Mesh Simplification with Smooth Surface Reconstruction

O. Volpin[1], A. Sheffer[1], M. Bercovier[2], L. Joskowicz[1]

Abstract

In this work a new method for surface reconstruction is introduced, which simplifies the original object mesh and then builds a smooth surface on top of it. It handles original models containing free-form faces, and is not restricted to initial triangular or polygonal faces. The method is illustrated by building of several free form surfaces from an arbitrary topology mesh.

Keywords: Clustering, Subdivision, Finite-Element Method(FEM), Piecewise Bézier Surface, Limited Curvature Regions, Thin Plate Energy, Quadrilateral Mesh.

1 Introduction

Manipulation and display of geometric objects are the two most common operations in graphics and geometric modeling. These operations become cumbersome and time-consuming when the number of faces used to model the objects increases, as is common in many applications. Often, the large number of faces does not reflect the real object complexity, but is the result of the algorithms used to construct their shapes. For example, most algorithms for creating polyhedral surfaces from data sampled on a regular 3D grid produce meshes with a large number of small faces. Similar problems are encountered in data reconstruction from laser range scanners.

Model simplification methods seek to produce smaller models by creating representations with as few faces as possible while maintaining the object topology and deviating as little as possible from the original model geometry.

[1]Institute of Computer Science, The Hebrew University, Jerusalem 91904, Israel.
[2]DER Genie Informatique, Pole Universitaire Leonard de Vinci, France.

Construction of a smooth surface based on the nodal faceted representation is often a necessary step for both model analysis and display.

In this work a new method for surface reconstruction is introduced, which simplifies the original object mesh and then builds a smooth surface on top of it. The proposed algorithm produces a smooth surface and maintains the object topology. Due to the subdivision into limited curvature regions, the energy functional is stabilized and the result provide a smoother surface. It constructs a quadrilateral simplified mesh which is more suitable for analysis; it handles original models containing free-form faces, and is not restricted to initial triangular or polygonal faces; thanks to the use of finite element techniques complex surface structures including non-manifold can be treated; the deviation of the reconstructed surface from the original can be estimated and bounded. This method is most suitable for CAD/CAM and FEA (finite element analysis) applications.

2 Previous Work

As noted by Eck [5]: "The difficulty of dealing with complicated models is evident by the extensive recent research on the topic".

Many works on mesh simplification were written in the context of fast rendering and display of polygonal models (e.g. [6], [9]). Those works provide algorithms that are both very fast and allow big reduction in the number of elements used. However such methods are usually not applicable for CAD purposes since the original model topology is not preserved. Those methods produce a triangular description of the models which is not suitable for analysis and most CAD/CAM systems.

Several methods for mesh simplification and reconstruction are based on the topology of the original mesh. For example, Turk [16] proposes a method in which polygonal surfaces are "re-tiled" by triangulating a new set of vertices that replaces the original one using mutual tessellation. Schmitt [13] uses a top-down approach to simplify a regular rectangular mesh by refining an approximation mesh of piecewise patches until it is within a given error bound of the original mesh. Kalvin and Taylor [8] present a domain-independent method for simplifying polygonal meshes based on a bounded approximation criterion which produces a simplified mesh within a prescribed error-bound from the original and uses a subset of the original mesh vertices. These topology-based methods are restricted to polygonal meshes and some like Turk restrict themselves to triangular meshes only.

Such algorithms are relatively fast but are very dependent on the initial mesh. Another disadvantage of this type of algorithms is that they do not solve some of the intersection problems that arise during re-meshing which in some cases might result in self-intersecting models, starting from a valid original model.

A different type of algorithms tries to capture physical properties of the mesh using different energy functionals. This approach was introduced in Hoppe [7], where an energy based mesh optimization method was suggested. In Eck and Hoppe [4] a method for reconstructing a G^1 surface was presented. The method uses a subdivision method for constructing the smooth surface from the quadrilateral mesh (Peters [12]). To achieve the desired tolerance, a special reconstruction procedure was used. In the first stage of the algorithm, the construction of the quadrilateral mesh, a big reduction in the number of elements can be achieved, however due to the subdivision method used for the surface construction the number of elements is at least quadrupled to achieve the desired smoothness.

Those algorithms result in much smoother and better meshes, but many of these require non-linear optimization and all take much more time than the simpler methods mentioned above. These methods use global energy functionals which are sensitive to small areas of high curvature deviation, which can cause global disturbances.

Non of the methods above addresses the issues of initial non-linear surfaces or non-manifold topology.

3 Algorithm overview

In this work an algorithm that combines the advantages of both approaches is presented. The method simplifies the original object mesh and then builds a smooth surface on top of it. The proposed algorithm produces a smooth surface and maintains the object topology. Due to the subdivision into limited curvature regions, the energy functional is stabilized and the result provide a smoother surface.

The main stages of the algorithm are:

1. Subdivision of the surface into restricted curvature deviation regions, using a topology based method with a bounded error. The regions are constructed based on clustering of the original mesh faces. (Figure 1(a) to (b))

2. Generation of a boundary conforming finite element quadrilateral mesh of the regions. The element size for meshing is given as a parameter of the final number of elements and the deviation tolerance. (Figure 1(c))

3. Construction of a smooth surface over the quadrilateral mesh using the plate energy method. Inter-patch approximation of G^1 continuity constraints are added to achieve the desired smoothness. (Figure 1(d))

The stages are explained in detail in the following sections and demonstrated in figure 1.

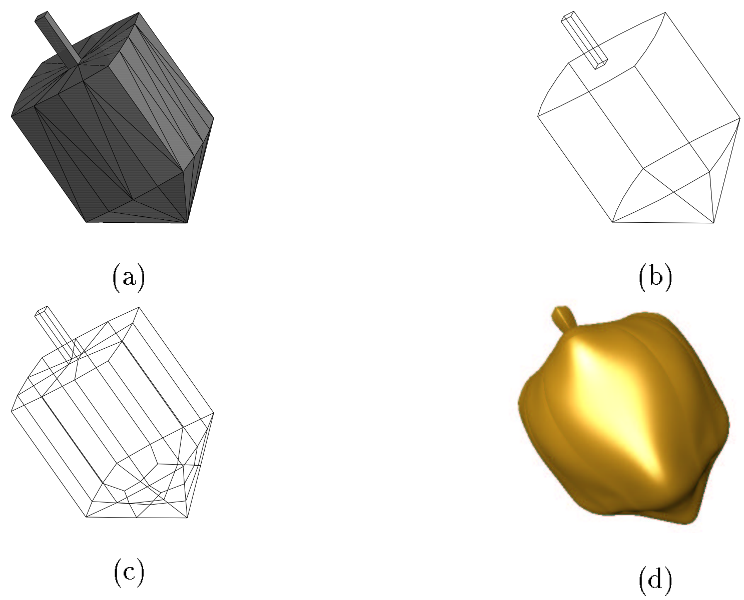

Figure 1: The stages of the surface reconstruction demonstrated on a simple spinning top example. (a) Original model (containing linear triangle and quad elements). (b) Subdivision into regions. (c) Quadrilateral mesh of the model regions (d) The reconstructed smooth surface.

4 Regions construction

The first step of the model simplification is the subdivision of the initial surface **S** into simple regions of restricted distance and curvature deviation.

The subdivision is based on clustering of planar facets into maximal clusters with topological connectivity, maintaining a set of user-controlled constraints. When the model contains only planar faces, the clustering algorithm is applied directly on the model, with the resulting clusters defining the subdivision regions. When the model contains non-planar faces, they are approximated using a tolerance based linear faceting to simplify the surface analysis. The clustering is then applied on the resulting facets, and an additional analysis stage is performed to group the original faces based on the facets clusters.

In both cases the faces of each region are then merged into a single *topological* virtual face using the virtual B-Rep [14], allowing mesh generation over the region as a single surface.

4.1 Clustering

The set of linear facets is subdivided into clusters using a greedy type method. As a result, clustering is extremely fast and can be applied on very large sets of facets. The clustering procedure is not restricted to closed or regular meshes (sets of facets).

In the description of the clustering the following terms are used. A **cluster** is defined as a set of connected facets and the **boundary** of a cluster as the set of edges lying on its perimeter. The facets that contain a boundary edge, but are not yet part of the cluster are called its **border facets**.

The creation of a cluster begins with a selection of an initial “seed” facet that grows through a process of accretion; **border facets** (i.e. facets adjacent to the current cluster boundary) are merged into the evolving cluster if they satisfy the required clustering criteria. A cluster eventually stops growing when there are no more facets on its boundary that can be merged. The process is demonstrated in figure 2.

It is a “greedy” method; it doesn’t backtrack or undo any merging once it is done. The “seed” facets for growing the clusters are selected randomly from the set of unclustered facets on the mesh, and the process ends when all the mesh facets belong to clusters. The boundary facets for each cluster come from facets not already belonging to another cluster.

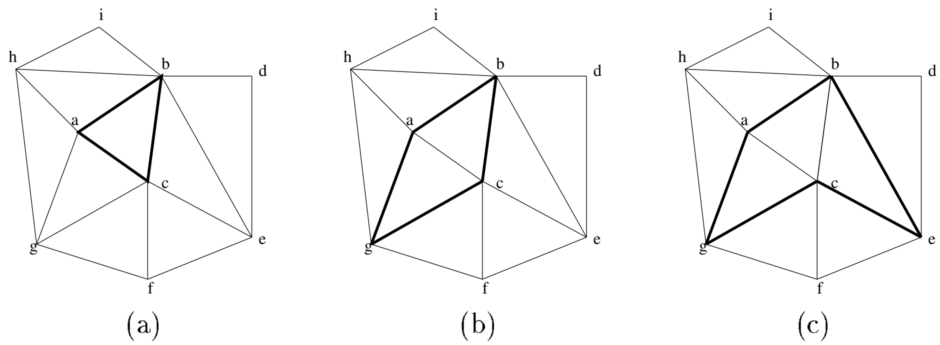

Figure 2: A visualization of the clustering procedure. (a) An initial cluster containing only the "seed" facet f_{abc}. The boundary of the cluster contains edges $\bar{ab}$, $\bar{bc}$ and $\bar{ca}$. The border facets (facets sharing a boundary edge with the cluster) are f_{acg}, f_{ahb}, f_{bec}. (b) The cluster after the addition of the border facet f_{acg}. The boundary of the cluster contains edges $\bar{ab}$, $\bar{bc}$, $\bar{cg}$ and $\bar{ga}$. The border facets are f_{ahb}, f_{bec}, f_{cfg} and f_{gha}. (c) The cluster after the addition of the border facet f_{bec}.

4.2 Clustering Criteria

The fundamental growing step of the algorithm is the expansion of a cluster boundary through the merging of a border face. A border face f_b is accepted into a cluster c if it satisfies a set of merging criteria. The criteria to satisfy are chosen by the user from the set described below.

For each cluster an approximating plane is defined as the weighted average of the planes of the facets in cluster with the weight based on the facet's area. Defining each plane pl as the tuple (N, d), where N is the plane normal and d is the nearest distance from the plane to the center of coordinates, we have for each point (x, y, z) on the plane

$$N_x x + N_y y + N_z z + d = 0$$

where N_t stands for the t coordinate of N. Defining the plane for each face f_i in the cluster as $p_i = (N_i, d_i)$ we get for the cluster c

$$p_c = (N_c, d_c) = (\frac{1}{m \times area(c)} \sum_{i=1}^{m} area(f_i) N_i, \frac{1}{m \times area(c)} \sum_{i=1}^{m} area(f_i) d_i).$$

Using those definitions the set of applicable merging criteria is defined as:

1. **Bounded angle between adjacent facets** - the angle between the plane of the new facet f_b and the planes of facets in c that share a common edge with it, is below a user defined angle ϕ.

2. **Bounded distance between facets and the cluster plane** - the normal distance between the vertices of the new cluster facets $\{f_i\}_{i=0}^{m} \cup f_b$ and the plane of the new cluster $p_{c \cup f_b}$ is below a given distance tolerance δ. It is sufficient [15] to check the distance of the vertices of f_b from the existing cluster plane p_c; if this distance is above the tolerance δ, compute the plane $p_{c \cup f_b}$ and check the distance from all cluster vertices to it. If the new plane doesn't satisfy this criteria, the facet f_b is not added to to the cluster.

3. **Bounded angle between the planes of facets and the cluster plane** - The angle between the planes $\{p_i\}_{i=0}^{m} \cup p_{f_b}$ of the facets $\{f_i\}_{i=0}^{m} \cup f_b$ and the plane $p_{c \cup f_b}$ of the new cluster is below a user defined angle ψ. Similar to criteria 2 above, the plane $p_{c \cup f_b}$ needs to be recomputed only if the angle between p_{f_b} and p_c is above the threshold ψ. As in criteria 2, if the new plane doesn't satisfy this criteria, the facet f_b is not added to to the cluster.

These criteria guarantee construction of cluster regions with restricted plane deviation and with restrictions on both local and global curvature.

Criteria 1 (bounded local angle) prevents absorption of prominent minor details into the the surrounding region (like the lips or nose in figure 6). The use of criteria 2 (distance from plane), bounds the normal deviation of the region's mesh from the initial surface, and hence controls the distance between the reconstructed and initial surfaces. Criteria 3 guarantees regions of relative smoothness, which is a pre-condition for construction of smooth surface afterwards. High curvature deviation will create high oscillation of the surface.

In the spinning top example (figure 1), the values used were $\phi = 40°$, $\delta = 1.5$ (for model side length of 22) and $\psi = 40°$.

A subset of the criteria can be used when some of the restrictions are required.

4.3 Non Linear Faces - Linearization and Analysis

When the model contains non-linear faces, a set of planar facets approximating them is constructed. This is done in order to reduce the complexity of the clustering process to one of considering linear surfaces only. The faceting is used solely to approximate the faces and to allow simple clustering of the facets. The only requirements on it are of conformity and constrained deviation (distance and angle) from the original face surface. The faceting constructed is the same faceting used for shaded object display and other computations on the object, as done in commercially available packages (e.g. [19]). Hence it's computations is not an overhead of the clustering algorithm.

After the merging of the facets into clusters is completed, the clusters are analyzed to group the original faces based on the facets clusters. (as described in detail in [15]).

5 Mesh Generation

After the regions are formed, a surface meshing algorithm is used to generate a boundary conforming finite element quadrilateral mesh $\mathbf{Q}$ of the region virtual faces. The meshing is done using the paving algorithm described in [2]. It handles any face topology or geometry structure. This algorithm is widely used in finite element analysis. In this work it is introduced as a surface subdivision tool. The element size for the edges and faces mesh is given as a parameter of the final number of elements and the deviation tolerance.

A simple example of the surface mesh is shown in figure 1(c). Figures 6(c) and (e) show the effect of different element size on the mesh and on the final surface.

6 Smooth surface construction

Once the mesh $\mathbf{Q}$ is generated, the final step of the procedure is to build a smooth surface $\tilde{\mathbf{S}}$ approximating the mesh.

Composite free form surfaces, that approximate a given mesh are constructed by assembling tensor product type patches (such as non-uniform B-splines or Bézier). This approach is based on the concept of Thin Plate Energy, i.e. functionals that are not directly derived from the patch standard parameterization, and depend on the definition of a local reference plane for each element. Hence each mesh element Q_k is initially approximated by a planar quadrilateral element Ω_k. The reference parametric space $\mathbf{\Omega}$ for the mesh is then defined as the union of the planar quadrilaterals, each of which is an image of the reference domain $\Omega' = [0,1] \times [0,1]$ under a bilinear transformation.

$$\mathbf{\Omega} = \cup_{k=1}^{N} \Omega_k$$

For each Ω_k, a local coordinate system is introduced. The displacement in the normal direction is defined as in the classical FEM (Finite Element Method) [18]. This displacement defines the energy functional at the element level. The resulting surface minimizes the global energy functional, which is built as the sum of the local functionals. In the global coordinate system, patches of the resulting surface can be written in the form:

$$\tilde{S}_k = \sum_{i=0}^{n} \sum_{j=0}^{m} P_{i,j}^k \varphi_{i,j}(u,v), \quad (u,v) \in [0,1] \times [0,1],$$

where $P_{i,j}^k$ are nodal displacements in the global coordinate system, corresponding to the patch k and $\varphi_{i,j}$ are the shape functions.

Additional geometrical conditions must be imposed in order to obtain a smooth surface, as will be discussed in subsection 6.2.

6.1 Definition of the local coordinate system and the energy functional

Consider now one element Q_k of the mesh $\mathbf{Q}$. The first step is to approximate this non-planar element by a planar quadrilateral Ω_k as close as possible to the original one, such that the normal to the constructed quadrilateral is estimated normal to the initial element. The plate energy will be defined relative to this new plane. A local coordinate system is defined so that the Z-direction coincides with the normal direction of the planar element Ω_k. By analogy with the thin plate approximation of shells for surface construction, one first defines a local functional at the element level E_k and the global energy functional for the whole mesh is taken as the sum of the local functionals over all mesh elements.

The local patch energy is constructed in two steps. The functional formulation is given on the plane quadrilateral related to the underlying mesh. The unknown displacement field is decomposed into a bending displacement $\bar{z}(\hat{x}, \hat{y})$ normal to the local plane defined by Ω_k, and into a local $(\hat{x}, \hat{y})$ plane displacement. Construction of the local energy functional is described in details in [17]. The energy functional over Ω_k is defined as:

$$
\begin{aligned}
E_k = \alpha \int\int_{\Omega_k} \left((\frac{\partial \bar{z}}{\partial \hat{x}})^2 + \left(\frac{\partial \bar{z}}{\partial \hat{y}}\right)^2 \right) d\hat{x} d\hat{y} + \\
\beta \int\int_{\Omega_k} \left(\frac{\partial^2 \bar{z}}{\partial \hat{x}^2} + \frac{\partial^2 \bar{z}}{\partial \hat{y}^2}\right)^2 - 2(1-\nu) \left(\frac{\partial^2 \bar{z}}{\partial \hat{x}^2} \frac{\partial^2 \bar{z}}{\partial \hat{y}^2} - \left(\frac{\partial^2 \bar{z}}{\partial \hat{x} \partial \hat{y}}\right)^2 \right) d\hat{x} d\hat{y} + \\
\gamma \int\int_{\Omega_k} \left(\left(\frac{\partial^3 \bar{z}}{\partial \hat{x}^3}\right)^2 + 3 \left(\left(\frac{\partial^3 \bar{z}}{\partial \hat{x}^2 \partial \hat{y}}\right)^2 + \left(\frac{\partial^3 \bar{z}}{\partial \hat{x} \partial \hat{y}^2}\right)^2 \right) + \left(\frac{\partial \bar{z}^3}{\partial \hat{y}^3}\right)^2 \right) d\hat{x} d\hat{y}
\end{aligned}
\quad (1)
$$

where $\bar{z}(\hat{x}, \hat{y})$, as defined above, is the normal displacement relative to the local element plane and ν is the Poisson's coefficient, $0 \leq \nu \leq 0.5$.

Depending on the obtained mesh and the final design goal, different α, β and γ are applied. The correlation of the α, β and γ parameters is used to obtain the desired surface properties as a combination of minimal surface area, minimal curvature and minimal curvature change. Increasing α makes the surface more planar and "tight". β affects the surface smoothness, and increasing γ makes the surface more rounded and "fat". In the spinning top example (figure 1) the values used were $\alpha = 0.3$, $\beta = 0.1$, $\gamma = 0.6$.

It is important to note that the quadratic functional is independent of the actual position of the plane $O\hat{x}\hat{y}$ (Ω_k) and of *any underlying parameterization.*

6.2 Discrete G^1 continuity conditions

In order to obtain a G^1 condition at a common n-curve vertex, it is necessary that the tangents to the boundaries of all n patches sharing that vertex lie in the same plane.

In the present approach one fixes the relations between tangents at the vertex instead of fixing the direction of the normal, which is then determined according to the minimization of the energy functional. Globally smooth surface (without "bumps" near the vertices) can be obtained since the geometric conditions at the vertices do not fix a priori the resulting normal direction (such a procedure has better shape preserving properties as discussed in [10]). Since the considered energy functionals are closely related to the structure of the mesh, the resulting surface will correspond to the geometry of the given mesh.

In the present approach the following pseudo C^1 linearized smoothness conditions are imposed along the common edge:

$$
q_{nj} - q_{n-1,j} = p_{1j} - p_{0j} \quad j = 1, \ldots, n-1 \quad (2)
$$

The conditions are visualized in figure 3.

This does not result in G^1 surface in general. However the deviation of the normals for two neighboring patches is bounded by its maximal "inconsistency" at vertices. The deviation is relatively small when near the middle of the segment of parameterization. Moreover, since at the ends of the segments the G^1 condition results from the choice of the geometric conditions at the vertices, it will compensate relatively large deviations at the vertices [10].

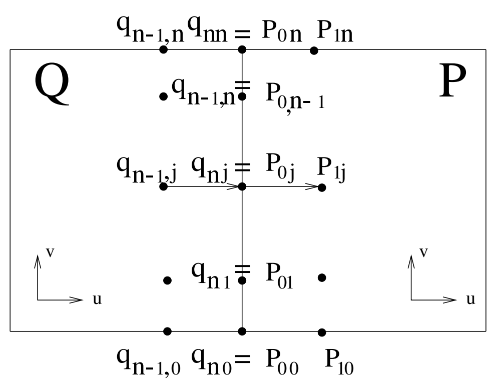

Figure 3: Smoothness condition along common edge.

7 Error Bounds

In this section it is shown that the deviation of the final surface $\tilde{\mathbf{S}}^q$ (where q stands for the solution restricted to the subspace defined by the FE (Finite Element) mesh $\mathbf{Q}$) from the original surface mesh $\mathbf{S}$ is bounded in L^2 norm by ϕ, where ϕ is a quadratic function of the meshing element size and the local curvature of the original surface $\mathbf{S}$ over the area of the new mesh elements. The coefficients of ϕ are constants depending solely on the original surface quality. Hence, the deviation tolerance can be controlled directly by the mesh element size and the maximal curvature (angle between surface and plane) of the region clusters.

As proven in Ciarlet [3], given a boundary value problem such as the above one, whose solution $\mathbf{S}$ is sufficiently smooth, and $\pi_q\mathbf{S}$ the polynomial bilinear interpolate over the original mesh ($\pi_q\mathbf{S}$ is well-defined since $\mathbf{S}$ is assumed to be sufficiently smooth), the following inequality holds :

$$||\mathbf{S} - \pi_q \mathbf{S}||_{L^2} \leq C_1 l^2 |\mathbf{S}|_{L^2} \tag{3}$$

where C_1 is a constant independent of the mesh, l is the maximal length of an element edge in the FE mesh $\mathbf{Q}$, and $|\mathbf{S}|_{L^2}$ is the L^2 semi-norm of the second derivatives of $\mathbf{S}$ over a mesh element. Now let $\tilde{\mathbf{S}}^q$ be the computed surface simplified solution. Then, by the present construction of the smooth surface over the FE mesh

$$||\tilde{\mathbf{S}}^q - \pi_q \mathbf{S}||_{L^2} \leq C_2 l^2$$

where C_2 depends on the original surface only [1]. Hence by equation 3 the resulting construction satisfies:

$$||\tilde{\mathbf{S}}^q - \mathbf{S}||_{L^2} \leq ||\tilde{\mathbf{S}}^q - \pi_q \mathbf{S}||_{L^2} + ||\mathbf{S} - \pi_q \mathbf{S}||_{L^2} \leq (C_1 |\mathbf{S}|_{L^2} + C_2) l^2$$

and so we have the deviation of the constructed surface $\tilde{\mathbf{S}}^q$ from the original surface $\mathbf{S}$ bounded by a quadratic function of the size of the FE mesh $\mathbf{Q}$ elements and the curvature of the clustered regions.

8 Examples

The examples below demonstrate the mesh simplification and smooth surface reconstruction processes described in this paper.

Figure 4 shows a reconstruction of an initial free form (NURB) surface. The initial data in figure 4(a) contains 213 NURB and cylindrical surfaces. The model is subdivided into regions based on angle tolerance only. The regions are shown in figure 4(b). Figure 4(c) shows a quadrilateral conformal mesh of the regions containing 125 elements. The final reconstructed surface based on the mesh is shown in figure 4(d). The surface is based on quartic Bézier piecewise polynomial patches.

Figure 5 shows a reconstruction of a car model described by a triangular mesh. The mesh contains 1114 elements. The regions based on distance and angle tolerance are shown in figure 5(b). The simplified mesh (5(c)) consist of 189 quadrilateral conformal mesh elements. The final reconstructed surface based on the mesh is shown in figure 5(d).

Figure 6 shows a smooth surface build from initial triangular data. The data (figure 6(a)) is the triangulation of a human face (the Nefertiti statue). The original model consists of 1747 triangles. The subdivision into regions based on distance and angle tolerances is shown in figure 6(b). The regions are highly irregular due to the model complexity. The model was analyzed with two different mesh element sizes shown in figures 6(c) and 6(e), resulting in meshes with 286 and 1146 elements respectively. The surfaces based on two meshed are shown on figures 6(d) and 6(f). The difference in the surface detail level is a result from the difference in the compression ratio.

Figures 7 demonstrate the complete surface reconstruction process of a non-manifold model of the Volkswagen Beetle car, starting from a quadrilateral surface mesh. First, the mesh (7(a)) is divided into clusters of restricted curvature (7(b)). Then a quadrilateral boundary conforming mesh of each cluster region is constructed (7(c)). The number of mesh elements is reduced from 308 element to 146. The major reductions are on the front hood and the trunk of the car, where the original surface has many small mesh elements describing small details. Note that the new mesh elements are more uniform in size which is an important property for any finite element analysis or other computations. The topology of the original mesh is fully preserved, including the relatively small (compared to mesh element size) light holes, and the non-manifold structure at the trunk light hood. The surface constructed over the mesh at the last step of the algorithm is smooth over all the car structure.

The running times for the examples above are comparable with those in the works of Hoppe and Eck ([5], [7]) for similar model sizes.

9 Conclusions

This work presents a new method for reconstruction of a smooth surface over an original mesh data. During the reconstruction, a simplification of the original object is achieved. Since the surface is subdivided into restricted curvature regions before the construction of the simplified mesh, the faces of the mesh have restricted curvature as well. As a result, the energy minimization method is both stable and efficient.

The present approach provides error bounds on the approximation that can be controlled by the simplification parameters. The use of finite element techniques allows the algorithm to work with no restriction on complex mesh structures containing N-vertex T-nodes and on non-manifold topologies as demonstrated in figure 7. Our experiments on a variety of data sets show that the algorithm can work on complex geometries including objects with large curvature changes. This approach allows to extend face simplification to free-form surfaces, and is not limited to only linear faces like other ones. The surface reconstruction method used does not require any subdivision of the simplified mesh, hence the introduced data reduction is preserved.

Further research is required to reduce and regularize the randomness in the cluster regions construction resulting from the random choice of seed and accreted boundary facets. Other topics of interest include examining and experimenting with different criteria for faces clustering and introducing adaptive refinement tools.

References

[1] M. Bercovier and O. Volpin, Hierarchical Bézier Surface Over Irregular Mesh, Technical Report, Leibnitz, Jerusalem, 1996. Submitted to Computer Graphic Forum Journal.

[2] T. Blacker and M. Stephenson, Paving: A new approach to automated quadrilateral mesh generation, *International Journal for Numerical Methods in Engineering, 32*, 1991, pp. 811-847.

[3] P. G. Ciarlet, *Numerical Analysis of the Finite Element Method*, Les Presses De L'Université De Montreal, 1976,

[4] M. Eck and H. Hoppe, Automatic Reconstruction of B-Spline Surface of Arbitrary Topological Type. *SIGGRAPH '96 Proceedings*, 1996.

[5] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery and W. Stuetzle, Multiresolution Analysis of Arbitrary Meshes, *Computer Graphics (Annual Conference Series, 1995. SIGGRAPH '95)*, Aug, 1995, pp. 173-182.

[6] T. S. Gieng, B. Hamann, K. I. Joy, G. L. Schussman and I. J. Trotts, Smooth Hierarchical Surface Triangulations, *Proceeding Visualization 96, IEEE Computer Society*, 1997.

[7] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle, Mesh Optimization, *Computer Graphics Proceeding, Annual Conference Series 93*, 1993, pp. 19-26.

[8] A. D. Kalvin and R. H. Taylor, Superfaces: Polygonal Mesh Simplification with Bounder Error, *IEEE Computer Graphics and Applications 16,3*, 1996, pp. 64-77.

[9] R. Klein, G. Liebich and W. Strasser Mesh Reduction with Error Control *Proceeding Visualization 96, IEEE Computer Society*, 1996, pp. 311-318.

[10] T. Matskewich, O. Volpin and M. Bercovier, "Discrete" G^1 Assembly of Patches Over Irregular Meshes, *Curves and Surfaces in Geometric Design*, Academic Press Limited, 1997.

[11] L. Kobbelt, Interpolatory Subdivision on Open Quadrilateral Nets with Arbitrary Topology, To appear in Computer Graphics Forum.

[12] J. Peters, C^1 continuity surface over irregular mesh, *CAD, 5*, 1995, pp. 115-141.

[13] F-J. M. Schmitt, B. A. Barsky and W-H. Du, An Adaptive Subdivision Method for Surface-Fitting from Sampled Data, *Computer Graphics 20,4*, 1986, pp. 179-188.

[14] A. Sheffer, T. Blacker and M. Bercovier, Virtual Topology Operators for Meshing, *Proceeding: 6th International Meshing Roundtable*, Utah, USA, pp.49-66, October 1997. To appear in International Journal of Computational Geometry and Applications.

[15] A. Sheffer, T. Blacker and M. Bercovier, Clustering: Automated Detail Suppression using Virtual Topology, *AMD-Vol. 220, Trends in Unstructured Mesh Generation*, ACME, 1997, pp. 57-64.

[16] G. Turk, Re-Tiling Polygonal Surfaces, *Computer Graphics, 26,2*, July, 1992, pp. 55-64.

[17] O.Volpin, M. Bercovier and T. Matskewich, A Comparison of Invariant Energies for Free Form Surface Construction, *Proceeding: Computer Aided Geometric Design: New Trends And Applications*, Crete, Greece, June 1997. Submitted to International Journal of Shape Modeling.

[18] O. C. Zienkiewicz, The Finite Element Method in Engineering Science, McGrawHill Publishing, London, 1971.

[19] ACIS 3.0 Geometric Modeler, Programmers Reference. Spatial Technology, 1997.

Acknowledgments

We would like to thanks Dr. L. Kobbelt for providing us the Volkswagen mesh example from [11].

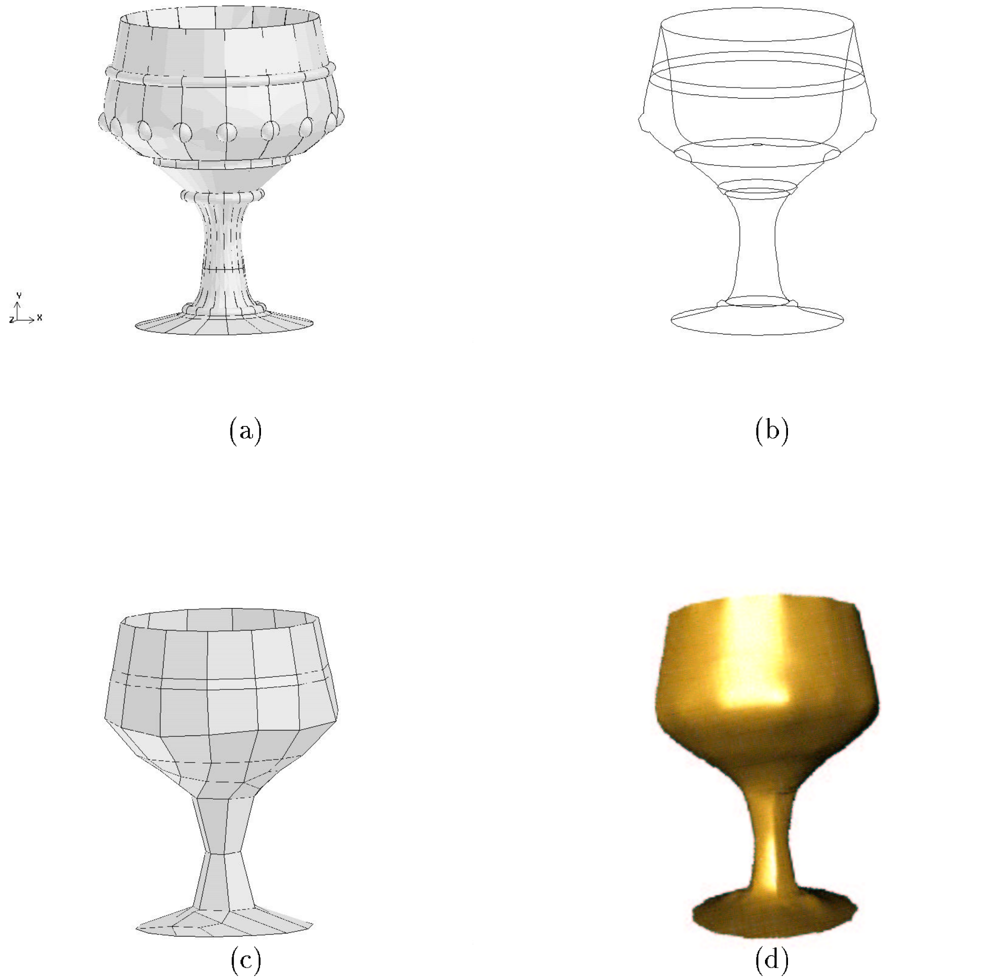

Figure 4: Mesh simplification of a free-form (NURB) glass. (a) original (free-form) mesh; (b) virtual faces resulting from clustering; (here we used very big angle tolerance; (c) the simplified quadrilateral mesh of the glass; (d) the reconstructed smooth polynomial surface.

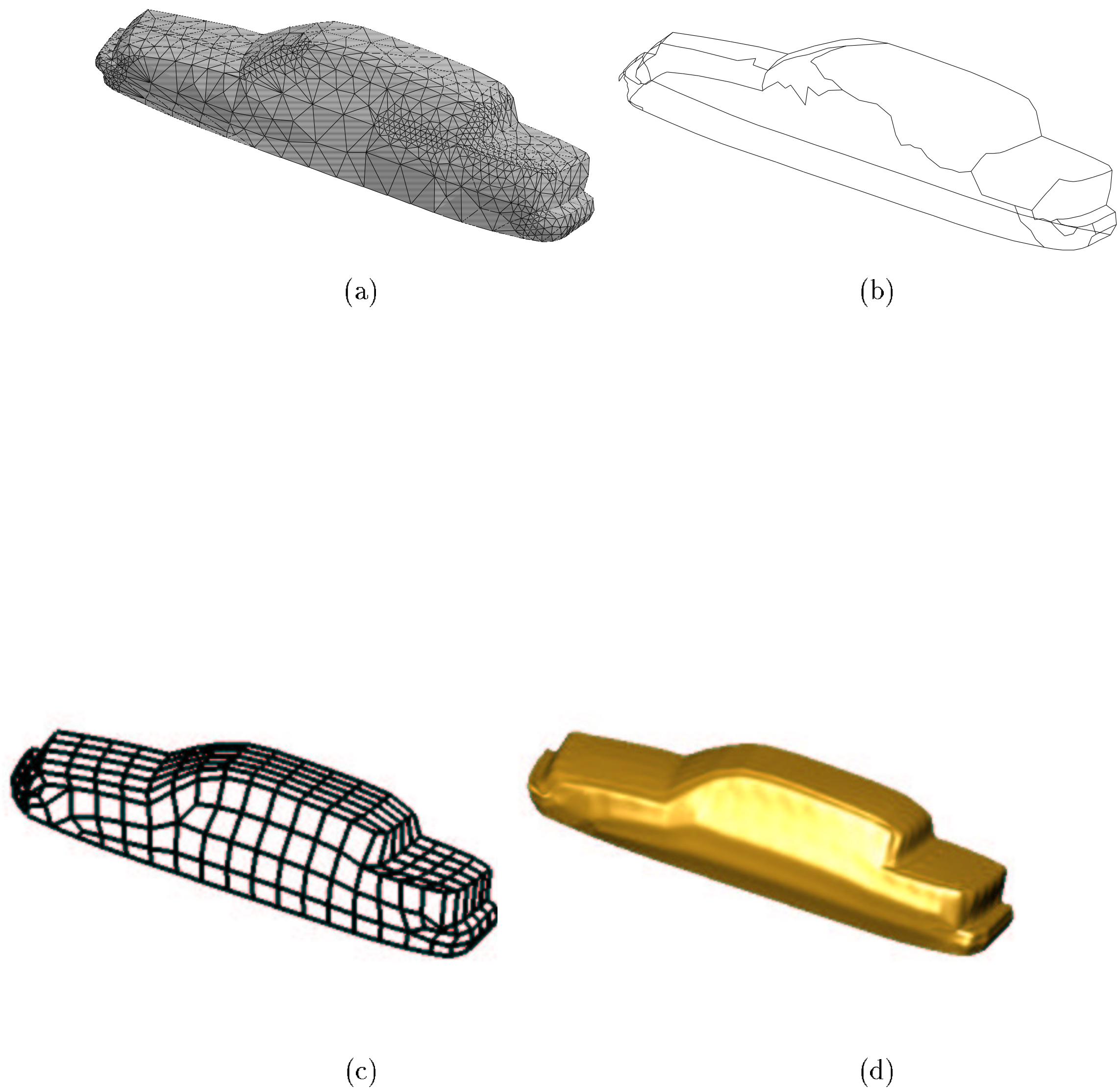

Figure 5: Mesh simplification of an arbitrary topology car. (a) original mesh; (b) virtual faces resulting from clustering; (c) the simplified quadrilateral mesh; (d) the reconstructed smooth polynomial surface.

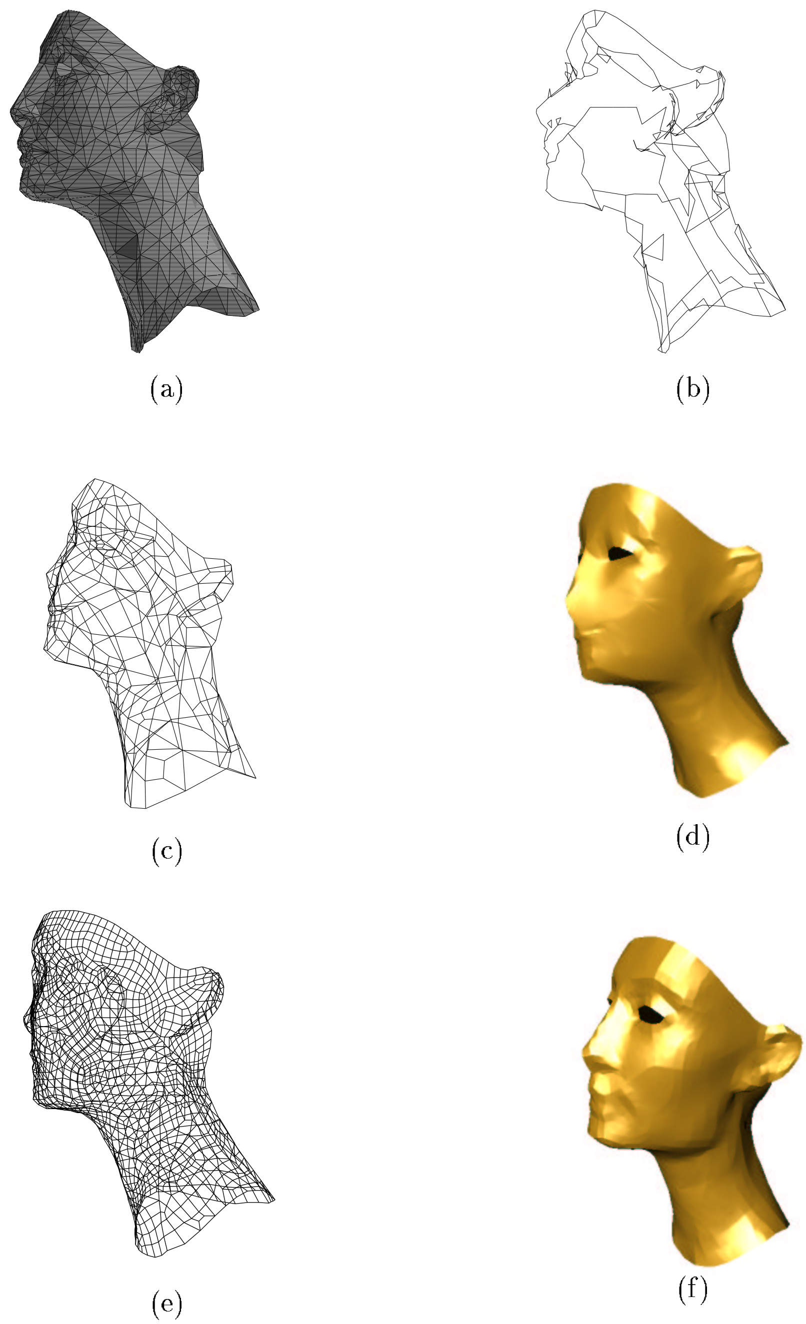

Figure 6: Mesh simplification of a human face (Nefertiti) (a) original mesh; (b) virtual faces resulting from clustering; (c) simplified mesh with big element size; (d) the smooth polynomial surface over mesh (c); (e) simplified mesh with small element size; (f) The smooth polynomial surface over mesh (e).

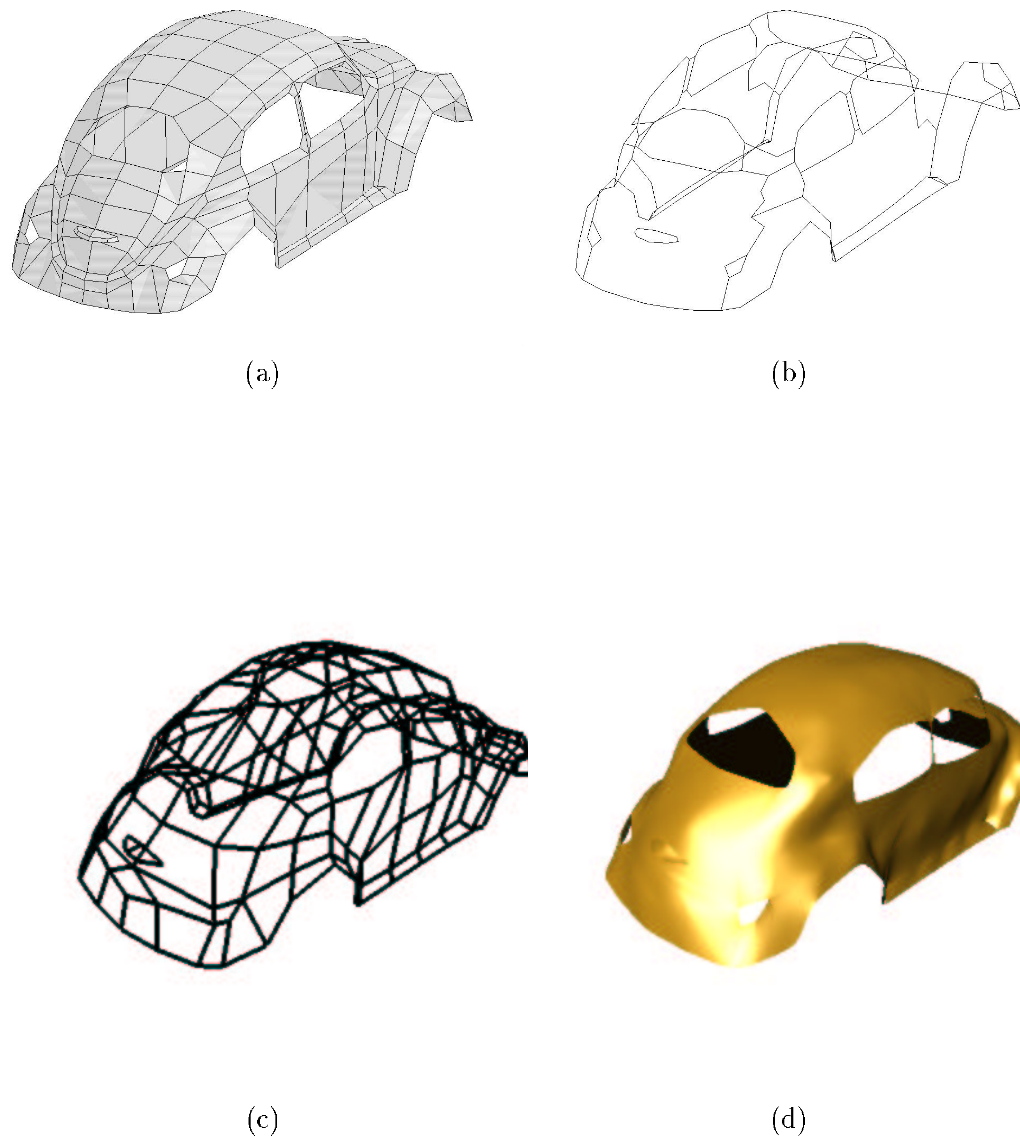

Figure 7: The mesh simplification of the VW mesh; (a) original mesh; (b) virtual faces resulting from clustering; (c) the quadrilateral mesh of the car; (d) the reconstructed smooth surface.