Published in final edited form as:
Graph Models. 2011 July 1; 73(4): 127–139. doi:10.1016/j.gmod.2011.01.002.

Quality Mesh Smoothing via Local Surface Fitting and Optimum Projection

Jun Wang and **Zeyun Yu***
Department of Computer Science, University of Wisconsin-Milwaukee, 3200 N. Cramer St., Milwaukee, WI 53211, USA

Abstract

The smoothness and angle quality of a surface mesh are two important indicators of the "goodness" of the mesh for downstream applications such as visualization and numerical simulation. We present in this paper a novel surface mesh processing method not only to reduce mesh noise but to improve angle quality as well. Our approach is based on the local surface fitting around each vertex using the least square minimization technique. The new position of the vertex is obtained by finding the maximum inscribed circle (MIC) of the surrounding polygon and projecting the circle's center onto the analytically fitted surface. The procedure above repeats until the maximal vertex displacement is less than a pre-defined threshold. The mesh smoothness is improved by a combined idea of surface fitting and projection, while the angle quality is achieved by utilizing the MIC-based projection scheme. Results on a variety of geometric mesh models have demonstrated the effectiveness of our method.

Keywords

Mesh smoothing; Mesh quality improvement; Surface fitting; Maximum inscribed circle

1 Introduction

Surface meshes are frequently used for two primary purposes: 3D visualization and numerical simulation. Accordingly, the "goodness" of a surface mesh is typically measured by its smoothness and angle quality. Good smoothness implies less reconstruction noise or artifact on a surface model while high-quality angles means that no angle should be close to 0° or 180°. The smoothness property is critical in 3D visualization – noisy surface meshes often result in inaccurate interpretation of the models. On the other hand, the angle quality makes a significant impact on the approximation accuracy of numerical solutions. Ideally, meshes with uniform angles would be most desirable in simulation.

In pratice, however, many meshes fail to satisfy one or two of these requirements. For instance, triangular surface meshes generated by iso-surfacing techniques such as the Marching Cube method (Lorensen and Cline, 1987) possess good smoothness but often contain very sharp angles (see Fig. 1a), which make the meshes inappropriate for use in numerical simulation. On the other hand, quadrilateral surface meshes can be simply generated from 3D imaging data by extracting the outward faces of segmented voxels. These

*Corresponding Author (yuz@uwm.edu; Phone: 1-414-229-2960; Fax: 1-414-229-6958).

meshes, if converted into triangular meshes, have very good angles (either 45° or 90°) but are so noisy that it is hard to visualize the structural details on the surface (see Fig. 1b). The goal of this paper is to present a set of efficient algorithms to process an abitrary surface mesh so that both the smoothness and angle quality will be significantly improved.

A wide variety of mesh smoothing algorithms have been proposed in recent years. Most of them, however, are designed just from the graphical point of view, i.e. to improve the mesh smoothness or, in other words, to reduce mesh noise for reconstruction, rendering or visualization purposes (Alliez et al., 2005). Laplacian iterative smoothing is one of the most common and simplest techniques for mesh smoothing (Field, 1988). During each iteration, all the vertices of a mesh are adjusted to the barycenter of the neighboring region. Because of its simplicity, many variants of this method have been developed (Freitag, 1997; Canann, 1998). Taubin (1995) proposed a mesh smoothing method by using a simple, isotropic technique to improve the smoothness of a surface mesh. Desbrun et al. (1999) extended Taubin's work to smooth irregular meshes by using geometric flows. Ohtake et al. (2000) extended the Laplacian smoothing by combining geometric smoothing with parameterization regularization. Peng (2001) gave a denoising algorithm for geometric data represented as a semiregular mesh on the basis of adaptive Wiener filtering. Despite their high speed, these methods often yield significant volume shrinkage and undesired mesh distortion. Another popular smoothing approach is based on the energy minimization technique. Welch and Witkin (1994) described an approach to designing and fairing freeform shapes represented by triangulated surfaces. Kobblet (1997 2000) proposed a general algorithm to fair a triangular mesh with arbitrary topology in R^3 by estimating the curvature for the mesh model. These methods are time-consuming due to the complicated energy functions to be minimized. Recently, feature-preserving mesh smoothing methods (Clarenz et al., 2000; Desbrun et al., 2000; Zhang and Fiume, 2002; Bajaj and Xu, 2003; Jones et al., 2003; Fleishman et al., 2003; Sun et al., 2007, 2008; Li et al., 2009) have drawn more and more attention. Bajaj and Xu (2003) proposed a partial differential equation (PDE)-based anisotropic diffusion approach for processing noisy surfaces and functions defined on surfaces. Although these methods can achieve high smoothness of mesh, they rely on the formation of a "shock" term to preserve details, resulting in significant computational costs. Jones et al. (2003) developed a feature-preserving smoothing algorithm by adopting local first-order predictors statistically defined on triangulated surface meshes. Fleishman et al. (2003) introduced a similar method based on iterative bilateral filtering, a non-linear variation of Gaussian smoothing that weighs sample points based on their similarity to the one being processed. However, finding a set of appropriate parameters is not an easy task. Sun et al. presented fast feature-preserving mesh denoising approaches by normal-filtering and vertex-updating (Sun et al., 2008) and random walks (Sun et al., 2007). Li et al. (2009) adopted the weighted bi-quadratic Bezier surface fitting and uniform principal curvature techniques to smooth surface meshes.

Many of the methods mentioned above are devoted only to improving the smoothness of a mesh. The angle quality of a mesh is equally if not more important, especially when the numerical simulation based on the mesh is taken into consideration (Zhang et al., 2006). The quality of a mesh can be measured by either geometry-dependent (Liu and Joe, 1994) or solution-dependent (Berzins, 1998) criteria. The one we use in our study is by measuring the angles of the triangulated surfaces, a strategy commonly used in the mesh generation and smoothing community (Zhou and Shimada, 2000; Xu and Newman 2005). The mesh quality may be improved by a combination of three major techniques: inserting/deleting vertices (Shewchuk, 2003; Coll et al., 2006), swapping edges/faces (Gooch, 2002; Yamakawa and Shimada, 2009), and moving the vertices without changing the mesh topology (Field, 1988; Wang and Yu, 2009). The last one, also known as mesh smoothing, is the strategy we will explore in the present paper. Different criteria can be utilized to move a vertex, including the

local angle-based methods (Zhou and Shimada, 2000) and global optimization methods (Chen, 2004, 2007; Nealen et al., 2006).

To enhance the smoothness of a mesh and improve its angle quality, we propose an effective mesh smoothing approach based on quadric surface fitting and maximum inscribed circle (MIC) techniques for 3D triangular or quadrilateral meshes. As shown in Fig. 2, the basic idea of this method is to project the center of the MIC at each vertex onto the locally fitted surface and update the current vertex with the projection point to achieve both good mesh smoothness and high angle quality. Numerous experiments on biological and engineering models have demonstrated the effectiveness of our approaches in achieving these two goals. The remainder of the present paper is organized as follows. Section 2 is focused on the details of the mesh smoothing algorithm. The mesh quality improvement is described in Section 3. We present some experimental results and quality analysis in Section 4, followed by the conclusion in Section 5.

2 Mesh Smoothing Algorithm via Surface Fitting

As illustrated in Fig. 2, the input in our method is a triangular surface mesh that may be corrupted with noise. For extremely noisy surface meshes with, the volume-preserving Laplacian smoothing method (Desbrun et al., 1999) with uniform weights is performed before we carry out the following surface fitting-based mesh smoothing. For each vertex, the k-ring neighboring vertices are found according to the topological relationship present in the mesh model. The local surface patch around a point can be approximated with a quadric surface: $S(u, v) = (u, v, h(u, v))$, a parametric representation in a local coordinate system, as shown in Fig. 3, where p is the origin, h-axis is along the normal vector $\boldsymbol{n}$ at p on S, and u-, v- axes are orthogonal vectors in the tangent plane at p. According to the surface theory (do Carmo, 1976), the local shape of a surface around p can be represented with the Darboux system $D(p) = (T_1, T_2, \boldsymbol{n}, k_1, k_2)$, where (T_1, T_2), (k_1, k_2) are principal directions and curvatures, respectively.

We extend this principle to discrete mesh models and construct the local coordinated system for each vertex on the meshes. Since the construction of a local system mainly relies on the h-axis, i.e., the normal vector of the vertex, we first estimate the normal vector for each vertex with the area-weighted averaging method as follows:

$$\boldsymbol{n}_i = \sum_{j=1}^{N} A_{f_j} \cdot \boldsymbol{n}_{f_j}, f_j \in F_i \tag{2}$$

where A_f, $\boldsymbol{n}_f$ are respectively the area and normal vector of face f in F_i. We consider v_i, $\boldsymbol{n}_i$ as the origin and z-axis of the local coordinate system respectively. Then the x- and y- axes of the local coordinate system are arbitrarily chosen in the plane located at v_i and orthogonal to the normal $\boldsymbol{n}_i$. After constructing the local coordinate system, we find all the neighboring vertices for each vertex within the k-ring neighborhood. The optimal value of k depends on local surface geometries. In the implementation of our algorithm, the user can choose either a fixed ring number or an adaptive ring number. For the latter case, the standard deviation of curvatures within a neighborhood (up to 10 layers) is used to adaptively determine the value of k. In other words, the vertices within the chosen k-ring neighborhood should have relatively small curvature variation.

For each vertex, the global coordinates of all the neighboring vertices are transformed to the local coordinates by homogeneous transformation and then the local coordinates obtained are used to fit an analytical quadric surface:

$$h(u, v)=au^2+buv+cv^2+eu+fv+g \tag{3}$$

The classic least square fitting method is adopted to find the coefficients in Eq (3), although more advanced techniques, such as the moving least-square method (Levin 1998), may be used. After the quadric surface is fitted at a vertex, the new position for the vertex must be chosen somewhere on the surface. A simply way to do is by projecting the vertex along its normal vector onto the quadric surface and updating it with the projection position (Wang and Yu, 2009). Because the normal vector is exactly the *z*-axis in the local coordinate system, the local coordinate of the projected point of the vertex should be (0, 0, *g*). The mesh is smoothed by finding the global coordinate with an inverse homogeneous transformation from the local coordinate (0, 0, *g*). This method is very easy to implement and works well in reducing the noise of the surface mesh.

3 Mesh Quality Improvement via Optimized Projection

Although the simple projection method described above can produce smooth meshes, the resulting meshes may contain very skinny angles and sometimes mesh self-intersection may occur. In fact, there are many directions in which a vertex can be projected onto the fitted surface without loss of surface smoothness. To achieve both mesh smoothness and angle quality, however, we must choose a good projection direction among all such candidates. Let us first consider a simpler but analogous problem for angle quality improvement in 2-D meshes (see Fig. 4a): Given a vertex and its adjacent neighbors forming a polygon, finding the new position for the vertex such that the new mesh within this polygon would have the highest angle quality.

Ideally the new vertex should be the one that maximizes the minimal angles in all the triangles in Fig. 4(a). However, finding exactly such a point inside a general planar polygon is nontrivial and is typically approximated by using such methods as Laplacian smoothing (Field., 1988) or angle-based (Zhou and Shimada, 2000) techniques. These approximation approaches, as will be shown shortly, can lead to very low angle quality or even twisted edge connections. In this paper, we adopt another criterion for finding the optimal position of a vertex inside an arbitrary planar polygon. A point v_i' inside a polygon *P* is said to be optimal if the minimal distance from v_i' to the polygon edges is maximized:

$$v_i'=\max_{\{v\in PI\}} D_i(v)=\max_{\{v\in PI\}}\left(\min_{\{w\in PE\}}\|v-w\|\right) \tag{4}$$

where *PI* stands for the interior of the polygon *P* and *PE* for the edges of *P*. The optimization problem above can be thought of as a skeletonization or thinning problem using so-called grassfire model (Blum, 1967): the boundary of an object (a polygon in our case) is set on fire that propagates in the same speed in all directions and the skeleton is the loci where the fire fronts meet and quench each other. Apparently the position that is last quenched inside the polygon is the one that satisfies Eq. (4). Equivalently, the skeletons can be detected with the centers of maximum disks within the polygon (Blum, 1973). Among all such disks, the one with the maximal radius is known as the maximum inscribed circle (MIC) of the polygon and its center would give the optimal position v_i' in Eq. (4).

In general, the skeleton (or medial axis) of a planar polygon is a combination of line and parabolic segments, on which any interior point is equidistant from the polygon boundary

(Lee, 1982). According to (Held and Huber, 2001, 2009), the center of the MIC of a planar polygon is located on one of the joint points of such segments. Fig. 4 illustrates the MIC-based mesh smoothing approach. Fig. 4(a) shows the original 2D meshes centered at v and the surrounding polygon is $P = \{v_1, v_2, \ldots, v_6, v_1\}$. With Held's (2001 With Held's (2009) fast algorithm, the medial axis and the MIC of P can be calculated as shown in Fig. 4(b), in which the medial axis of P is depicted by the red curve segments and c_{mic} is the center of the MIC (in yellow). By choosing c_{mic} as the new vertex of v, the smoothed mesh is given in Fig. 4(c). The minimal and maximal angles in the original mesh are 15° and 149°, which are improved to 28° and 113°, respectively in the smoothed mesh.

Fig. 5 shows a more complicated case. The original mesh with 13 vertices and 24 triangle faces is shown in Fig. 5(a). During the mesh smoothing, the boundary vertices are assumed to remain unchanged. Fig. 5(b) & (c) are the mesh smoothing results with the Laplacian and angle-based smoothing methods. The result with the MIC-based method is given in Fig. 5(d). Clearly our method can still produce decent results even in this extreme case, while the meshes become distorted with the other two methods.

To further quantify the mesh smoothing quality, we consider another example as shown in Fig. 6, consisting of 447 vertices and 892 triangle faces. We compare the mesh smoothing results generated by the Laplacian method, angle-based method and the MIC-based method. The angle and size of the original and smoothed meshes are given in Table 1. Because the boundary vertices are fixed, the triangles incident to the boundary vertices are excluded from the calculation. From the table, we can see that the MIC-based method yields the highest mesh quality compared to the Laplacian method and the angle-based method in terms of angle and size attributes.

The MIC-based method described is well suited for 2-D mesh smoothing. To extend it to 3-D surface mesh smoothing, we need to make some modifications because the polygon surrounding a vertex is most likely non-planar. Let S be the quadric surface fitted with the k-ring neighboring vertices of vertex v_i. We outline the surface smoothing algorithm as follows. For every vertex v_i,

- **Step 1** Fit a plane *pln* using the incident vertices $NV_1(v_i)$ of v_i, (see Fig. 7).
- **Step 2** Project all vertices in $NV_1(v_i)$ onto *pln* and get the new positions $NVP_1(i) = \{vp_j \mid j = 1, 2, \ldots, m\}$.
- **Step 3** For the new projection points that form a planar polygon, solve the 2-D mesh smoothing problem using the MIC-based method. In other words, find a new position, denoted as v_{c_i}, for v_i so that the triangles within the polygon are optimized.
- **Step 4** Project v_{c_i} onto the quadric surface along the normal of *pln*, yielding the final position v_i'.

This above surface smoothing procedure is iterative and terminates when the maximal displacement among all the vertices of the mesh is less than a pre-defined threshold. Fig. 8 gives the final result of the molecule 2CMP. In Fig. 8(c), the minimal and maximal angles of the final mesh are 19.31° and 133.32° and the minimal and maximal sizes are 0.26 and 3.23 respectively.

The method described can achieve high angle quality and smoothness from relatively blunt surface meshes. For meshes with sharp features, the quadric fitting and projection may blur sharp features, due to the use of isotropic k-ring neighborhood. To cope with this problem, we take advantage of anisotropic surface fitting and projection strategies. For each vertex,

we first identify one of its incident faces, having the most similar normal vector to the normal vector of this vertex. The found face is treated as a seed face and then a region growing process is performed by adding the surrounding faces whose normal vectors are close to the normal of the seed face. This process is terminated when the number of faces within the current region reached a pre-defined value. The surface fitting and optimized projection described above are then applied to this anisotropic neighborhood of the current vertex. By using this anisotropic surface fitting and projection strategy, sharp features of mesh can be faithfully preserved.

4 Results and Discussion

All algorithms described have been implemented in Visual C++ and OpenGL, running on a Pentium IV PC with 2.0GHz. A user-friendly GUI has been created encapsulating the implemented algorithms and will be made freely available to the community. Many 3D models have been tested and a number of them are demonstrated below. To compare the performance of different methods, we attempt to generate the best results for each method by adjusting their parameters. The smoothness of all surface meshes below is mainly judged by eyes, while the angle quality is quantitatively measured by histograms (in percentage) with respect to angle degrees (0° – 180°). Additionally, the angle quality is visually illustrated by color map in the zoomed-in meshes – The color of a triangle is green if its minimal angle is greater than 40° and red if its minimal angle is less than 15°. Otherwise, the color is linearly interpolated from red (15°) to green (40°). In our algorithm, two important parameters are the rings number of neighboring vertices k and the maximum vertex displacement threshold T_d.

In Fig. 9, we give the mesh smoothing result for the molecule model 2HAO, chosen from the Protein Data Bank. Shown in Fig. 9(a) is the original mesh generated by using the approach proposed by Yu (2009). This quadrilateral mesh has 228,132 vertices and 228,096 faces. The smoothing result using the direct projection method without MIC-based angle quality improvement is shown in Fig. 9(b). Figure 9(c) shows the final result after two iterations of applying the MIC approach described. This mesh demonstrates visually very good surface smoothness, in contrast to the original cube-like noisy mesh. The angle quality of this mesh is illustrated in a small, enlarged mesh in Fig. 9(c). Fig. 10 gives the angle histograms (in percentage) of the meshes smoothed by the present approach with and without the MIC scheme, where the angles used are obtained by dividing each quadrilateral into two triangles. Apparently, The MIC method can improve the angle quality of the mesh, as compared to the direct projection approach (Wang and Yu, 2009).

Fig. 11 and Fig. 12 demonstrate the mesh smoothing results from 3D imaging data. Fig. 11(a) shows a cross section of the 3D electron tomographic reconstruction of the T-tubule structure. The initial surface mesh, illustrated in Fig. 11(b), was extracted using the automatic image segmentation method (Yu et al., 2008). Fig. 11(c) shows the mesh after applying our surface smoothing algorithm. Fig. 12(a) is an MRI image of the heart and Fig. 12(b) is the segmented surface model made of cube-like noisy quadrilateral meshes. Using the mesh smoothing method proposed; the final mesh is shown in Fig. 12(c). Fig. 13 shows the angle histograms of the two meshes generated by our approach. From both examples, we can see that our mesh smoothing approach can be used in conjunction with image segmentation for geometric modeling of 3D biomedical imaging data.

With three different mesh models shown in Figs. 14, 16, and 18, we aim to compare the algorithms described in this paper with several recent mesh smoothing approaches including Zhou et al.'s (2000),Ohtake et al.'s (2002),Fleishman et al.'s (2003),Jones et al.'s (2003),Sun et al.'s (2007 2008), and Nealen et al.'s (2006) methods. In all the three

examples, our method can effectively smooth the surface noise. The enlarged meshes in Figs. 14,16, and 18 as well as the histograms in Figs. 15,17, and 19 also demonstrate that our approach can achieves significantly higher angle quality, as compared to the aforementioned methods. An additional nice property of our approach is that, in smoothing a surface mesh, our method is capable of preserving sharp features by choosing an anisotropic, curvature-sensitive *k*-ring neighborhood when performing the least square surface fitting. This is demonstrated in Figs. 14 and 16 and especially in Fig. 18, where the sharp corners in the fandisk model are very well preserved. The result by Nealen et al.'s method has the best angle quality in Fig. 14, while the sharp features around the boundaries of "i" and "H" are significantly blurred. These examples show that our methods can perform as effectively as Sun et al.'s methods (2007 2008) in smoothing mesh noise and preserving sharp features, but our approaches can achieve significantly higher angle quality. The computational costs are given in Table 2.

While the histograms seen above have clearly shown the effectiveness of our method in improving mesh angle quality, we make here a comparison among our MIC-based algorithm, the Laplacian optimization method (Nealen et al., 2006), and the 3D version of the angle-based method (Zhou et al., 2000) that was developed specifically to improve the mesh quality for 2D polygonal meshes. Figs. 20 and 21 show the comparison of the mesh improvement on the angel mesh model. In this example, Laplacian optimization method yields the lowest angle quality compared with the other two methods. The angle-based method is able to improve the angle quality to some extent (see Fig. 21b), but our method can achieve higher mesh quality in terms of angle distribution as well as minimal and maximal angles in the mesh (see Fig. 21d).

To demonstrate the fidelity of the smoothed mesh relative to the original surface, the Hausdorff distance between the two meshes is calculated using the software tool called Metro (Cignoni et al., 1998). Fig. 22 shows a detailed comparison of the Hausdorff distance results, where the horizontal axis is the absolute distance value between the smoothed mesh and the original mesh, and the vertical axis is the corresponding histogram (in percentage) with respect to each distance value. From this figure, we can see that our method yields small Hausdorff distances, indicating that our method produces very close surface mesh relative to the original model, in addition to its high efficiency in noise removal and angle quality improvement.

It is worth noting that the MIC strategy used in our algorithm shares similar intuition to the angle-based method (Zhou et al., 2000) in that the new position of a vertex should be as far as possible to the surrounding polygon (1-ring neighbors of the vertex). In fact, these two methods give exactly the same result when the surrounding polygon is a triangle, because the center of the MIC is identical to the intersection of the angular bisectors. When the surrounding polygon has 4 or more edges, however, the behavior of the angle-based method would be unpredictable because the new position of the vertex is the sum (or weighted sum) of the projections on all angular bisectors. With the MIC method, the new position, as the center of the MIC, is unique and is always optimal in terms of the distance to the surrounding polygon.

In the MIC-based method, the input mesh must not be self-intersecting. Otherwise, the MIC is not well defined where self-intersection occurs. In the present work, we do not have theoretical guarantees on the convergence of the algorithm. However, our method is iterative and terminates when meshes in two subsequent iterations are almost the same (measured by the maximum of vertex displacements). The numerous examples tested have shown a fast convergence of our method (typically converged in 2–3 iterations). While we do not have

mathematical proof on the quality improvement of angles, all examples we studied have shown significant angle quality improvement.

5 Conclusions

In this paper, we have developed a novel, effective mesh smoothing method based on surface fitting and optimum vertex projection techniques. The local surface is fitted using a small neighborhood around each vertex so that the fitted surfaces vary smoothly from a vertex to its neighbors and accordingly the projected new positions on the fitted surfaces also vary smoothly, which guarantees the high smoothness of the final mesh in our method. On the other hand, the angle quality and size uniformity of a mesh are achieved by employing an optimization method based on the maximum inscribed circle of a polygon. An added benefit of our approach is its ability of preserving sharp geometric features (such as edges, corners, etc.) by choosing anisotropic neighborhoods in the surface fitting procedure.

Our mesh smoothing algorithm can handle a variety of meshes, including molecular models, imaging data and industrial CAD meshes. As demonstrated in the results, the method proposed can generate meshes with high quality in terms of the smoothness, angle and size attributes. This is useful and critical in such applications as surface reconstruction, visualization, and particularly numerical simulation. As part of our future work, we plan to explore more theoretically the convergence of the MIC-based mesh smoothing. The gurantee on quality improvement of mesh angles using the MIC strategy is also an interesting problem we would like to tackle in the future.

Acknowledgments

We would like to thank Dr. Thouis R. Jones, Dr. Andrew Nealen for providing us with the smoothing results of several surface meshes using their methods, and Dr. Yutaka Ohtake, Dr. Xianfang Sun for making available the source code of their algorithms. The work described was supported in part by an NIH Award (Number R15HL103497) from the National Heart, Lung, and Blood Institute (NHLBI) and by a subcontract from the National Biomedical Computation Resource (NIH Award Number P41 RR08605). The content is solely the responsibility of the authors and does not necessarily represent the official views of the sponsors.

References

Alliez, P.; Attene, M.; Gotsman, C.; Ucelli, G. Recent Advances in Remeshing of Surfaces. In: De Floriani, L.; Spagnuolo, M., editors. Shape Analysis and Structuring. Springer; 2007.

Bade, R.; Haase, J.; Preim, B. Simulation und Visualisierung. 2006. Comparison of fundamental mesh smoothing algorithms for medical surface models; p. 289-304.

Bajaj C, Xu G. Anisotropic diffusion on surfaces and functions on surfaces. ACM Trans Gr. 2003; 22(1):4–32.

Berzins M. A solution-based triangular and tetrahedral mesh quality indicator. SIAM J Sci Comput. 1998; 19:2051–2060.

Blum, H. A transformation for extracting new descriptors of shape. In: Whaten-Dunn, Weiant, editor. Proceedings of the Symposium on Models for the Perception of Speech and Visual Form. MIT Press; Cambridge, Mass: 1967. p. 362-380.

Blum H. Biological shape and visual science (Part I). J Theol' Biol. 1973; 38:205–287.

Canann, SA.; Tristano, JR.; Staten, ML. An approach to combined Laplacian and optimization-based smoothing for Triangular, quadrilateral, and quad-dominant meshes. Proceedings of the 7th International Meshing Roundtable; 1998. p. 479-494.

Carl OG. A mesh-database-independent edge- and face-swapping tool. Int J Number Meth Engng. 2007; 0:1–27.

Chen, L. *Mesh smoothing schemes based on optimal* Delaunay *triangulations*. Proceedings of the 13th International Meshing Roundtable; 2004. p. 109-120.

Chen L. Optimal anisotropic simplicial meshes for minimizing interpolation errors in Lp-norm. Mathematics of Computation. 2007; 76(257):179–204.

Cignoni P, Rocchini C, Scopigno R. Metro: measuring error on simplified surfaces. Computer Graphics Forum. 1998; 17(2):167–174.

Clarenz, U.; Diewald, U.; Rumpf, M. IEEE Visualization 2000. 2000. Anisotropic geometric diffusion in surface processing; p. 397-405.

Coll, N.; Guerrieri, M.; Sellares, JA. Mesh modification under local domain changes. Proceedings of the 15th International Meshing Roundtable; 2006. p. 39-56.

Desbrun, M.; Meyer, M.; Schroder, P.; Barr, AH. Implicit fairing of irregular meshes using diffusion and curvature flow. Proceedings of the 26th annual conference on Computer graphics and interactive techniques; 1999. p. 317-324.

Desbrun, M.; Meyer, M.; Schroder, P.; Barr, AH. Graphics Interface. 2000. Anisotropic feature-preserving denoising of height fields and bivariate data; p. 145-152.

do Carmo, MP. Differential Geometry of Curves and Surfaces. Prentice-Hall; 1976.

Field DA. Laplacian smoothing and Delaunay triangulations. Communications in Applied Numerical Methods. 1988; 4:709–712.

Fleishman, S.; Drori, I.; Cohen-Or, D. Bilateral mesh denoising. Proceedings of ACM SIGGRAPH; 2003. p. 950-953.

Freitag, LA. AMD Trends in Unstructured Mesh Generation. Vol. 220. 1997. On Combining Laplacian and Optimization-based Mesh Smoothing Techniques; p. 37-43.

Held M. VRONI: An engineering approach to the reliable and efficient computation of Voronoi Diagrams of points and line segments. Computational Geometry: Theory and Application. 2001; 18(2):95–123.

Held M, Huber S. Topology-oriented incremental computation of Voronoi Diagrams of circular arcs and straight-line segments. Computer-Aided Design. 2009; 41(5):327–338.

Jones, T.; Durand, F.; Desbrun, M. Non-iterative, feature preserving mesh smoothing. Proceedings of ACM SIGGRAPH; 2003. p. 943-949.

Kobbelt, L. Discrete fairing. Proceedings of the 7th IMA Conference on the Mathematics of Surfaces; 1997. p. 101-131.

Kobbelt L. Discrete fairing and variational subdivision for freeform surface design. The Visual Computer. 2000; 16(3/4):142–158.

Lee DT. Medial axis transformation of a planar shape. IEEE Trans Pattern Anal Mach Intell PAMI-4. 1982; 4:363–369.

Levin D. The approximation power of moving least-squares. Mathematics of Computation. 1998; 67(224):1517–1531.

Li Z, Ma L, Jin X, Zheng Z. A new feature-preserving mesh-smoothing algorithm. The Visual Computer. 2009; 25(2):139–148.

Liu A, Joe B. Relationship between tetrahedron shape measures. BIT. 1994; 34:268–287.

Lorensen, W.; Cline, H. Marching Cubes: a high-resolution 3D surface construction algorithm. Proceedings of ACM SIGGRAPH; 1987. p. 163-169.

Nealen, A.; Igarashi, T.; Sorkine, O.; Alexa, M. ACM GRAPHITE. 2006. Laplacian Mesh Optimization; p. 381-389.

Ohtake, Y.; Belyaev, AG.; Bogaevski, IA. Polyhedral surface smoothing with simultaneous mesh regularization. Proceedings of the Geometric Modeling and Processing 2000; 2000. p. 229

Ohtake, Y.; Belyaev, AG.; Seidel, HP. Vision, Modeling, and Visualization 2002. 2002. Mesh smoothing by adaptive and anisotropic Gaussian filter; p. 203-210.

Peng, J.; Strela, V.; Zorin, D. A simple algorithm for surface denoising. Proceedings of IEEE Visualization 2001; 2001. p. 107-112.

Shewchuk, JR. Updating and constructing constrained delaunay and constrained regular triangulations by flips. Proceedings of the Nineteenth Annual Symposium on Computational Geometry; 2003. p. 181-190.

Sun, X.; Rosin, P.; Martin, R.; Langbein, F. IEEE Trans Visualization and Computer Graphics. Vol. 13. 2007. Fast and effective feature-preserving mesh denoising; p. 925-938.

Sun X, Rosin P, Martin R, Langbein F. Random walks for feature-preserving mesh denoising. Computer Aided Geometric Design. 2008; 7(25):437–456.

Taubin, G. A signal processing approach to fair surface design. Proceedings of the 22nd annual conference on Computer graphics and interactive techniques; 1995. p. 351-358.

Wang, J.; Yu, Z. A novel method for surface mesh smoothing: applications in biomedical modeling. Proceedings of the 18th International Meshing Roundtable; 2009. p. 195-210.

Welch, W.; Witkin, A. Free form shape design using triangulated surfaces. Proceedings of the 21st annual conference on Computer graphics and interactive techniques; 1994. p. 247-256.

Xu, H.; Newman, TS. 2D FE quad mesh smoothing via angle-based optimization. Proc., 5th Int'l Conf. on Computational Science; 2005. p. 9-16.

Yamakawa, S.; Shimada, K. Removing self intersections of a triangular mesh by edge swapping, edge hammering, and face lifting. Proceedings of the 18th International Meshing Roundtable; 2009. p. 13-29.

Yu, Z. A list-based method for fast generation of molecular surfaces. Proceedings of the 31st International Conference of IEEE Engineering in Medicine and Biology Society; 2009. p. 5909-5912.

Yu Z, Holst M, Hayashi T, Bajaj CL, Ellisman MH, McCammon JA, Hoshijima M. Three-dimensional geometric modeling of membrane-bound organelles in ventricular myocytes: Bridging the gap between microscopic imaging and mathematical simulation. Journal of Structural Biology. 2008; 64(3):304–313. [PubMed: 18835449]

Zhang, H.; Fiume, EL. Advances in Modeling, Animation, and Rendering. 2002. Mesh smoothing with shape or feature preservation; p. 167-182.

Zhang Y, Xu G, Bajaj C. Quality meshing of implicit solvation models of biomolecular structures. Computer Aided Geometric Design: Special Issue on Applications of Geometric Modeling in the Life Sciences. 2006; 23(6):510–530.

Zhou, T.; Shimada, K. An angle-based approach to two-dimensional mesh smoothing. Proceedings of the 9th International Meshing Roundtable; 2000. p. 373-384.

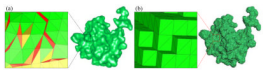

Figure 1.
(a) An example of triangular meshes generated by the marching cube method, showing very good smoothness but low angle quality. The color of a triangle is green if its minimal angle is greater than 40° and red if its minimal angle is less than 15°. Otherwise, the color is linearly interpolated from red (15°) to green (40°). (b) An example of quadrilateral meshes having uniform and regular squares but showing very noisy surfaces

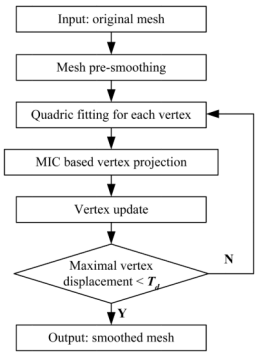

Figure 2.
The flowchart of mesh smoothing method

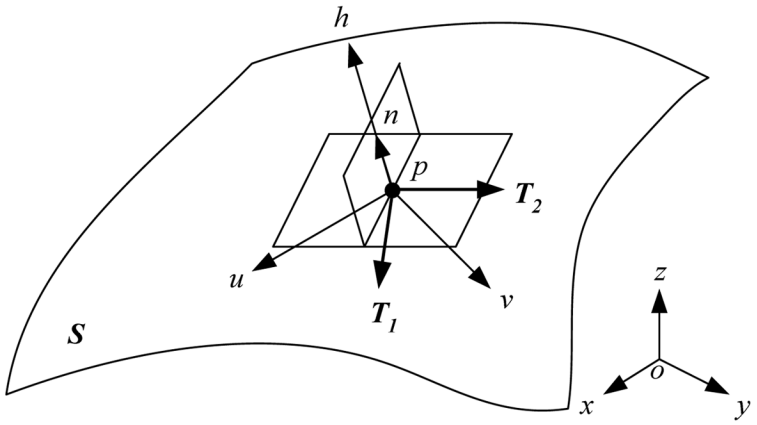

Figure 3.
An illustration of the local coordinate system on a surface

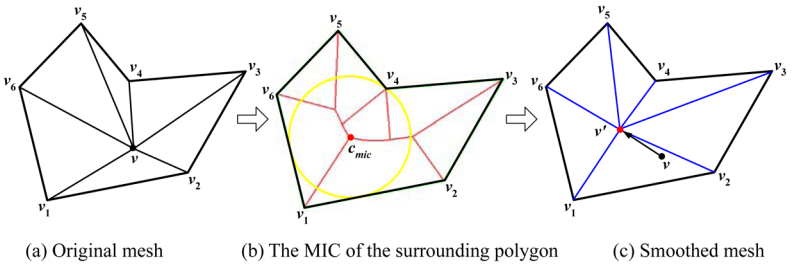

Figure 4.
Illustration of the mesh smoothing approach based on the maximum inscribed circle (MIC)

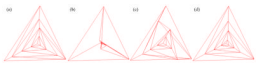

Figure 5.
Mesh smoothing for (a) a 2D mesh, using (b) the Laplacian method, (c) the angle-based smoothing method, and (d) The MIC-based smoothing method

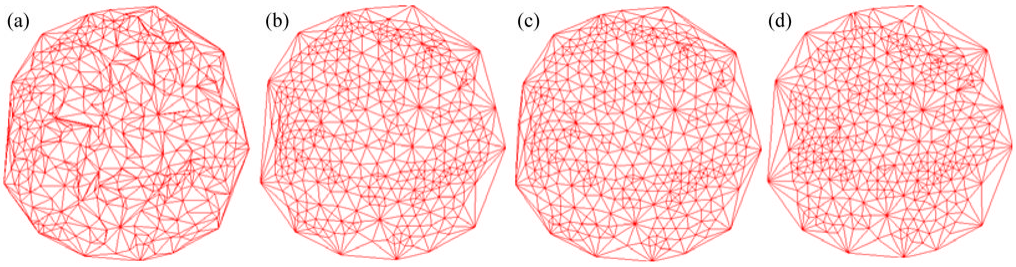

Figure 6.
Mesh smoothing for (a) a 2D mesh, using (b) the Laplacian method, (c) the angle-based smoothing method, and (d) the MIC-based smoothing method

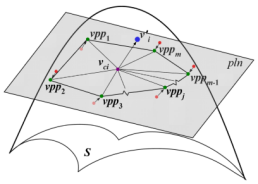

Figure 7.
Illustration of the surface projection algorithm

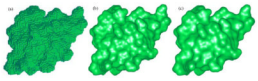

Figure 8.
Mesh smoothing result of (a) the original mesh with (b) 1 iteration, and (c) 2 iterations

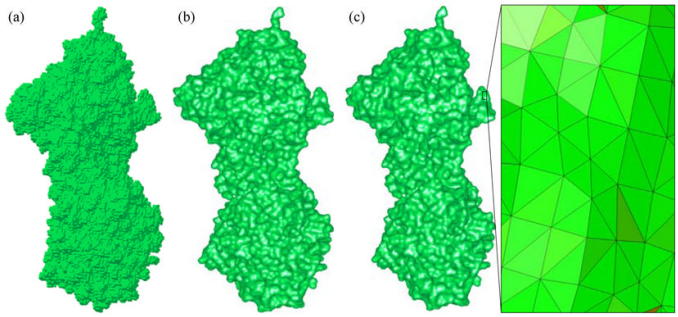

Figure 9.
Mesh smoothing of the molecule 2HAO. (a) The original mesh. (b) The mesh smoothed by our method before the MIC-based quality improvement. (c) The mesh smoothed by our method after quality improvement

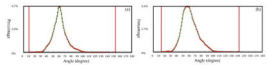

Figure 10.
The angle histograms of the meshes shown in Fig. 9(b) and (c). (a) Before the MIC-based quality improvement. (b) After the mesh quality improvement. The angle bounds of (a) and (b) are [10.12°, 153.81°] and [14.56°, 143.62°], respectively. It is clear that the MIC-based method significantly improves the angle quality of the mesh. The red bars in the histograms indicate the minimal and maximal angles in the meshes

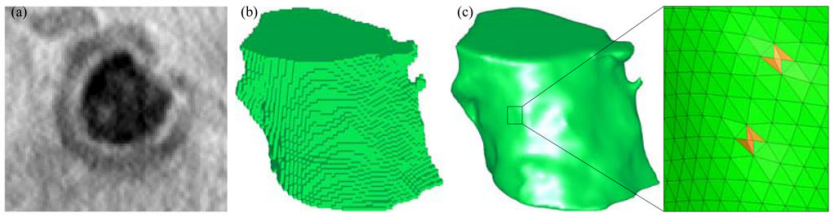

Figure 11.
Mesh smoothing of the surface model extracted from 3D imaging data. (a) An electron tomographic reconstruction of t-tubule structures (courtesy of Dr. Masahiko Hoshijima, UC-San Diego). (b) Segmented cube-like noisy surface mesh. (c) The mesh smoothing result with our method

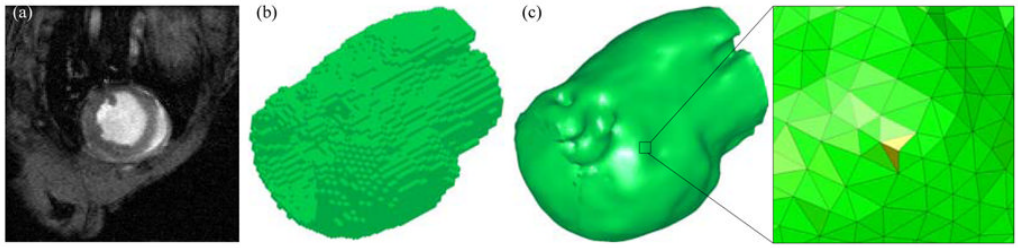

Figure 12.
Mesh smoothing of the surface model extracted from MRI imaging data. (a) A cross section of the MRI data of the heart (courtesy of Dr. Masahiko Hoshijima, UC-San Diego). (b) Segmented cube-like surface mesh. (c) The mesh smoothing result with our method, showing more informative details on the surface

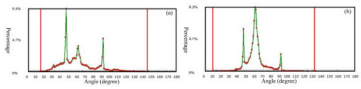

Figure 13.
The angle histograms of the two meshes in Fig. 11(c) and Fig. 12(c). The angle bounds of (a) and (b) are [15.41°, 148.30 °] and [11.27 °, 130.38 °], respectively

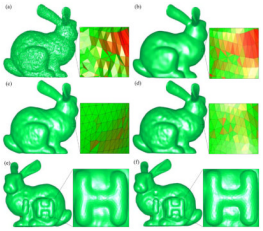

Figure 14.
Smoothing results of the noisy Stanford bunny mesh with “i” and “H” as shown in (a). (b) Ohtake et al.’s method with n = 8 (Ohtake et al. 2002). (c) Nealen et al.’s method with Laplacian matrices of uniform weights, and $f = 0$, $s = 0.3$ (Nealen et al., 2006). (d) Our method with $N = 20$ and $T_d = 5\%$ X the mean edge length of mesh. The color of a triangle is green if its minimal angle is greater than 40° and red if its minimal angle is less than 15°. Otherwise, the color is linearly interpolated from red (15°) to green (40°). (e) The front view of Nealen et al.’s smoothing result. (f) the front view of our smoothing result. Note that sharp features, such as the boundaries of “i” and “H”, are significantly blurred in (e).

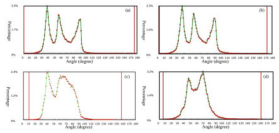

Figure 15.
The angle histograms of the meshes in Fig 14(a–d). The angle bounds of (a), (b), (c) and (d) are [0.05°, 175.23°], [1.32°, 170.42°], [9.00°, 158.65°], and [7.81°, 161.39°], respectively

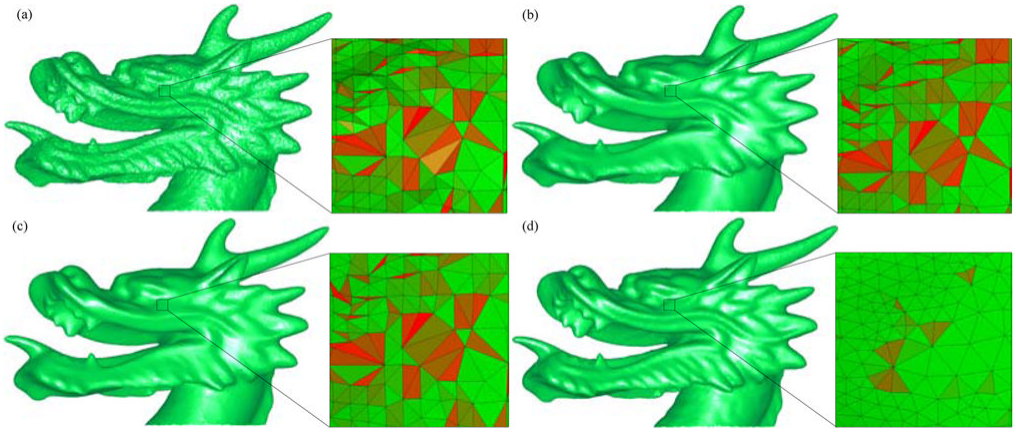

Figure 16.
Smoothing results of the dragon head mesh. (a) Original noisy mesh. (b) Jones et al.'s method (Jones et al., 2003), where σ_f =4, σ_g = 1. (c) Sun et al.'s method (Sun et al., 2007), where $n1$ = 5, $n2$ = 25, T = 0.55. (d) Our method (k = 3, T_d = 4% X the mean edge length of mesh)

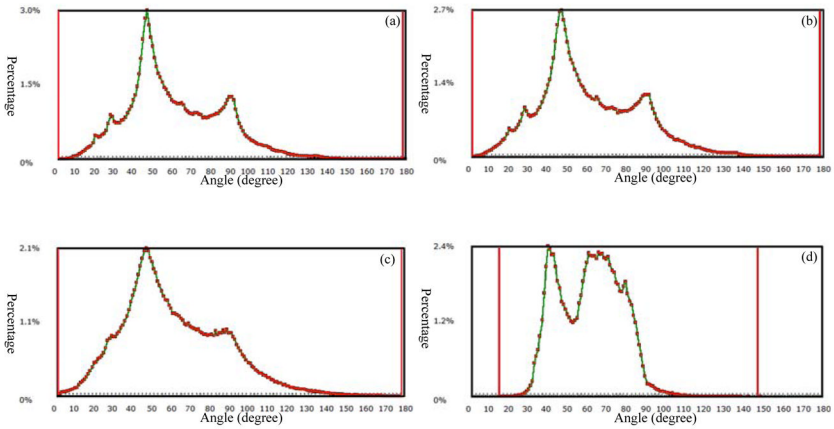

Figure 17.

The angle histograms corresponding to the meshes in Fig. 16(a–d). The angle bounds of (a), (b), (c) and (d) are [0.10°, 177.54°], [0.29°, 177.01°], [0.13°, 177.91°], and [13.82°, 147.02°], respectively

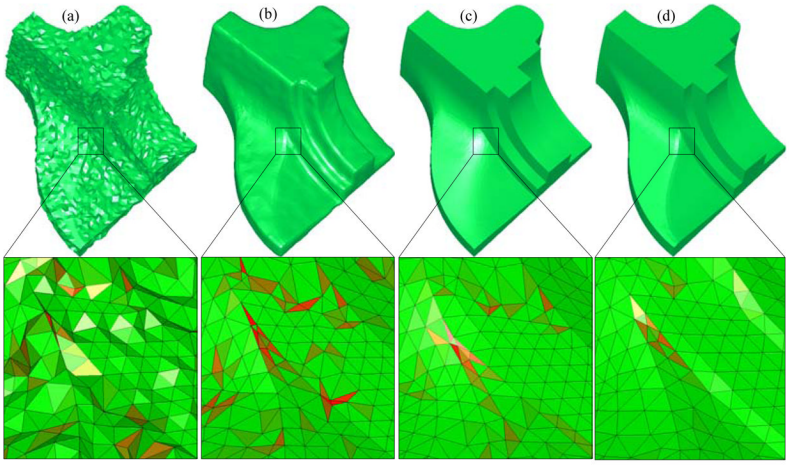

Figure 18.
Smoothing results of the fandisk mesh. (a) Original noisy mesh. (b) Fleishman et al.'s method (Fleishman et al., 2003), where $n = 5$. (c) Sun et al.'s method (Sun et al., 2008), where $n1 = 4$ and $\beta = 8$. (d) Our method ($N = 10$, T_d = 6% X the mean edge length of mesh)

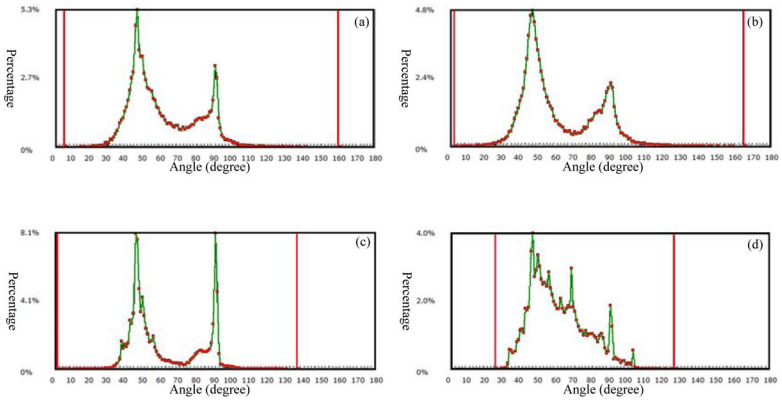

Figure 19.
The angle histograms of the meshes in Fig. 18(a–d). The angle bounds of (a), (b), (c) and (d) are [6.02°, 159.53°], [2.41°, 167.82°], [1.29°, 137.89°], and [25.29°, 127.16°], respectively

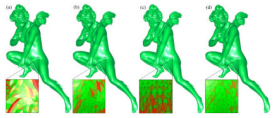

Figure 20.

Smoothing results of the angel mesh. (a) Original surface mesh. (b) The 3D implementation of Zhou et al.'s method (Zhou et al., 2000), where $n = 3$. (c) Nealen et al.'s method with Laplacian matrices of uniform weights, and $s = 1.0$ (Nealen et al., 2006). (d) Our method ($k = 3$, $T_d = 4\%$ X the mean edge length of mesh). Again, the color of a triangle is green if its minimal angle is greater than 40° and red if its minimal angle is less than 15°. Otherwise, the color is linearly interpolated from red (15°) to green (40°)

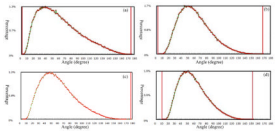

Figure 21.
The angle histograms of the meshes in Fig. 20(a–c). The angle bounds of (a), (b), (c) and (d) are [1.24°, 173.49°], [3.83°, 167.06°], [0.62°, 167.06°] and [11.38°, 152.74°], respectively

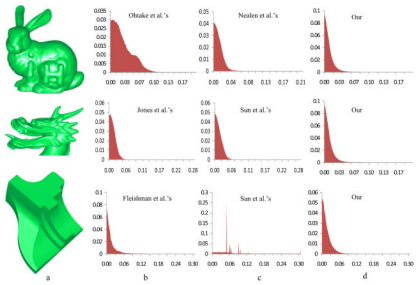

Figure 22.

The histograms show the Hausdorff distances between the meshes obtained by each approach and the original meshes. The horizontal axis is the error (absolute distance value) from the denoised mesh to the original mesh, and the vertical axis is the corresponding percentage to each error value. (a) Data set, and the corresponding Hausdorff distances in (b), (c), (d) by Ohtake et al.'s, Jones et al.'s, Fleishman et al.'s, Nealen et al.'s, Sun et al.'s and our methods, respectively

Table 1

Angle and size attributes of the mesh in Fig. 6

	Angle			Size	
	Min	**Max**	**20° < α < 160°**	**Min**	**Max**
Original mesh	0.14°	179.22°	77.7%	0.001	0.931
Laplacian	8.78°	164.35°	89.7%	0.003	0.182
Angle-based	12.23°	146.98°	95.3%	0.004	0.164
MIC-based	18.62°	126.71°	98.7%	0.011	0.105

Table 2

The comparison of computational time (sec.)

	Fandisk v/t: 7123/14242	**Bunny v/t: 35325/70646**	**Dragon v/t: 100056/199924**
Ohtake et al.'s	-	3.013	-
Fleishman et al.'s	0.071	-	-
Jones et al.'s	-	-	61.237
Sun et al.'s	0.133	-	2.018
Nealen et al.'s	-	2.486	-
Our	1.175	5.889	17.324