this energy. For example, take a piece of paper and bend it. There is no metric distortion; Gauss curvature does not change.

We use the discrete bending energy proposed by Wardetzky et al. [WBH<sup>\*</sup>07], which uses the edge-based mean curvature vector [HP04] and can be derived using non-

conforming Crouzeix-Raviart finite elements. The energy is a sum of contributions from hinge stencils

$$E(M) = \sum_{e \in M} \frac{3\|e\|^2}{2A_e} \cos\left(\frac{\theta_e}{2}\right)^2, \quad (1)$$

where  $A_e$  is the combined area of the two triangles adjacent to the edge  $e$ , and  $\theta_e$  is the dihedral angle at  $e$ . The non-linear gradient of the energy can be computed efficiently; for an explicit formula see [WBH<sup>\*</sup>07].

## 3. Constrained Optimization Problem

Our fairing method is modeled as a constrained non-linear optimization problem. The constraints assure that during the smoothing process, no point of the surface deviates more than a prescribed distance from its initial position. The definition of the constraints involves a deviation measure that takes into account the maximum deviation rather than the usual integral of the squared distance. We first define the discrete maximum deviation measure, then describe the optimization problem.

For two meshes  $M$  and  $N$  that have the same connectivity, we define the discrete *maximum deviation measure*  $d_\infty$  as

$$d_\infty(M, N) = \max_i \|m_i - n_i\|_{\mathbb{R}^3}, \quad (2)$$



**Figure 4:** Noise removal from the scanned blade model with 390 k triangles. Details are shown on the right.

where  $m_i$  and  $n_i$  are the vertices of  $M$  respectively  $N$ . The measure  $d_\infty$  describes a metric on the set of all meshes with a fixed connectivity. Let  $B_\varepsilon(M)$  denote the closed ball with radius  $\varepsilon$  around a mesh  $M$  with respect to this metric. Geometrically speaking, a mesh  $N$  is in the set  $B_\varepsilon(M)$  if each vertex  $n_i$  of  $N$  lies in the Euclidean ball of radius  $\varepsilon$  around the corresponding vertex  $m_i$  of  $M$ .

Now we can state the optimization problem. *Given a mesh  $M$  and a positive  $\varepsilon$ , find a mesh  $N$  that minimizes the discrete bending energy over all meshes  $\tilde{N} \in B_\varepsilon(M)$ .*

### 3.1. Discrete Deviation Measure

Consider two surfaces described by continuous functions  $f, g$  mapping a domain  $A$  into  $\mathbb{R}^3$ ; the (continuous) maximum deviation measure is then

$$\delta_\infty(f, g) = \sup_{x \in A} \|f(x) - g(x)\|_{\mathbb{R}^3}.$$

The objective of this section is to relate the discrete maximum deviation measure  $d_\infty$  to its continuous counter-

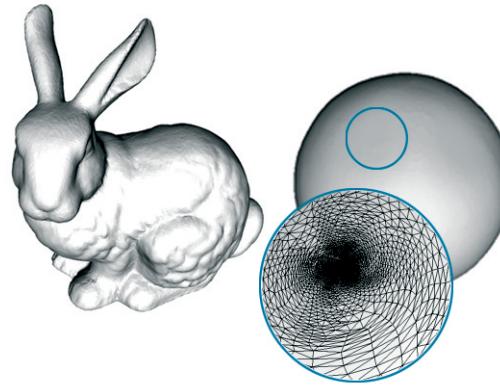
part  $\delta_\infty$ . At first the maximum deviation measure is only defined for parametrized surfaces. Since the maps  $f$  and  $g$  are only required to be continuous, the measure  $\delta_\infty$  can be evaluated for meshes as well. Let the domain  $A$  be a mesh embedded in  $\mathbb{R}^3$ . Then every mesh  $M$  in  $\mathbb{R}^3$  with the same connectivity as  $A$  can be parametrized by the continuous function  $I_M : A \rightarrow \mathbb{R}^3$  mapping each vertex of  $a_i$  of  $A$  to the coordinates of the corresponding vertex  $m_i$  of  $M$  and linear interpolation in the triangles. For two such functions  $I_M$  and  $I_N$  representing meshes  $M$  and  $N$  the continuous deviation measure  $\delta_\infty(I_M, I_N)$  equals the supremum of the function  $\|I_M - I_N\|_{\mathbb{R}^3}$ . The supremum is attained at a vertex of the mesh, hence for meshes the continuous measure equals the discrete measure

$$\delta_\infty(I_M, I_N) = d_\infty(M, N).$$

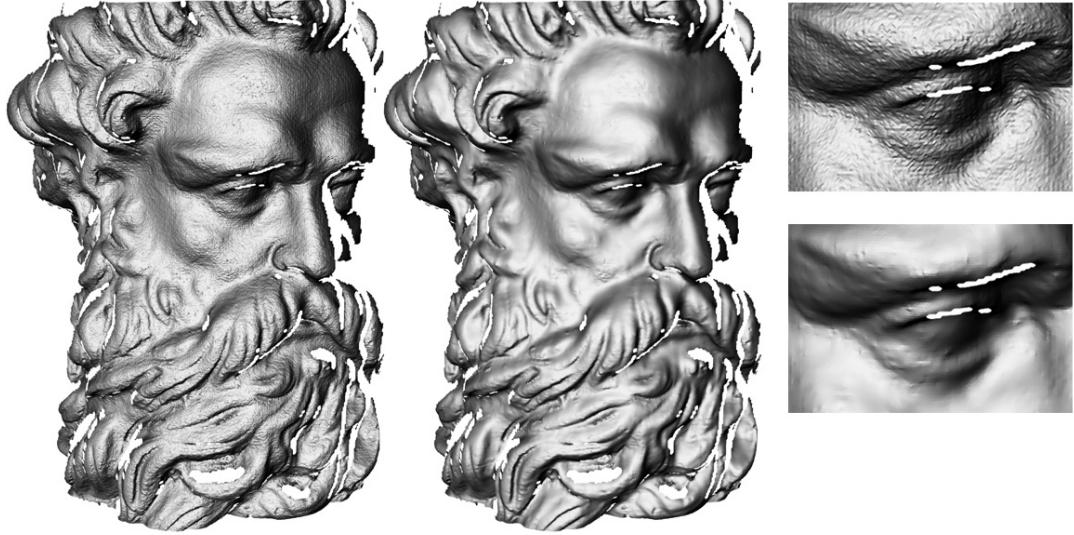
### 4. Minimization Procedure

The constraints we are dealing with are spheres around all vertices of the mesh. In order to simplify the minimization procedure we approximate the spheres by boxes with side length  $2\varepsilon$ . *Active set Newton solvers* with gradient projection are among the most effective solvers for large-scale box constrained non-linear optimization problems. For a general introduction to active set and gradient projection methods we refer to [NW06]. For an analysis and discussion of solvers of this type for large-scale problems see [LM99, Kan01] and references therein. First, we briefly describe the general scheme and describe in Section 4.1 how to approximate the required second-order information in an efficient and robust way.

By listing the coordinates of all vertices we identify a mesh with a point in  $\mathbb{R}^n$  where  $n$  equals three times the number of vertices in the mesh. Let  $x^0 \in \mathbb{R}^n$  be the point representing the initial mesh. Then the feasible set  $\Omega_{x^0, \varepsilon}$  is the  $n$ -dimensional cube with edge length  $\varepsilon$  and midpoint  $x^0$ . The projection  $P$  of a point  $x$  in  $\mathbb{R}^n$  onto  $\Omega_{x^0, \varepsilon}$  can be computed by projecting each coordinate  $x_i$  of  $x$  onto the interval  $[x_i^0 - \varepsilon, x_i^0 + \varepsilon]$ , where  $x_i^0$  is the  $i$ -th coordinate of  $x^0$ . During the minimization process we search for (usually non-unique) minima along projected rays. The idea is that whenever a bound is encountered, the direction of the ray is bent such



**Figure 5:** The bunny model (70k triangles) is shown on the left and after unconstrained minimization of the discrete bending energy on the right. The zoom shows the difference in size of the triangles corresponding to the region around an ear. This demonstrates the robustness of the discrete energy.



**Figure 6:** Noise removal from a range image of a part of a Neptune statue (height ~10 cm) with a maximum deviation tolerance of 0.04 mm.

that the ray remains in the feasible set. The projected ray starting at a point  $x$  with initial direction  $v$  is obtained by projecting the ray  $x + t v$  onto the feasible set, i.e. it is given by

$$r(t) = P(x + t v).$$

This describes a piecewise linear path that, after first hitting a face of the feasible set, leads along the boundary. Note that for any  $t$  this can be evaluated by a simple loop over all coordinates.

The *set of active constraints* for a point  $x \in \Omega_{x^0, \varepsilon}$  is defined as

$$A(x) = \{i \in \{1, 2, \dots, n\} \mid \|x_i - x_i^0\| = \varepsilon\}.$$

For a point away from the boundary the active set is empty, and for a vertex of the cube  $\Omega_{x^0, \varepsilon}$  all constraints are active. Let  $E : \Omega_{x^0, \varepsilon} \rightarrow \mathbb{R}$  be our energy and let  $H(x)$  denote the Hessian and  $g(x)$  the gradient. The Newton direction  $v(x)$  at a point  $x \in \Omega_{x^0, \varepsilon}$  is

$$v(x) = -(H(x))^{-1} g(x).$$

The active set method iterates the following two steps. Let  $x^i$  be the current position.

**1. Cauchy step.** Compute the gradient direction  $g(x^i)$ . Find a point  $y^i$  along the projected path  $P(x^i + t g(x^i))$  that produces a sufficient reduction of energy; i.e. that fulfills

$$E(y^i) < E(x^i) - \lambda \langle g(x^i), y^i - x^i \rangle, \quad (3)$$

where  $\lambda < 1$  is a constant. In our experiments we always set  $\lambda = 0.01$ . The last term in the equation uses the gradient to compute how much decrease of energy we can expect when moving in direction  $y^i - x^i$ . The equation guarantees that the actual choice of  $y^i$  produces a decrease that is at least within a fraction of the estimate which is important to guarantee rapid decrease of the energy.

**2. Subspace minimization.** The point  $y^i$  lies in some face of the feasible cube  $\Omega_{x^0, \varepsilon}$ . In this step we perform a minimization step constrained to this face. More specifically: determine the active set  $A(y^i)$  of  $y^i$  and compute the reduced Hessian  $H_r$  and the reduced gradient  $g_r$  with respect to the free variables. Here reduced means: for all  $i \in A(y^i)$  remove the  $i$ -th column and row from  $H$  and the  $i$ -th entry from  $g$ . Then compute the reduced Newton direction  $v_r^i = H_r^{-1} g_r$ . The next position  $x^{i+1}$  is a point on the projected path  $P(y^i + t v_r^i)$  that satisfies the condition

$$E(x^{i+1}) < E(y^i) \quad (4)$$

ensuring a decrease of the energy.

A local minimum is reached if the gradient either vanishes or is orthogonal to the boundary.

In steps 1 and 2 we search for a *good* point on a projected ray  $r(t)$ . Such a point is found by an iterative procedure, which starting from an initial guess  $t = t^0$  either monotonically increases or monotonically decreases the value of  $t$ . It is increasing if the guessed point  $r(t^0)$  fulfills the descent condition (eq. (3) in step 1 and eq. (4) in step 2) or

otherwise decreasing. In each iteration the value of  $t$  is multiplied by a factor of 2 (increasing case) or 0.5 (decreasing case). The iteration terminates if the descent condition is no longer fulfilled (increasing case) or if it is fulfilled (decreasing case). In the case of an increasing sequence the last value satisfying the descent condition is used. In each Cauchy step (respectively Newton step) we use the result of the previous iteration as an initial guess for  $t$ .

#### 4.1. Approximation of the Hessian

Since the Hessian of the discrete bending energy is not necessarily positive definite, there is no guaranty that the Newton direction is indeed a descent direction. One way to deal with this issue is to replace the Hessian by a positive definite approximation. Note that we only modify the search direction but still minimize the same energy. This technique is used in inexact Newton methods.

To approximate the Hessian of the bending energy we use the Hessian of the thin plate energy with the initial surface as parameter domain. This matrix has the form  $SM^{-1}S$ , where  $S$  is the usual cotan matrix [PP93, DMSB99], and  $M$  contains the masses; e.g.  $M$  is a diagonal matrix containing the Voronoi areas. Since the Laplace matrix  $S$  has a kernel consisting of the constant functions we add the constant value 0.1 to all diagonal entries of  $S$ . The resulting approximation of the Hessian is positive definite, can be written as the square of a matrix, and decouples the  $x$ ,  $y$ , and  $z$  coordinates. Therefore the linear systems that need to be solved have better condition (the condition number is only the square root of the original), higher sparsity and lower dimension. This allows the approximate Newton direction to be computed and, in effect, the method to be applied to larger models. This is demonstrated on the Chinese lion model with 1.3 m triangles in Figure 1.

A related approximation of the Hessian of the bending energy has been proposed and used to accelerate the integration of Willmore flow in [WBH<sup>\*</sup>07]. A main difference is that the approximate Hessian we use is a product of square matrices, whereas their approximation is a product of non-square matrices.

**Efficiently solving the linear system.** We need to solve the linear system

$$(SM^{-1}S)_r v_r = g_r, \quad (5)$$

where  $g$  is the gradient of the discrete bending energy eq. (1). Note that we solve only the reduced system, indicated by the subindex  $r$ . The variables in the active set remain fixed; the corresponding rows and columns are removed from the system. It is not necessary to completely solve the system; approximating the solution by performing only a few iterations of preconditioned conjugate gradients is usually more efficient for large-scale problems. These type of methods are called Newton-CG methods. The performance of the method

essentially depends on the choice of a good preconditioner. We compute a sparse Cholesky factorization of  $S$  only once as a preprocessing step and use the factorization in each iteration to build a *constraint preconditioner* [NW06] for solving the reduced system (5).

## 5. Experimental Results

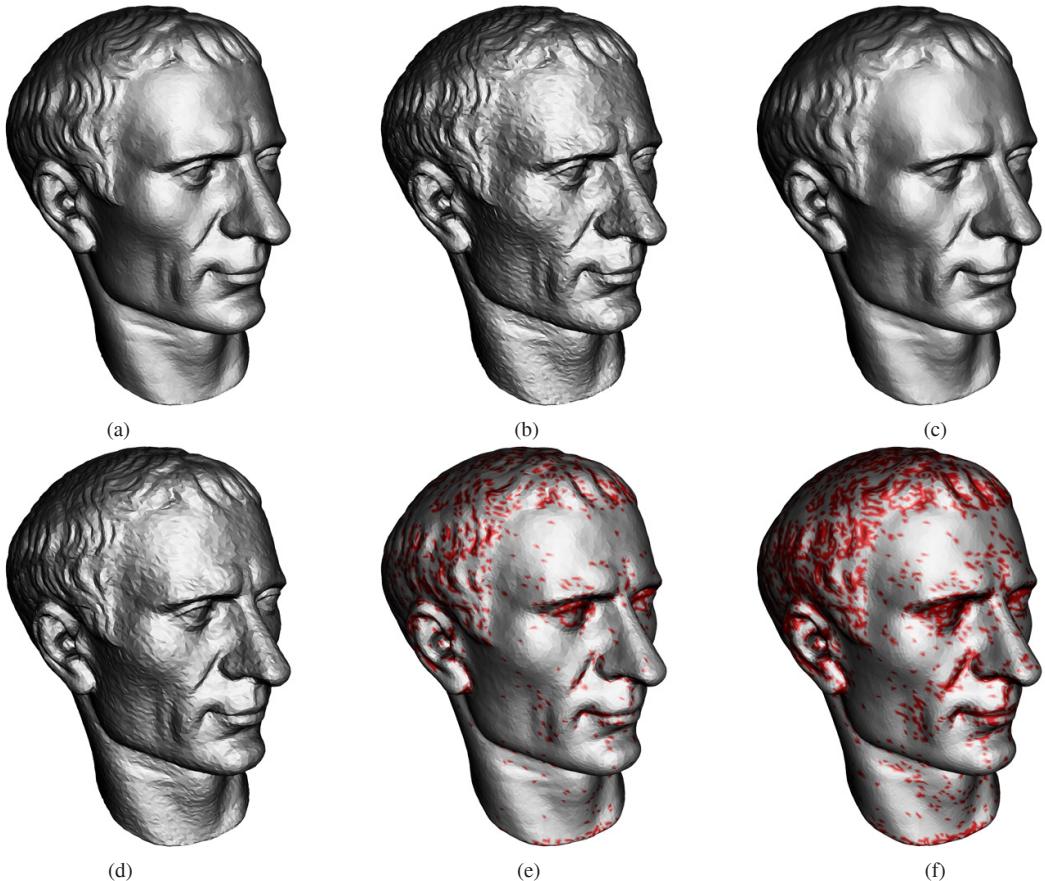
We have tested our method on laser-scanned real-world models, except for the pelvis model shown in Figure 2 that has been extracted from computed tomography data. No artificial noise has been added to the models, we tested the method with the noise inherent to the scanned data. The only exceptions are Figures 7 and 8 where artificial noise has been added to the models in order to have a goal surface for the comparison of methods.

Noise removal from range images is shown in Figures 3 and 6. These models have a regular quad connectivity which we triangulated in order to get planar faces. The two range images differ in size and resolution: the image of the Caesar model has 190 k triangles and a height of 30 cm whereas the Neptune model has 270 k triangles and a height of 10 cm. Accordingly, we specified a smaller value for the maximum deviation tolerance for Neptune than for Caesar, 0.04 mm for Neptune and 0.1 mm for Caesar. These values are much smaller than the size of features like the eyelids or wrinkles, which are preserved. The third image of Figure 3 shows the smoothed Caesar mesh colored by the distance of the initial (noisy) to the smoothed surface. The colors seem to be randomly distributed, which indicates that mainly noise has been removed and little structured deviation, like shrinkage of features, took place. The Figures 1 and 4 show models with clean connectivity and filled holes. The Chinese lion model has 1.3 m triangles and a height of 10 cm and the blade has 390 k triangles and measures about one meter in height. We used 0.1 mm maximum deviation tolerance for the Chinese lion and 1 mm for the blade.

The pelvis model (Figure 2) suffers from artifacts caused by a rudimentary volume-extraction method. The  $\varepsilon$  maximum deviation for this model is 1 mm which suffices to produce a smooth model.

We compare our method to the feature-preserving anisotropic diffusion as described in [HP04] on the armadillo model in Figure 8. Despite the larger amount of noise used in this example, our method produces a result that comes close to the original surface. The anisotropic diffusion focusses on preserving the features and even enhances them; e.g. the teeth are sharper than in the original model. On the other hand the noise is also preserved for a longer time. At the instant when the noise is removed, the non-feature regions are already oversmoothed. For the goal of reproducing the original shape, our method produces better results.

We compare our method to the unconstrained gradient



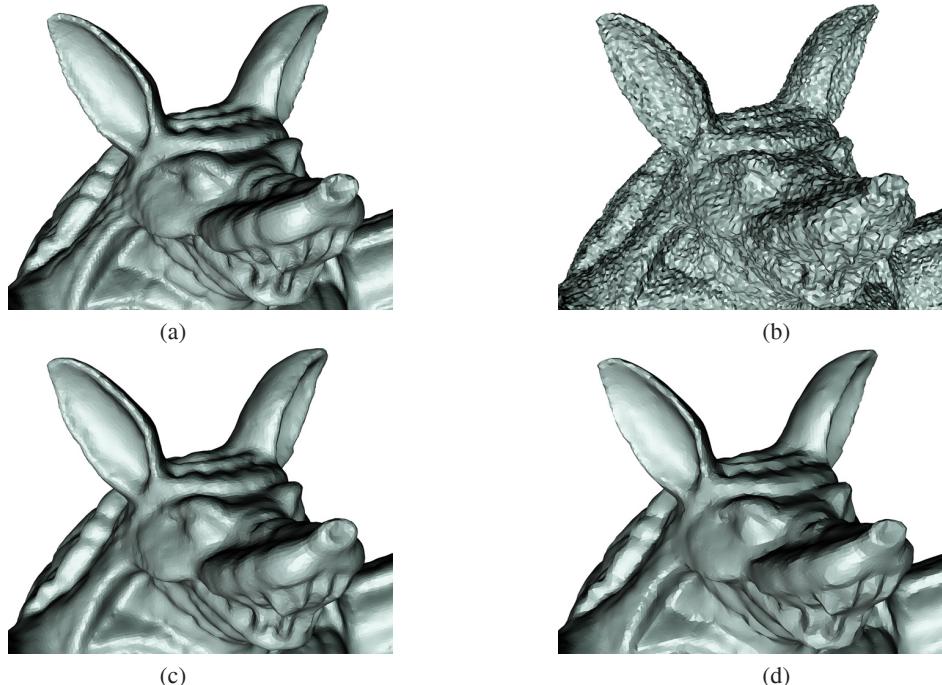
**Figure 7:** Comparison of the constrained optimization and the unconstrained gradient flow of the bending energy on the Caesar model (a). The model has been artificially corrupted by random noise with 0.2 mm maximum deviation (b), where the height of the object is 30 cm. The constrained optimization (c) keeps all vertices within 0.2 mm distance. The bottom row shows the gradient flow at different times. Vertices that leave the 0.2 mm distance are colored red. The first vertices leave the 0.2 mm distance (d). When the same bending energy is reached (e) the maximum distance is 2 mm and shrinkage effects appear for example at the eyelids. Further time steps (f) induce more shrinkage.

flow of the bending energy on the Caesar model in Figure 7. The model has been corrupted by random noise with a maximal deviation of 0.2 mm, where the height of the object is 30 cm. The constrained optimization (top row right) keeps all vertices within 0.2 mm distance to the noisy mesh. The bottom row shows snapshots of the unconstrained evolution. Vertices that leave the 0.2 mm distance are colored red. The first image shows the moment when the first vertices leave the 0.2 mm distance. The model is still very noisy. When the fairness energy equals the fairness energy of the surface produced by the constrained optimization, the maximum distance is already 2 mm and shrinkage effects appear in regions with high curvature, e.g. at the eyelids. Further time steps induce more shrinkage.

Shrouds [NGH<sup>\*</sup>03] allows the smoothing of surface meshes with spatial constraint that are defined with the help

of a volume grid. Our method offers different improvements over this scheme. Shrouds is only defined for meshes that enclose a volume, whereas our method can process arbitrary surface meshes with or without boundary. The maximum deviation used for denoising scan data in our examples is very small; e.g. 0.1 mm for an object that has a bounding box with edge length 100 mm. This means that the volume grid that Shrouds generates would have about  $10^9$  grid points. The mesh that Shrouds extracts from the volume grid would be much larger than the initial mesh. Additionally, since the optimization method uses a variant of an explicit gradient descent, one expects a large number of iterations.

**Optimization procedure.** The active set Newton scheme produces a rapid decrease of energy. All presented examples, including the 1.3 m triangles Chinese lion model, were stopped after 10-25 iterations. The most time-consuming op-



**Figure 8:** On the armadillo model (a) that has been artificially corrupted with random noise (b) a comparison of our method (c) with feature preserving anisotropic diffusion (d) is shown. The anisotropic flow sharpens features, but regarding the reproduction of the original model the result of our method is much closer.

eration in each iteration is the computation of the Newton direction, which requires solving a linear system. In comparison to this, the procedure that finds the new position along a projected ray  $r(t)$  is fast.

An alternative optimization procedure to the proposed Newton scheme would be an explicit projected gradient descent; i.e. the subspace minimization step is skipped. A benefit of this is that one does not need to compute the Newton direction, which simplifies the implementation and decreases the computational cost of an iteration. On the other hand, it is well known that for this type of problem an explicit gradient descent usually requires a large number of steps. The comparison shown in Table 1 demonstrates that the explicit gradient descent is only efficient if the mesh has nicely shaped triangles and the deviation tolerance is very small; whereas the Newton scheme performs well on all the examples.

An implicit gradient descent can handle irregular meshes and larger steps, but the computation of an implicit step itself requires solving a nonlinear equation, cp. [WBH<sup>\*</sup>07]. Even if the equation is linearized (semi-implicit scheme) the cost of computing the gradient direction is comparable to the cost of computing the Newton direction.

**Future work.** Future work addresses extending the approach to point cloud surfaces. This would remove the need

Model	#T	Max. Tol. in mm	Newton		Exp. Gradient	
			#iter	cost (s)	#iter	cost (s)
Caesar	190 k	0.1	20	103	95	172
Caesar	190 k	0.3	20	101	>500	>850
Blade	390 k	1	15	216	90	364
Blade	390 k	3	15	209	>500	>2000
Blade*	100 k	1	15	48	250	278

**Table 1:** A comparison of running times of the proposed Newton scheme and an explicit gradient descent is shown. Different values of the maximum deviation tolerance were tested on the Caesar and the blade model. The procedures were stopped at equal values of the energy. The last row shows running times of the methods on an irregular mesh generated by simplifying the blade model.

to generate a mesh before being able to apply the method to scan data. We are interested in developing strategies to estimate the accuracy of the scan, which would help to automate the smoothing process. Moreover we experiment with weighted maximum deviation measures.

**Acknowledgements.** This work was supported by the DFG Research Center MATHEON "Mathematics for Key Technologies" in Berlin. We would like to thank Max Wardetzky for valuable discussions, Hans Lamecker for generating the pelvis model, and Miriam Kabaum for her support. The models are courtesy of Aim@Shape Project, IMATI, INRIA, Zuse Institute Berlin, and Stanford University.

## References

- [Ale02] ALEXA M.: Wiener filtering of meshes. In *Proceedings of the Shape Modeling International* (2002), IEEE Computer Society, pp. 51–57.
- [BMF03] BRIDSON R., MARINO S., FEDKIW R.: Simulation of clothing with folds and wrinkles. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003), pp. 28–36.
- [B003] BELYAEV A., OHTAKE Y.: A comparison of mesh smoothing methods. In *Proceedings of the Israel-Korea Bi-National Conference on Geometric Modeling and Computer Graphics* (2003), pp. 83–87.
- [Bob05] BOBENKO A. I.: A conformal energy for simplicial surfaces. *Combinatorial and Computational Geometry* (2005), 133–143.
- [BS05] BOBENKO A. I., SCHRÖDER P.: Discrete Willmore flow. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Geometry Processing* (2005), pp. 101–110.
- [BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *Proceedings of ACM SIGGRAPH* (1998), pp. 43–54.
- [BX03] BAJAJ C., XU G.: Anisotropic diffusion on surfaces and functions on surfaces. *ACM Transactions on Graphics* 22 (2003), 4–32.
- [CDD\*04] CLARENZ U., DIEWALD U., DZIUK G., RUMPF M., RUSU R.: A finite element method for surface restoration with smooth boundary conditions. *Computer Aided Geometric Design* 21, 5 (2004), 427–445.
- [CDR00] CLARENZ U., DIEWALD U., RUMPF M.: Anisotropic geometric diffusion in surface processing. *Proceedings of IEEE Visualization* (2000), 397–405.
- [CSM03] COHEN-STEINER D., MORVAN J.-M.: Restricted Delaunay triangulations and normal cycles. *Proceedings of the ACM Symposium on Computational Geometry* (2003), 237–246.
- [Del01] DELINGETTE H.: On smoothness measures of active contours and surfaces. In *Proceedings of the IEEE Workshop on Variational and Level Set Methods* (Washington, DC, USA, 2001), IEEE Computer Society.
- [DMSB99] DESBRUN M., MEYER M., SCHRÖDER P., BARR A. H.: Implicit fairing of irregular meshes using diffusion and curvature flow. *Proceedings of ACM SIGGRAPH* (1999), 317–324.
- [FDC003] FLEISHMAN S., DRORI I., COHEN-OR D.: Bilateral mesh denoising. *Proceedings of ACM SIGGRAPH* (2003), 950–953.
- [GHDS03] GRINSPUN E., HIRANI A., DESBRUN M., SCHRÖDER P.: Discrete shells. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2003), pp. 62–67.
- [Gib98] GIBSON S. F. F.: Constrained elastic surface nets: Generating smooth surfaces from binary segmented data. In *Medical Image Computing and Computer-Assisted Intervention - MICCAI* (1998), Wells W. M., Colchester A. C. F., Delp S. L., (Eds.), Springer, pp. 888–898.
- [GP01] GROSS M., PAULY M.: Spectral processing of point-sampled geometry. *Proceedings of ACM SIGGRAPH* (2001), 379–386.
- [HP04] HILDEBRANDT K., POLTHIER K.: Anisotropic filtering of non-linear surface features. *Computer Graphics Forum*, 23(3) (2004), 391–400.
- [JDD03] JONES T. R., DURAND F., DESBRUN M.: Non-iterative, feature-preserving mesh smoothing. *Proceedings of ACM SIGGRAPH* (2003).
- [Kan01] KANZOW C.: An active set-type newton method for constrained nonlinear systems. In *Complementarity: Applications, Algorithms and Extensions* (2001), Ferris M., Mangasarian O., Pang J.-S., (Eds.), Kluwer Academic Publishers, pp. 179–200.
- [LBH\*01] LIU X., BAO H., HENG P., WONG T., PENG Q.: Constrained fairing for meshes. *Computer Graphics Forum* 20 (2001), 115–122.
- [LM99] LIN C.-J., MORÉ J. J.: Newton’s method for large bound-constrained optimization problems. *SIAM Journal on Optimization* 9, 4 (1999), 1100–1127.
- [MS92] MORETON H. P., SÉQUIN C. H.: Functional optimization for fair surface design. In *Proceedings of ACM SIGGRAPH* (1992), pp. 167–176.
- [NGH\*03] NIELSON G. M., GRAF G., HOLMES R., HUANG A., PHIELIPP M.: Shrouds: Optimal separating surfaces for enumerated volumes. In *Symposium on Visualization* (2003), Eurographics Association.
- [NISA06] NEALEN A., IGARASHI T., SORKINE O., ALEXA M.: Laplacian mesh optimization. In *Proceedings of ACM GRAPHITE* (2006), pp. 381–389.
- [NW06] NOCEDAL J., WRIGHT S. J.: *Numerical Optimization (2nd edition)*. Springer Series in Operations Research. Springer-Verlag, New York, NY, 2006.
- [PM87] PERONA P., MALIK J.: Scale space and edge detection using anisotropic diffusion. *IEEE Computer Society Workshop on Computer Vision* (1987).
- [PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Experim. Math.* 2, 1 (1993), 15–36.
- [PSZ01] PENG J., STRELA V., ZORIN D.: A simple algorithm for surface denoising. *Proceedings of IEEE Visualization* (2001).
- [SCO04] SORKINE O., COHEN-OR D.: Least-squares meshes. In *Proceedings of Shape Modeling International* (2004), IEEE Computer Society Press, pp. 191–199.
- [SK00] SCHNEIDER R., KOBBELT L.: Generating fair meshes with  $G^1$  boundary conditions. *Proceedings of ACM SIGGRAPH* (2000), 247–256.
- [Tau95] TAUBIN G.: A signal processing approach to fair surface design. *Proceedings of ACM SIGGRAPH* (1995).
- [TM98] TOMASI C., MANDUCHI R.: Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision* (1998), pp. 839–846.
- [TWBO02] TASDIZEN T., WHITAKER R., BURCHARD P., OSHER S.: Geometric surface smoothing via anisotropic diffusion of normals. *Proceedings of IEEE Visualization* (2002), 125–132.
- [VVR06] VOLODINE T., VANDERSTRAETEN D., ROOSE D.: Smoothing of meshes and point clouds using weighted geometry-aware bases. In *Proceedings of Geometric Modeling and Processing* (2006), pp. 687–693.
- [WBH\*07] WARDTZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *to appear in Computer Aided Geometric Design* (2007).
- [WN01] WESTGAARD G., NOWACKI H.: A process for surface fairing in irregular meshes. *Computer Aided Geometric Design* 18, 7 (2001), 619–638.
- [YB02] YOSHIZAWA S., BELYAEV A. G.: Fair triangle mesh generation with discrete elastica. In *Proceedings of Geometric Modeling and Processing* (2002), IEEE Computer Society, pp. 119–123.
- [YBS06] YOSHIZAWA S., BELYAEV A., SEIDEL H.-P.: Smoothing by example: Mesh denoising by averaging with similarity-based weights. In *Proceedings of the IEEE International Conference on Shape Modeling and Applications* (2006), IEEE.