A Novel Skeleton Extraction Method by Iterative Edge Collapses on the Basis of the Quadric Error Metrics

Hung-Kuang Chen
Electronic Engineering Department,
National Chin-Yi University of Technology,
hank@ncut.edu.tw

Sih-Wei Chen
Electronic Engineering Department,
National Chin-Yi University of Technology,
szwei.chen0318@gmail.com

Abstract

Recently, the skeleton extraction from 3D models has been proven to be a very useful technique applied to various applications such as medical image analysis, computer animation, object recognition, and feature analysis, etc., and was intensively studied for over two decades. Numerous techniques were proposed to capture the main features of the input object and to generate a curve skeleton of the object so it can be used for the aforementioned applications. Previous works have suggested three types of approaches to do skeleton extraction, i.e., voxelization followed by thinning, Reeb graph conversion, and mesh contraction. In this paper, we adopt the framework of the mesh contraction approach suggested by Oscar K.C. Au et. el. Different from their work, we suggests applying the mesh contractions using the quadric error metrics and constrains the attraction by the sum of the edge lengths of the stars of a vertex. According to our experimental results, our new approach is successful and is able to generate a set of precise skeletal approximations from the input object.

Keywords：skeleton extraction, mesh contraction, quadric error metrics

1. Introduction

3D mesh skeleton extraction is a process to extract one dimensional feature, or the skeletons, from an input 3D mesh, which is commonly used to describe the topological structure of the given 3D mesh. This issue has been applied to various kinds of applications such as the 3D morphing/deformation [8], computer animation [1], 3D model repair [3], computer vision, and medical image analysis, etc., and was intensively studied for several decades.

Tons of techniques have been proposed previously. Three typical types of algorithm are developed recently, i.e., the approaches using the Reeb graph, the approaches through voxelization and thinning, and the approaches that make use of mesh contraction. In this paper, we have developed a new contraction-based technique on the basis of the method proposed in [6], which finds a set of skeletons from the input mesh by contracting and shrinking the mesh inward along the surface normal. Different from [6], we have adopted the edge collapse operation and the quadric error metrics (QEM) respectively developed by [2] and [5].

2. Algorithm Overview

An illustration of our algorithm is given by the flow diagram shown in the Figure 1.

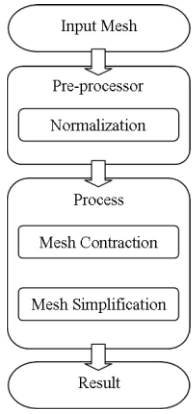


Figure 1: The flow diagram of our algorithm.

Prior to the skeletonization, the input mesh is normalized to a 2x2x2 volume where each coordinate is in the range of [-1, 1], and the scale of each axis is store for later use of recovery.

The process of skeletonization is an iterative process of contraction and shrinking by mesh simplification. The contraction is held by moving the vertices of the mesh along the opposite direction of their surface normal, which gradually approaches a zero volume skeletal structure. Afterwards, the contracted mesh will be simplified iteratively to a set of skeletons guided by the quadric error metrics.

Furthermore, we have assumed that the input mesh to be triangular and denoted as **M**(**V**, **T**) where **M** is the mesh itself, **V** is the set of vertices of the mesh, and **T** is the set of triangular faces (or the triangles).

3. Mesh Contraction

Our contraction method is based on the approach proposed in [6], its purpose is to iteratively contract the mesh inward by moving the vertices along the opposite direction of their surface normal and finalized with a nearly zero volume skeletal-like structure which resembles the medial axis of the input mesh.

To facilitate the contraction, we have to compute with the Laplacian operator **L**, which is essentially an $n \times n$ matrix (n=|V|) where each elements are computed as follows.

$$\mathbf{L}_{ij} = \begin{cases} \omega_{ij} = \cot\alpha_{ij} + \cot\beta_{ij}, & \text{if} \quad e_{ij} \in E \\ \sum_{e_{ij} \in E}^{k} -\omega_{ik}, & \text{if} \quad i = j; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Note that α_{ij} and β_{ij} respectively represent the angles of the opposite corners of the left and right adjacent triangles of the edge (i, j) as depicted by Figure 2.

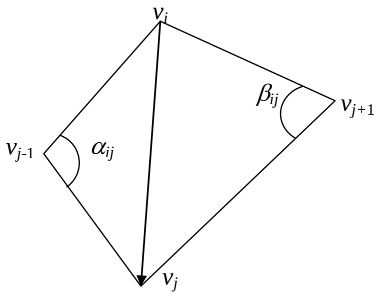


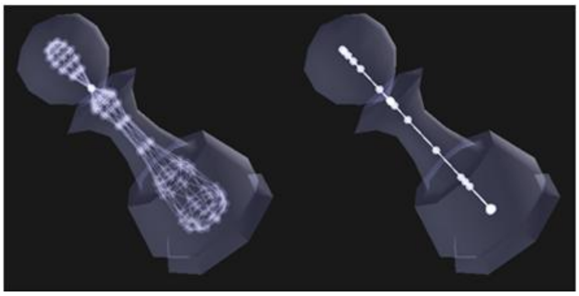
Figure 2: The two opposite angles of an edge (i, j).

After the operator L is computed, we will find the contracted mesh by

$$\begin{bmatrix} \mathbf{W}_L^t \mathbf{L} \\ \mathbf{W}_H^t \end{bmatrix} \mathbf{V}^{t+1} = \begin{bmatrix} 0 \\ \mathbf{W}_H^t \mathbf{V} \end{bmatrix} \tag{2}$$

Where t represents the number of iterations, and $\mathbf{W}_L$ controls the extent of inward forces and $\mathbf{W}_H$ controls the forces toward the original surfaces

We may fine tune the two matrices $\mathbf{W}_L$ and $\mathbf{W}_H$ to achieve better results. Each contraction moves the vertices further inward and thus reduces the volume of the object. An example (the Chess model) is shown in Figure 3, where Figure 3(a)-(b) respectively shows the results after one and four times of contractions.



(a) **(b)**

Figure 2: The Chess mesh (a) after one time of contraction; (b) after four times of contraction.

Since the contraction is an iterative process, the mesh is altered after each contraction. In this paper, we will set W_L and W_H to $\sqrt{A}$ and 0.1, where A is the average of the area of the faces. To solve for the solution of Equation (2), we made used of the Cholesky function of GSL(GNU Scientific Library).

After the new vertex positions are find, WL and WH are updated as follows so that they can be used for the next contraction step.

$$\mathbf{W}_L^{t+1} = s_L \mathbf{W}_L^t \tag{3}$$

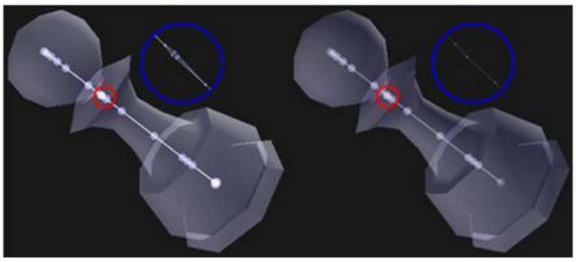
$$\mathbf{W}_{H,i}^{t+1} = \mathbf{W}_{H,i}^0 \sqrt{E_i^0 / E_i^t} \tag{4}$$

Note that s_L is a user defined constant. We have set it to 2.0 in this paper. To find new $\mathbf{W}_H$, we made used of the sum of the adjacent edges of the stars of the vertex. In Equation (4), E_i^0 and E_i^t respectively represents the contraction forces corresponding to the initial and the current forces.

Finally, with the substitution of the new vertex coordinates into Equation 1, we can get a new Laplacian operator **L** for further contraction. After several iterations of such steps, we may stop at a satisfactory result.

4. Mesh Simplification

After the iterative contraction, the shrinkage moved the vertices of the input mesh near to the medial axis of the 3D model. However, such resulting mesh is what we want. The connectivity of the mesh has not been changed; it is by far still a 2D mesh with nearly zero volume. To get a set of skeleton from it, we need to remove redundant edges and vertices other than the joins and bones by simplifying the mesh as shown in Figure 4.



(a) **(b)**

Figure 4: The contracted mesh of the Chess model: (a) before the mesh simplification; (b) after the mesh simplification stage.

In this stage, the mesh is simplified by repeatedly applying edge collapse operations to the mesh based on the costs estimated by the quadric error metrics. The process continues until only a one dimensional structure is left.

To guide the simplification, the sum of the *shape cost* and *sampling cost* is used. In each step, an edge with lowest such cost is selected and removed from the mesh until all the faces are eliminated and the mesh is reduced to a set of 1D structures. The calculations of the two costs are given as follows.

4.1 The Shape Cost

The shape cost is estimated with commonly used quadric error metrics (QEM) proposed by [5]. The calculation starts by finding the plane equation of each triangular face by substituting the vertex coordinates into Equation (5) as follows.

$$ax + by + cz + d = 0 \tag{5}$$

After the coefficients of the plane equation were solved, a matrix $\mathbf{Q}^f$ called the *face quadric* is computed as follows.

$$\mathbf{Q}^f = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix} \tag{6}$$

By summing up the face quadrics of the faces in the ring neighborhood of a given vertex, the vertex quadric $\mathbf{Q}_i$ of such vertex is computed.

$$\mathbf{Q}_i^v = \sum_{\forall f_j \in R(v_i)} \mathbf{Q}_j^f \tag{7}$$

The shape cost of collapsing edge $e_{ij}=(v_i, v_j)$ is calculated as follows.

$$C_{shape}(e_{ij}) = v_j^T \mathbf{Q}_i^v v_j + v_i^T \mathbf{Q}_j^v v_i \tag{8}$$

4.2 The Sampling Cost

To avoid over-simplified, the sampling cost is introduced to provide proper control over the simplification and to achieve a better result. In this paper, the sampling cost of an edge e_{ij} is given as follows.

$$C_{sampling}(e_{ij}) = \|v_i - v_j\| \sum_{e_{ik} \in E} \|v_i - v_k\| \tag{9}$$

4.3 Total Cost

The total cost used to guide the simplification is a weighted sum of the shape and the sampling costs and is computed as follows.

$$C_{total}(e_{ij}) = w_a C_{shape}(e_{ij}) + w_b C_{sampling}(e_{ij}) \tag{10}$$

In Equation 10, w_a and w_b respectively represents the weights of the shape and the sampling costs. In this paper, we have set $w_a = 1.0$ and $w_b = 0.1$.

5. Experimental Results

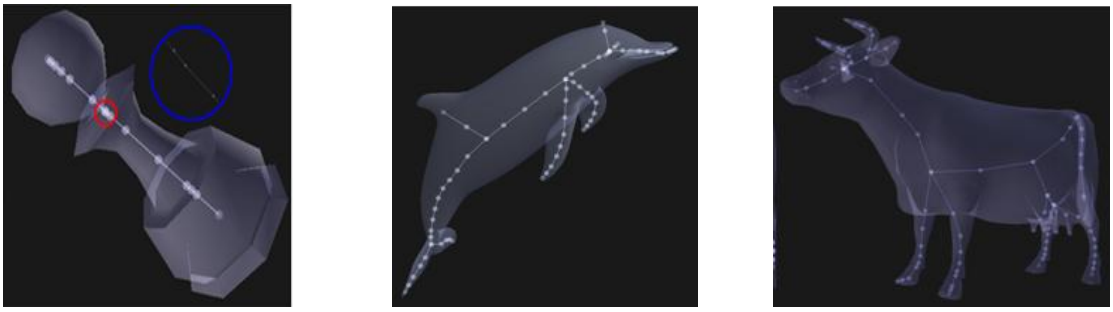
The algorithm we proposed in this paper has gone through a series of experiments to verify its fitness for skeleton extraction. The results showed that the proposed method is resistant to the affection of insignificant tiny features on the surfaces. As we have depicted in the Figure 5, the results of our method in the extraction of the skeletons of the Chess the Dolphin, and the Cow models are not affected by their insignificant features.

6. Concluding Remarks and Future Works

Through the experimental results, we find our method to be successful in the extraction of skeletal information from manifold surfaces. For non-manifold surfaces or the surfaces with boundaries, our method cannot work.

Furthermore, our method is not capable of dealing large meshes. The major causes resides in the massive amount of calculations resulted from the large matrix operations.

In future works, we will try two directions for possible improvements. The first is to speedup the matrix calculations with the support of massive



(a) (b) (c)

Figure 5: The extracted skeletons of (a) the Chess, (b) Dolphin, and (c) Cow models.

parallelism provided by concurrent GPU technology. The second way to deal with such issue might be achieved by segmenting the meshes so that the size of the matrix can be reduced considerably.

7. References

[1] A. Vasilakis, I. Fudos, “Skeleton-Based Rigid Skinning for Character Animation, “ In Proc. of GRAPP, pages 302–308, 2009

[2] H.K. Chen, C.S. Fahn, J- J.P. Tsai, and M.B. Lin, “A Novel Cache-based Approach to Large Polygonal Mesh Simplification, ” Journal of. Information Science and Engineering, Vol. 22, No. 4, pp. 843-861, 2006.

[3] H. Sundar, D. Silver, N. Gagvani, S. Dickinson, “Skeleton Based Shape Matching and Retrieval, “In Proc. Shape Modeling International (SMI), 2003.

[4] M. Desbrun, M. Meyer, P. Schröder, A. H. Barr, “Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow, “In Proc. of ACM SIGGRAPH, 317–324, 1999

[5] M. Garland and P.S. Heckbert. “Surface simplification using quadric error metrics, ” In Proc. of SIGGRAPH 1997, pages 209–216, 1997.

[6] Oscar Kin-Chung Au, Chiew-Lan Tai, Hung-Kuo Chu, Daniel Cohen-Or, and Tong-Yee Lee, “Skeleton Extraction by Mesh Contraction, ” In Proc. of SIGGRAPH 2008, ACM Transactions on Graphics, Vol. 27, No. 3, pp. 44:1-44:10 ,August 2008,

[7] Patanè G, Spagnuolo M, Falcidieno B. “A minimal contouring approach to the computation of the Reeb graph, ” IEEE Trans Vis Comput Graph. ;15(4): 583-95., 2009 Jul-Aug

[8] R. L. Blanding, G. M. Turkiyyah, D. W. Storti, M. A. Ganter, “Skeleton-based three-dimensional geometric morphing, “ In Proc. of Computational Geometry, 129-148, 2000.

[9] V. Pascucci, G. Scorzelli, P.-T. Bremer, and A. Mascarenhas, “Robust On-line Computation of Reeb Graphs: Simplicity and Speed, ” Proceedings of ACM SIGGRAPH 2007, pages 58.1-58.9, 2007.

[10] Y.-S. Wang, T.-Y. Lee, “Curve-Skeleton Extraction Using Iterative Least Squares Optimization, ” In Proc. of IEEE Transactions on Visualization and Computer Graphics , 2008.