Anisotropic filtering on normal field and curvature tensor field using optimal estimation theory

Min Liu Yushen Liu and Karthik Ramani
Purdue University, West Lafayette, Indiana, USA
Email: {liu66| liu28| ramani}@purdue.edu

Abstract

In this paper, we study the problem of mesh denoising for improving the single pass surface estimation on normals and curvature tensors. We focus mainly on the engineering objects represented as dense triangle meshes. In particular, a two run non-linear diffusion algorithm based on optimal estimation theory is proposed to adaptively filter out the undesired discontinuities introduced by noise while preserving the underlying features. We show that the proposed filter can successfully improve the local surface estimates while preserving the desired features in terms of tangential and curvature discontinuities.

1. Introduction

Smoothness refers to the mathematical notion of continuous differentiability, or continuity. When a smooth surface is approximated as a triangle mesh, it maintains only the positional continuity (no gaps) of the underlying surface, while the tangential continuity (no sharp angles) and the curvature continuity (no sharp radius changes) are lost due to the discrete nature of the mesh. This brings about the problem of estimating the discrete surface differential properties for the meshed objects. Discrete estimation of surface differential parameters is essential for many mesh based applications, but the problem is difficult if the underlying geometry contains natural surface discontinuities. The problem gets even harder when a mesh is constructed from scanned data since they carry measurement and quantization errors. The differential properties such as principal curvatures and directions are very sensitive [13] to those errors. A single pass estimation, based on either local surface fitting using the polynomial [15, 4, 7, 20] or the estimation of a curvature tensor in the local neighborhood [17, 11, 1, 14] are error prone, due to the presence of the natural discontinuities as well as the noise in the data. In practice, this means that the estimates computed on a local basis must be improved at a later stage.

The techniques related to isotropic denoising assume that there are no underlying surface discontinuities, therefore the errors in the estimates result from only noise which is randomly and uniformly distributed. Representative isotropic denoising methods includes *Laplacian smoothing* [18], *mean curvature* flow [3], and *curvature consistency framework* [15]. The first two methods [18, 3] achieve smoother surfaces by changing the vertex positions to relax all the curvature peaks, with improvements in the surface normal and curvature estimations as a byproduct; while the last method [15] improves the curvature estimation by minimizing a functional form related to a minimum variation of curvature estimates.

In real case, a typical mesh contains random noise as well as surface discontinuities. Anisotropic denoising methods, first introduced in image processing and later extended to geometric problems, were developed to preserve or enhance features like sharp edges or corners. Some methods smoothen height fields by controlling the weights in mean curvature flow [2, 5, 8]. Other methods [19, 16, 12] use diffusion filter to smoothen the normal field and then integrate this to height field to get the smoother surface.

In this work, we study the problem of mesh denoising for improving the single pass surface estimates. We target mainly the scanned engineering objects which are represented as dense triangle meshes, though it can be applied for general shapes too. Our proposed method, derived from a signal filter, originates from the optimal estimation theory [6]. In stead of modifying the vertex positions, we improve the normals and curvatures estimates by arguing that those differential properties are more important than removing noise in the geometry because, the typical two stages in reverse engineering: segmentation and surface fitting relies more on the good estimates of surface differential properties.

The focus of our work targets three problems:

- An estimation of discrete shape operator.

  We propose a simple per-face discrete curvature estimation method in term of curvature tensor. It benefits

IEEE International Conference on Shape Modeling and Applications(SMI'07)
0-7695-2815-5/07 $20.00 © 2007 IEEE

the edge-based mesh processing since the sharp variation of differential properties between the two neighboring triangles potentially defines a feature edge.

- An anisotropic filter on the normal and curvature fields.

  We adapt the optimal estimation theory into a two run non-linear diffusion process; one for normal consistency in which the filter is applied on a vector field (normals). And the other is for curvature consistency in which the filter is applied on a matrix field (curvature tensors). Each stage tries to preserve the underlying discontinuities (features) while smoothing the single pass normal and curvature estimates.

- A general framework for extending the diffusion process into the differential parameter fields with higher orders.

  We apply the non-linear diffusion filter both on the normal field and on the curvature field to handle engineering objects. We also show that it can be generalized to handle higher order differential properties.

2. The local geometry estimation for per-face

For each triangle f in a mesh, its normal direction n_f is well defined. The normal direction at each vertex p in the mesh is estimated by the weighted average of the normals of faces adjacent to p (1-ring neighborhood faces). We use the Nelson Max's weighting method [10] for weights (area of f divided by the squares of the lengths of the two edges that touch vertex p) as this produces more accurate normal estimates than other weighting approaches.

We estimate the per-face curvature by observing the normal variations along its three edges. For each triangle f, there is an associated shape operator in matrix format called curvature tensor Λ:

$$\Lambda_f = \begin{pmatrix} D_u n & D_v n \end{pmatrix} = \begin{pmatrix} \frac{\partial n_f}{\partial u}\cdot u & \frac{\partial n_f}{\partial v}\cdot u \\ \frac{\partial n_f}{\partial u}\cdot v & \frac{\partial n_f}{\partial v}\cdot v \end{pmatrix},$$

where (u,v) is a pair of orthogonal unit vectors describing an orthonormal coordinate system in the tangent frame of f. Multiplying a curvature tensor by any vector in the tangent plane gives the directional derivative of the normal in that direction: $\Lambda \cdot t = D_t n$.

The curvature tensor for each facet is constructed using method introduced in [14]. For a facet f, there are three well-defined directions (the edges) together with the differences in normals along those directions (computed from the per-vertex normal), refer Figure 1. In this figure, $n_i (i = 0,1,2)$ are the estimated normals on three vertices of the triangle f. The equations in Figure 1 provide a set of linear constraints on the elements of the second fundamental tensor, which may then be determined using least square fitting. Notice that the principal curvatures (k_1, k_2), and the principle directions (e_1, e_2) of the triangle f correspond to the eigenvalues and eigenvectors of Λ_f, while the determinant of the tensor $\mathrm{Det}(\Lambda_f)$ and the half trace of the tensor $\mathrm{Tr}(\Lambda_f)/2$ can be considered as a local estimation of Gaussian curvature K_f and Mean curvature H_f.

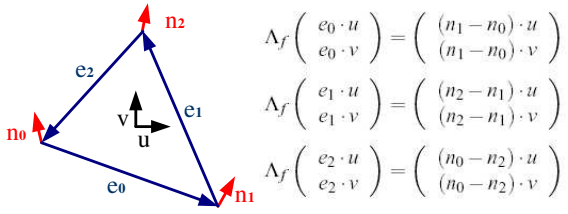


Figure 1. Estimate per-face curvature tensors

3 Anisotropic filtering based on the optimal estimation theory

The local differential properties estimated so far (refer Section 2) are error-prone due to the existence of surface discontinuities which violates the smoothness assumption for the underlying geometries. The general objective of an anisotropic filtering is to smooth out the surface discontinuities introduced by the random noise and preserve the true surface continuities.

The optimal estimation theory, originally introduced in [6] for signal processing and adapted by [9] in a curvature consistency framework for the edge detection on a range image, offers a good method for integrating measurements from multiple sources with different noise to predict a constant quantity. The main idea is that, *given a noisy source, its contribution to predict the behavior of a target in the next iteration is dependent on the variance of its prediction error in the past*. A source with a lower variance of estimation error is considered to have better fitness and will be given higher weight in the next prediction. This mechanism works elegantly for our purpose of anisotropic filtering on the initially estimated surface parameters. We use the properties of neighboring facets to redefine the properties of a target facet. At the beginning of optimal estimation flow, the full variational relaxation takes place. This relaxation smoothes out random noise and starts to distort the true surface discontinuities meanwhile. However, as the iteration progresses, the real nature of the local surface is learned by recording the estimation error variances in each step. In subsequent iterations, neighbors believed to be in the same

continuous region (i.e. the neighbor with low estimation error variance) will dominate the relaxation process. The relaxation across discontinuous regions will be penalized since they introduce large variances of estimation error.

Now we reformulate the above idea to our per-face geometry estimator. For each triangle f, consider its 1-ring neighborhood $N(f)$ (which are triangles sharing common edges or vertices with f) as a noisy prediction source. During the optimal estimation flow, differential properties of f, at a new iteration $l+1$ ($\triangle f^{l+1}$) is estimated by the weighted average of predictions from its neighbors. Each of its neighboring facet s, casts a vote $\tilde{\triangle}s$ on f. $\triangle f^{l+1}$ is given by the following equation;

$$\triangle f^{l+1} = \frac{\sum_{s\in N(f)} w_s^l \tilde{\triangle}s}{\sum_{s\in N(f)} w_s^l}, \tag{1}$$

where w_s^l is the weight assigned to a neighbor s for measuring how much the prediction of s will contribute to the behavior of the target. $\tilde{\triangle}s$ can be equal to $\triangle s$ but it is not necessary. If equal weights are assigned to each of f's neighbors, the above iterative process (Equation (1)) provides an isotropic filtering which is not desired for preserving underlying features in our case. For assigning weights to adjust the contribution of prediction from each neighbor, we use the following functions that are based on the optimal estimation theory [9];

$$w_s^l = \exp(-\frac{\sigma_s^{2^l}}{\gamma^l}), \tag{2}$$

$$\sigma_s^{2^l} = \frac{1}{k}\sum_{l=0}^{l} \varepsilon_s^{2^l}, \tag{3}$$

$$\gamma^l = \frac{2}{|N(f)|}\sum_{s\in N(f)} \sigma_s^{2^l}, \tag{4}$$

where $\sigma_s^{2^l}$ is the variance of prediction error in lth iteration and γ^l is the smoothing control parameter which is set by taking the mean of the error variance value among all the triangles in $N(f)$. ε_s measures the prediction error, i.e. difference between $\tilde{\triangle}s$ and $\triangle f$.

In equation (2), the weight is defined as a function of prediction error variance ($\sigma_s^{2^l}$), which is normalized by the mean variance (γ^l) among all neighbors who participate in current prediction. The exponential function is chosen to avoid a single neighbor from taking over the relaxation process by bonding the weights as the error variances approach zero. As desired by the optimal estimation theory, this weighting function gives higher weights to the neighbors with low error variances. In addition, it provides similar weights to the neighbors which are correlated with the target facet in the same subregion.

A key problem is, how to estimate the error so that the weights can be correctly adjusted. In the curvature consistency framework [15, 9], this iterative process is conducted in a single pass on the Augmented Darboux frame [15] (contains the normal and the curvature elements) of a vertex p, the diffusion is applied on both the normal and the curvature parameters and the error metric ε simply combines the estimation error of all elements:

$$\varepsilon_s^{2^l} = |n_f^l - \tilde{n_s}^l|^2 + |e1_f^l - \tilde{e1_s}^l|^2 + |k1_f^l - \tilde{k1_s}^l|^2 + |k2_f^l - \tilde{k2_s}^l|^2. \tag{5}$$

This error metric causes a major problem since it simultaneously fuses on two scalar properties (k_1 and k_2), and two vector properties (n, e_1). Cases exist where the scalar element dominates the error function so that the vector elements never contribute in this optimal estimation framework.

We apply the diffusion process to different runs corresponding to different order of differential parameter estimations so that in each run, the error variance can be measured in the same field.

We first apply the optimal estimation process on surface normals. Equation (1) in this run becomes;

$$n_f^{l+1} = \frac{\sum_{s\in N(f)} w_s^l n_s^l}{\sum_{s\in N(f)} w_s^l}. \tag{6}$$

and the error metric is given by the angular distance between estimated normal of f and the normal of neighboring facet s:

$$\varepsilon_s{}^l = 1 - \frac{n_f{}^l \cdot n_s^l}{|n_f{}^l \cdot n_s^l|}. \tag{7}$$

In the second run, the curvature tensor Λ estimated on f, is considered for the optimal estimation. Since the tensor Λ_f is defined on a local parameter space associated with three orthogonal directions (n_f, u_f, v_f), the tensor Λ_s of its neighboring facet s, which is defined on (n_s, u_s, v_s), cannot directly cast the vote on f. Therefore, a neighboring tensor has to be aligned in the space of the target facet (refer to Figure 2). For each Λ_s, this alignment forms a rotated tensor Λ_s', which will be used for prediction.

$$\Lambda_f^{l+1} = \frac{\sum_{s\in N(f)} w_s^l \Lambda_s'^l}{\sum_{s\in N(f)} w_s^l}. \tag{8}$$

Two error metrics can be used for calculating variances and weights. One is the variation between principal curvatures of f and those of its neighbor s and the other is the determinant of difference matrix between the estimated tensor Λ_f and Λ_s'.

Variation between principal curvatures contains two elements;

$$\varepsilon_1{}^l = \sqrt{(k_{1f}^l - k_{1s}^l)^2 + (k_{2f}^l - k_{2s}^l)^2} \quad \text{and} \quad \varepsilon_2{}^l = sin\theta_e^l. \tag{9}$$

The first element in Equation (9) is the Euclidean distance between (k_{1f}, k_{2f}) and (k_{1s}, k_{2s}), and the other is the

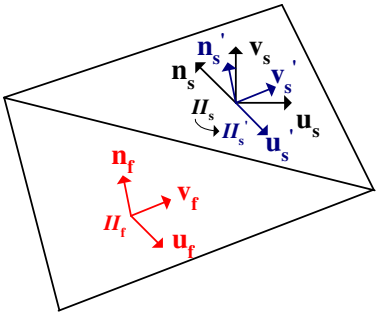


Figure 2. Rotate local space of tensor Λ_s to the space of neighbor triangle's tensor Λ_f,. This forms a rotated tensor Λ_s' .

angular distance between one of the principal directions e_{1f} and e_{1s}. θ_e^l is the angle between e_{1f} and e_{1s} in the lth iteration. We use $sin\theta_e^l$ to treat the acute angle the same as the obtuse angle because the principal direction is actually a line field rather than a vector field. The above two error metrics are not on the same scales and therefore should not be summed up directly into one error function. The variance of each element will be calculated separately to get two corresponding $\sigma_{s\,1}^{2l}$, γ_1^l and $\sigma_{s\,2}^{2l}$, γ_2^l which corresponds to the variance of principal curvature variations and the variance of principle direction variations. The final weight function (Equation (2)) is:

$$w_s^l = \exp\left(-\frac{\sigma_{s\,1}^{2l}}{\gamma_1^l} - \frac{\sigma_{s\,2}^{2l}}{\gamma_2^l}\right). \tag{10}$$

Another option for the error metric for curvature tensor diffusion is a simpler uniformed metric which is defined by the determinant of the tensor difference,

$$\varepsilon_s{}^l = \|\Lambda_f{}^l - \Lambda_s''\| \tag{11}$$

Generally speaking, the first error metric (Equation (10)) captures the distances of principal curvatures and principal directions among neighboring triangles. The second one (Equation (11)) can be considered as a measurement of total variation of curvatures along all the directions in the parameter space of triangle f. Both the error metrics have their own disadvantages. The first metric will have problems in the vicinity of umbilical points, since the estimation of the principal curvature directions gets unstable on umbilical point. The second error metric vanishes for singular differences matrices. Consequently, different curvature tensors may have a zero distance. We leave the improvements of the error metrics for curvature tensor as a future work.

4 Results

We have implemented the anisotropic mesh denoising algorithm as described in the previous section, and compared our results to that of two popular anisotropic denoising methods, *bilateral mesh denoising* algorithm [5] and *adaptive smoothing* algorithm [12]. We also compared our results of curvature denoising to the methods based on curvature consistency framework; isotropic denoising method proposed in [15], and the anisotropic scheme proposed in [9]. These two methods denoise the mesh on both the normal and the curvature field in a single pass. We use the following color map (see Figure 3) to show the two estimated principal curvatures through all the results.

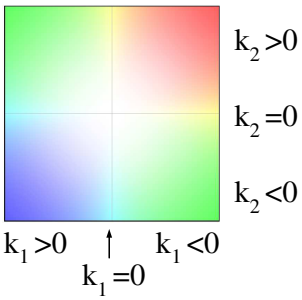


Figure 3. The color map of the curvature value.

Figure 4 shows the result of applying our anisotropic filter for denoising the Fandisk model (with 23964 triangles, 11984 vertices, 35946 edges). This model contains noise near the feature regions as well as global gaussian noise (Figure 4(a)). After applying five iterations of our denoising filter for normals, the estimated normals are smoothed while the underlying sharp variations on the normals are kept (see Figure 4(b) for the rendering effect of this smoothing). Figures 4(c) and 4(d) show the normal distributions (normals are rendered as red lines) of a small region (rectangle region in Figure 4(a)) before and after applying our normal filter. The sharp edges (colored as light blue) behave more regularly and directionally as the result of anisotropic denoising on normal field. Figures 4(e) and 4(f) show the curvature distributions of the model before and after applying our filter on the curvature tensor field, the small line strokes denote the minimum curvature directions. Figure 6 gives the results of two other anisotropic denoising algorithms, Figures 6(a) and 6(b) are the results of applying five iterations of bilateral mesh denoising proposed in [5], Figure 6(c) and 6(d) are the results of applying same number of iterations using the adaptive mesh denoising proposed in [12]. For this model, the bilateral denoising does not produce very good denoising results. Denoising effects on the curvature

estimates, as the byproducts of denoising the height field are shown in Figures 6(b) and 6(d). Notice the differences in the curvature distributions compared to the result of our method in Figure 4(f).

Figure 5 shows how our algorithm works on a noisy rocker arm model (Figure 5(a)). After applying our filter on the normal field, the smoothing result on the surface normal is shown in Figure 5(b). Figure 5(c) is the initial estimation of curvatures, denoising on the normal field will improve the curvature estimation too, see the result in Figure 5(d). Figures 5(e) and 5(f) show the results of applying five iterations of our filter on the curvature tensor field, using two error metrics proposed in Section 3, (Equation (11) and (9) respectively). They both work well and result in similar denoising effect (refer to Figure 5(e) and 5(f)). Since Equation (11) gives a simpler computation for error metric, we recommend the readers to use it.

We compared our result using anisotropic denoising on curvature tensor field to the following methods that improve the curvature estimates; one is the anisotropic scheme proposed in [9] (Figure 7(a)) and the other is the isotropic scheme proposed in [15] (Figure 7(b)). Our method achieves better results of smoothing curvature estimations while preserving the underlying discontinuities (refer to Figure 5(f)). Observe the difference in the curvature distribution along the flat region on the rocker arm model.

We also tested our method on some general shapes (see Figure 8 for example). The tests show that our anisotropic filter works well for the general shapes too.

5 A general anisotropic filtering framework

In theory, the framework of our anisotropic diffusion algorithm can be generalized for treating higher order features. The first step is to extend the per-face local geometry estimation to higher order differential properties. We show here how to extend it to third order differential properties. The third order differential properties can be defined as a "derivative of curvature" tensor that gives the changes in curvature along the surface. It is a $2\times 2\times 2$ rank-3 tensor with four unique entries:

$$\mathscr{C} = (D_u\Lambda, D_v\Lambda) = \left(\begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} b & c \\ c & d \end{pmatrix} \right).$$

To compute the above tensor, we need to estimate the curvature tensor Λ_p at each vertex p on the mesh. This can be done by the weighted average of the curvature tensors of its triangle neighbors:

$$\Lambda_p = \frac{\sum_{i=1}^{m} w_i \Lambda_i'}{\| \sum_{i=1}^{m} w_i n_i \|},$$

where the Λ_i' are the rotated tensor (see Figure 2) of the triangles in the 1-ring neighborhood of p, The tensor is rotated to align with the parameter space of vertex p. Weights w_i denote how much of the curvature tensor should be accumulated at each vertex p. The "Voronoi area" weighting scheme that takes the weight be proportion of the area of f that lies close to vertex p [11] is a good choice.

The "derivative of curvature tensor" of a triangle can then be estimated with a least-squares fit to the differences in the curvature tensors along three edges of each triangle, in the same manner as we did for curvature tensor estimation. For the anisotropic diffusion filter for the third order tensor, the iteration equation can be defined as,

$$\mathscr{C}_f^{l+1} = \frac{\sum_{s\in N(f)} w_s^l \mathscr{C}_s'^l}{\sum_{s\in N(f)} w_s^l}, \tag{12}$$

where $\mathscr{C}_s'^l$ denotes the rotated "derivative of curvature tensor", and the error metrics can then be defined as:

$$\varepsilon_s{}^l = \| \mathscr{C}_f{}^l - \mathscr{C}_s'^l \|. \tag{13}$$

6 Conclusions and future directions

We have presented a feature-preserving filter for the normals and the curvatures estimated on the dense mesh models. For some reverse engineering applications, the proposed filter is generally a sufficient preprocessing step for further geometric operations such as segmentation and surface fitting. We also showed that this anisotropic filter can be extended to improve higher order differential parameters. In future, testing the method for denoising on higher order fields will be carried out.

Our anisotropic filter of normals and curvatures can be used for stable extraction of feature edges corresponding to C^1 discontinuities and C^2 discontinuities that can help in segmenting dense meshes.

The current implementation of the iterative diffusion process simply specifies the number of iterations as a user defined parameter. In future, the iteration will be controlled by tracking the convergence of the differential properties.

References

[1] D. Cohen-Steiner and J.-M. Morvan. Restricted delaunay triangulations and normal cycle. In *SCG '03: Proceedings of the nineteenth annual symposium on Computational geometry*, pages 312–321, New York, NY, USA, 2003. ACM Press.

[2] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Anisotropic Feature-Preserving denoising of height fields and images. In *Graphics Interface*, pages 145–152.

[3] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Implicit fairing of irregular meshes using diffusion and curvature flow. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 317–324, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co.

[4] F. P. Ferrie, J. Lagarde, and P. Whaite. Darboux frames, snakes, and super-quadrics: Geometry from the bottom up. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(8):771–784, 1993.

[5] S. Fleishman, I. Drori, and D. Cohen-Or. Bilateral mesh denoising. *ACM Trans. Graph.*, 22(3):950–953, 2003.

[6] A. Gelb. *Applied Optimal Estimation*. MIT Press, Cambridge, Mass., 1974.

[7] B. Hamann. Curvature approximation for triangulated surfaces. pages 139–153, 1993.

[8] K. Hildebrant and K. Polthier. Anisotropic filtering of non-linear surface features. *Computer Graphics Forum*, 23(3):391–400, 2004.

[9] S. Mathur and F. P. Ferrie. Edge localization in surface reconstruction using optimal estimation theory. In *CVPR '97: Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, page 833, Washington, DC, USA, 1997. IEEE Computer Society.

[10] N. Max. Weights for computing vertex normals from facet normals. *J. Graph. Tools*, 4(2):1–6, 1999.

[11] M. Meyer, M. Desbrun, P. Schr, and A. Barr. Discrete differentialgeometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III, edited by H.C. Hege and K. Polthier*, pages 35–37, Heidelberg, 2002. Springer.

[12] Y. Ohtake, A. G. Belyaev, and I. A. Bogaevski. Mesh regularization and adaptive smoothing. *Computer-Aided Design*, 33(4):789–800, 2001.

[13] S. Petitjean. A survey of methods for recovering quadrics in triangle meshes. *ACM Comput. Surv.*, 34(2):211–262, 2002.

[14] S. Rusinkiewicz. Estimating curvatures and their derivatives on triangle meshes. In *3DPVT '04: Proceedings of the 3D Data Processing, Visualization, and Transmission, 2nd International Symposium on (3DPVT'04)*, pages 486–493, Washington, DC, USA, 2004. IEEE Computer Society.

[15] P. T. Sander and S. W. Zucker. Inferring surface trace and differential structure from 3-d images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(9):833–854, 1990.

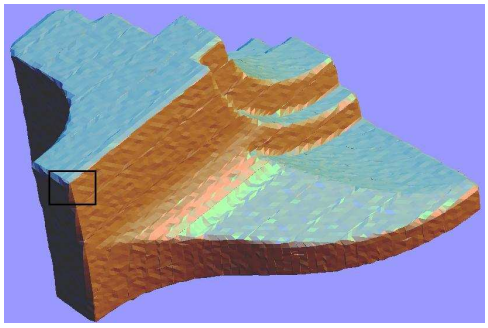
[16] T. Tasdizen, R. Whitaker, P. Burchard, and S. Osher. Geometric surface smoothing via anisotropic diffusion of normals. In *VIS '02: Proceedings of the conference on Visualization '02*, pages 125–132, Washington, DC, USA, 2002. IEEE Computer Society.

[17] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *ICCV '95: Proceedings of the Fifth International Conference on Computer Vision*, page 902, Washington, DC, USA, 1995. IEEE Computer Society.

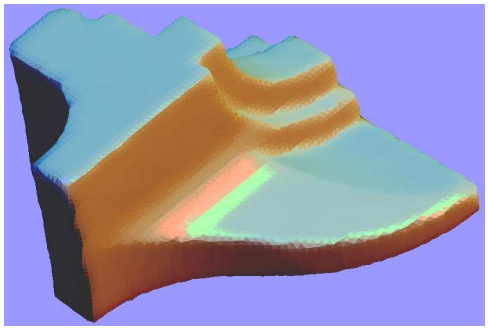
[18] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 351–358, New York, NY, USA, 1995. ACM Press.

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

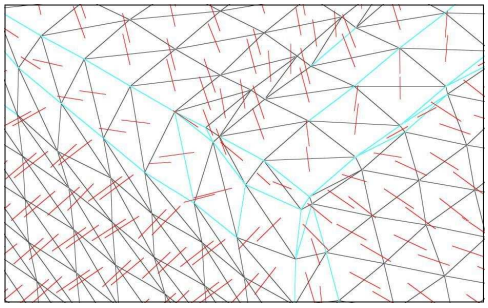
[20] S. Yoshizawa, A. Belyaev, and H.-P. Seidel. Fast and robust detection of crest lines on meshes. In *SPM '05: Proceedings of the 2005 ACM symposium on Solid and physical modeling*, pages 227–232, New York, NY, USA, 2005. ACM Press.



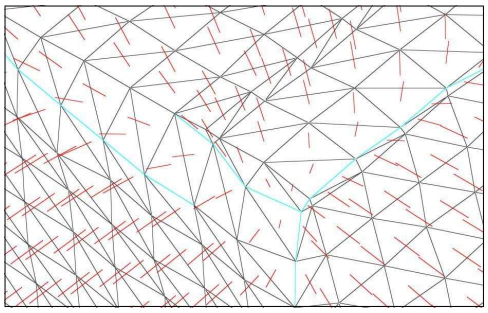
(a) Input noisy model (23964 triangles, 11984 vertices, 35946 edges)



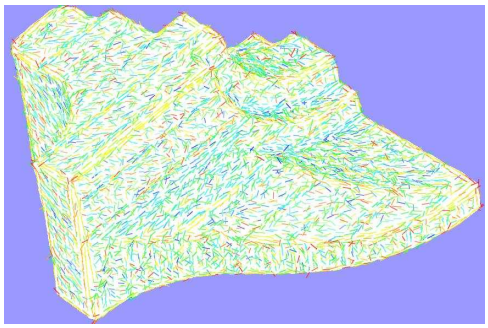
(b) Five iterations of our filter applied on the normal field



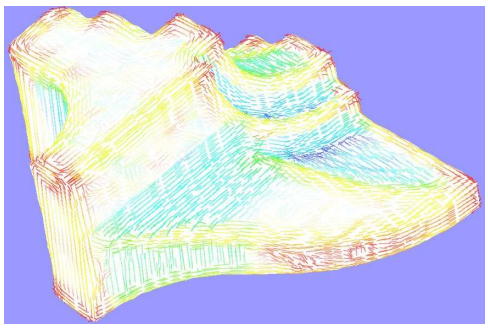
(c) Initially estimated of surface normals



(d) Normal estimates after anisotropic filtering on normal field

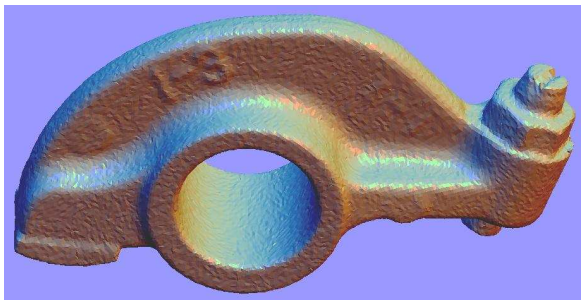


(e) Initial estimation of curvature values and directions, the small line strokes show the minimum curvature directions

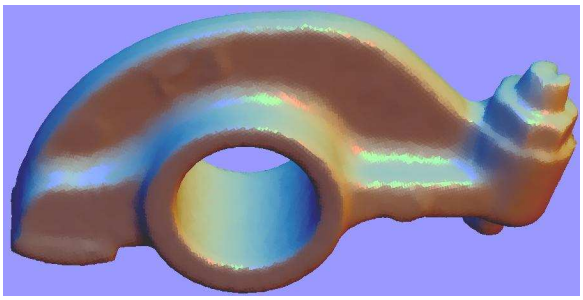


(f) Curvature estimates after applying 5 iterations of our anisotropic filter on curvature tensor field

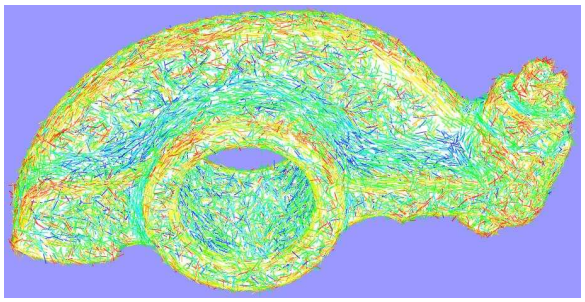
Figure 4. Our anisotropic filter applied on the normal field and the curvature tensor field for the fan disk model.



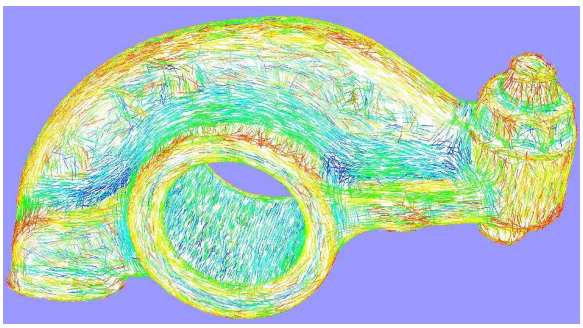
(a) The original model(80354 triangle, 40177 vertices and 120531 edges)



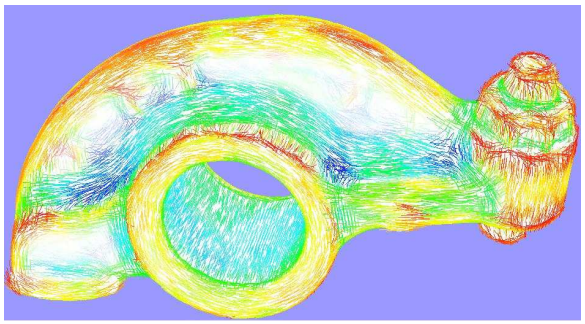
(b) Five iterations of our filter applied on the normal field



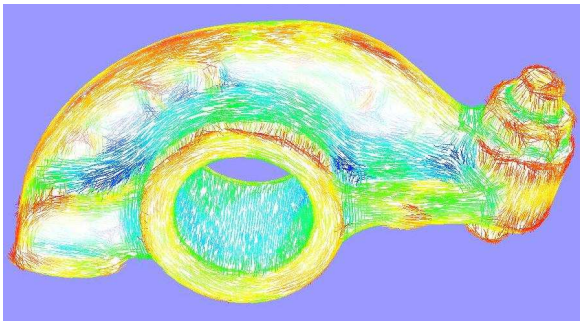
(c) Initial estimated curvature values and minimum curvature directions



(d) The curvature distribution after applying five iterations of normal diffusion

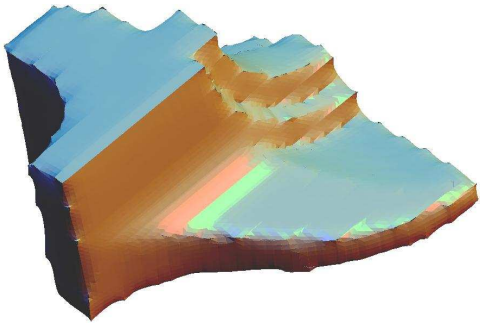


(e) The curvature distribution after applying five iterations of curvature tensor diffusion using error metrics in equation 11

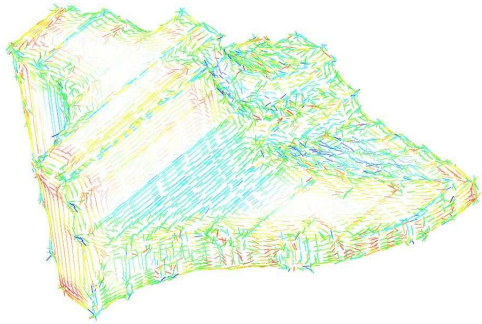


(f) The curvature distribution after applying five iterations of curvature tensor diffusion using error metrics in equation 9

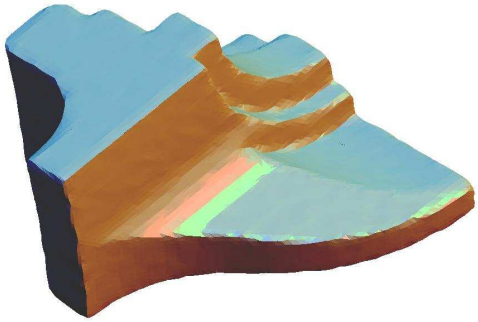
Figure 5. Five iterations of our filter applied on the normal field and then the curvature tensor field of a noisy rocker arm model (model courtesy of Leif Kobbelt)



(a) Denoising results using the bilateral filter [5] (5 iterations, $\sigma_f = 4$, $\sigma_g = 4$)



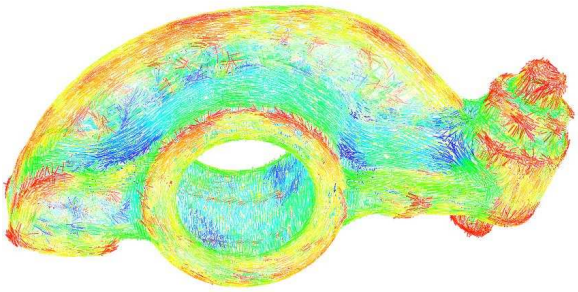
(b) Curvature distribution after applying the bilateral filter



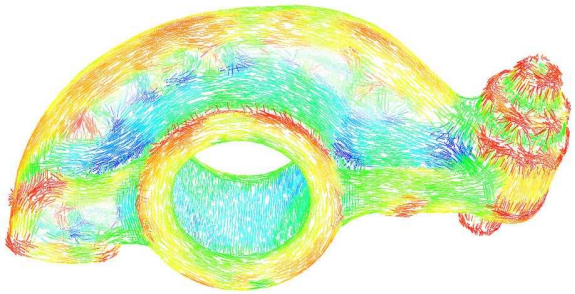
(c) Denoising results using the adaptive smoothing [12] (5 iterations or normal diffusion + 5 iterations on the hight field, $c = 2$)

(d) Curvature distribution after applying the adaptive smoothing filter

Figure 6. Comparisons to the result two other methods for the hight field denoising applied on the Fandisk model.

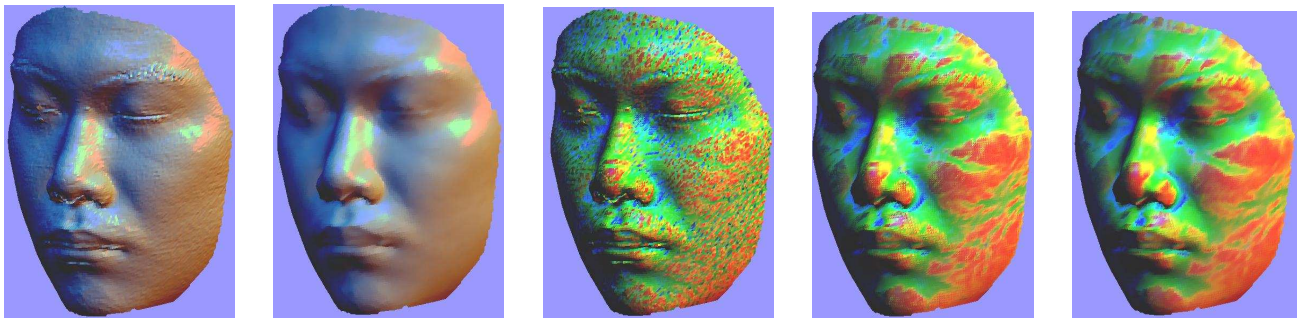


(a) Anisotropic denoising on the Augmented Darboux frame using the error metrics in Equation 5 [9]



(b) Isotropic denoising on the Augmented Darboux frame under the curvature consistency framework [15]

Figure 7. Comparisons to the results of two other methods for the curvature field denoising applied on the rocker arm model.



(a) The input noisy model

(b) The result of two iterations of our filter applied on normal field

(c) The curvature value estimated using initial per-face curvature estimation

(d) The byproduct of curvature value distribution after applying 2 iterations of our filter on the facet normals

(e) The curvature value distribution after applying 2 iterations of our filter on curvature tensor field

Figure 8. Illustration of our anisotropic filter applied on a general shape (data courtesy of Alexander Belyaev).