Mesh Simplification Based on Shading Characteristic

ZHANG Hui
Dept. Of BME
Southeast University
P.R.China 210096, NJ, JS
zh@seu.edu.cn

SHU Huazhong
Dept. Of BME
Southeast University
P.R.China 210096, NJ, JS
shu.list@seu.edu.cn

LUO Limin
Dept. Of BME
Southeast University
P.R.China 210096, NJ, JS
luo.list@seu.edu.cn

ABSTRACT

Mesh simplification is a key research area in scientific visualization and virtual reality. The paper presents a new method of polyhedral model simplification based on vision characteristic and edge collapse. This method determines model's vision characteristic by computing the maximum product of angle between normal of vertex and adjacent triangles with area of these triangles. It can achieve a high decimation rate, while keeping the details of vision. This algorithm can also be extended to triangle decimation criterion easily.

Keywords

Mesh Simplification Vision Characteristic Edge Collapse Triangle Decimation

1. INTRODUCTION

Object's geometry models are always presented by polyhedral meshes, in the application of interactive computer graphics such as computer animation, scientific visualization and virtual reality. The most common and widely used is triangle model, which can directly take advantage of graphics hardware structure and be accelerated. But on many occasions, to achieve the requirement of high fidelity, high resolution model is generally comprised of millions of triangle primitives. The total number used to present object often largely exceeds the capacity of real-time rendering of graphics hardware, which is a bottleneck of virtual simulation. One nature way to solve it is to simplify the mesh models, replacing the original object with respectively few faces while trying to keep characteristic of vision effect.

The mechanism of mesh simplification is divided into four groups: Sampling, Adaptive Subdivision, Decimation and Vertex Merging [Sch92a] [Ros93a][Hop96a][Gar97a].

(1) Sampling algorithms sample the geometry of the initial models, either with points upon the model's surface or voxels superimposed on the model in a 3D grid. They may have trouble achieving high fidelity since high frequency features are inherently difficult to sample accurately.

(2) Adaptive subdivision algorithms find a simple base mesh that can be recursively

***WSCG SHORT PAPERS** proceedings*
WSCG'2003, February 3-7, 2003, Plzen, Czech Republic.

subdivided to more and more closely approximate the initial model.

(3) Decimation techniques interactively remove vertices or faces from the mesh, re-triangulating the resulting hole after each step. These algorithms are relatively simple to code and very fast.

(4) Vertex merging schemes operate by collapsing two or more vertices of a triangulated model into a single vertex, which can in turn be merged with other vertices. Vertex merging is a fairly simple and easy-to-code mechanism, but algorithms use techniques of varying sophistication to determine which vertices to merge in what order. Most view-dependent algorithms are based on vertex merging.

The above presented algorithms, almost no one utilize the information of normal of surface and vertex. Based on the explanation of human vision system, man's eyes are more sensitive with the change of surface normal than the displacement of vertex position. Measurement of result model should be mainly based on vision effect, not simply on some error function. For the reason that most light model need generate normal of surface itself, the normal difference of vertex or surfaces is more sensitive to human vision. So Gorland and Heckbert suggest that, the way how to evaluate the result is not only based on the error of model, but also based on result image [Gar97a]. Jiang brings up a method using normal information. Every time Jiang's algorithm selects an edge, whose angle between two vertexes, is less than a threshold, to collapse [Jia99a]. Considering that it doesn't predict the result, it may cause some unwanted detail result such as face fold. And all, it doesn't take into account the influence of triangle area. Schroeder presents an algorithm based on the maximum angle between the averaged normal and the surface normal of the surrounding triangles [Xia96a]. He uses it as a measurement to delete the least important ones, and then try to re-triangulate the hole generated. That algorithm's result is influenced by the re-triangulation, so an appropriate re-triangulation method must be taken to keep the fidelity. Its result vertex set is also inside the original set.

Considering the importance of normal in mesh simplification, we present a new algorithm based on normal computing. It uses edge collapse, according to the product of the angle between vertex and adjacent triangle with the triangle area, to form a new vertex, while keeping the vision effect of destination. That is, trying to delete the less important vertex and keep the ones can mostly represent the model characteristic.

2. DEFINITION

Def 1: Mesh M is comprised of a group of triangles, adjacent edges and vertex, presented by vertex set V, edge set E and triangle set T.

Def 2: Adjacent vertex set NV (V_i) of vertex V_i are the vertex connected to vertex V_i by edge.

Def 3: Adjacent triangle set NT (V_i) of vertex V_i is the set of triangles whose vertex has V_i inside.

Def 4: Adjacent triangle set NT (T_i) of triangle T_i is comprised of combine adjacent triangle sets of three vertex of triangle.

Def 5: Characteristic value C (V_i) of vertex V_i is the number of vertex in V_i's adjacent vertex set, which not appear in couple.

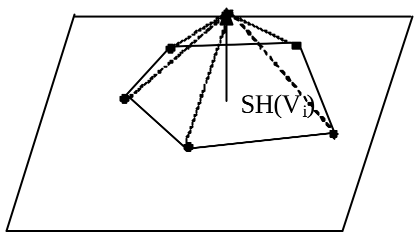
Def 6: Area of super plane SA (V_i) of vertex V_i is the sum area of triangulated NV (V_i). Ordering that area of triangle is A_k, then $SA(A_i) = \sum A_k$.

Def 7: Volume of super tetrahedron SV (V_i) of vertex V_i is the sum volume of V_i and NV (V_i). Ordering that volume of tetrahedron is Vol_k, so $SV(V_i) = \sum Vol_k$.

Def 8: Super height $SH(V_i) = \frac{SV(V_i)}{SA(V_i)}$, this value represents the curvature of that vertex, as shown in Fig 1.

Def 9: Vision characteristic value CA (V_i) of vertex V_i is the maximum product of triangle area and the angle between the normal of vertex and NT (T_i), similarly with the maximum angle defined by Schroeder in [Sch94a], which not taken into consider the factor of surrounding triangle's area. This factor may merge the tiny triangle in processing. As shown in Fig 2.

$$CA(V_i) = \max\{q(n_{vi}, n_k) \cdot A_k\} = \max\left\{\arccos\left(\frac{n_{vi} \cdot n_k}{|n_{vi}| \cdot |n_k|}\right) \cdot A_k\right\}$$



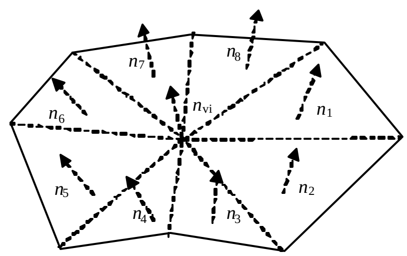
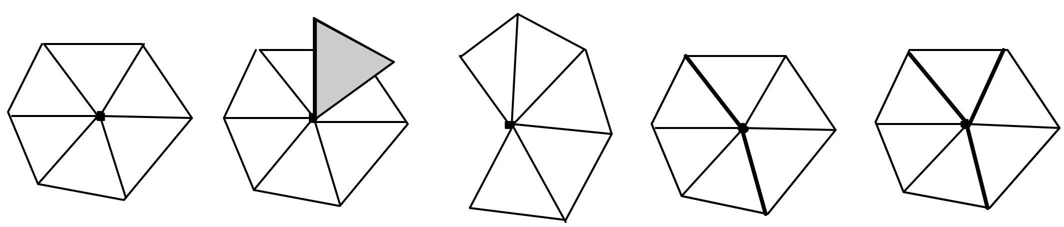


Figure 1 Super height of vertex V i **Figure 2 Vision characteristic of vertex V i**



(a) Simple Vertex (b) Complex Vertex (c) Edge Vertex (d) Inner Edge Vertex (e) Angle Vertex

Figure 3 Classification of vertex

Considering the different position of vertex related to mesh, it can be classified into simple vertex, complex vertex, edge vertex, inner-edge vertex and angle vertex, as shown in Fig. 3. Undoubtedly, simple vertex is the main style to be merged, while complex vertex and edge vertex is characteristic vertex, should be maintained. Inner-edge vertex and angle vertex is also the candidate be handled. These vertex, can be distinguished by characteristic value C(V_i), and their corresponding value is simple vertex 0, complex vertex greater than 2, edge vertex 2, inner-edge vertex 0, angle vertex 0. The difference of inner-edge vertex and angle vertex can be recognized by SH (V_i) or CA (V_i). The mainly processed vertex is carried on three classes: simple, inner-edge and angle vertex.

3. PRINCIPLE AND ALGORITHM

Our algorithm uses the mechanism of vertex merging, and it is described as below.

Edge collapse algorithm based on vision characteristic:

(1) Compute the normal $Nor(T_i)$ of each triangle;

(2) Use some methods to calculate the influence value of each vertex. The two methods we used are to calculate the vision characteristic CA (V_i) or super height $SH(V_i)$, which represent the influence of curvature to result image;

(3) Sort these vertexes, according to its influence in vision characteristic;

(4) Pick out the first vertex in the queue, as the first candidate point to collapse. Select another one V_j from adjacent vertex set NV (V_i). The policy is to select one which can generate the least characteristic vision value of changed vertex;

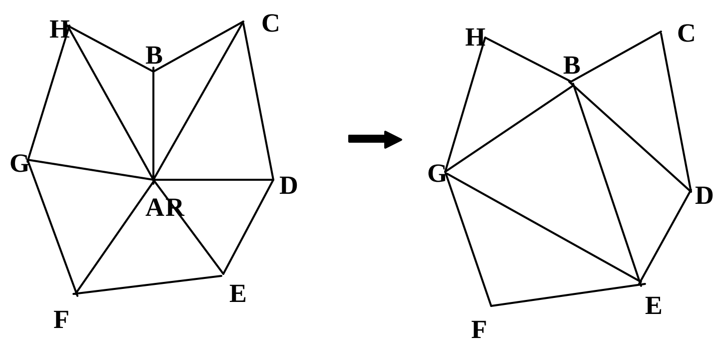
(5) Interpolate the related properties of V_i and V_j, such as vertex position, normal, texture. Then use the new vertex V_k to replace V_i and V_j, delete the regressive triangle. Calculate the value of new point V_k and its adjacent vertex set $NV(V_k) = NV(V_i) \cup NV(V_j)$, and arrange their new position in the vertex queue;

(6) Repeat step (4) and (5), until reach the required triangle number.

When the candidate edge E (V_i, V_j) is collapsed, it can delete 2 adjacent triangles, 3 adjacent edges, 1 vertex. In the above algorithm, the key step is how to select the two vertexes to be merged, that is, step (2) and (4). Step (2) uses some policies to generate a vertex queue, and then select the least important one to process. Step (4) is to select the adjacent edge to collapse after the first vertex is determined which causes the least change in vision. Some of details are discussed as below.

- Computation of the super height of vertex V_i, that is, distance from it to its super plane comprised of its adjacent vertex set.

It is common thought that the protruding points in space play an important role in vision, which should be maintained. The smooth ones are less important and can be removed. Obviously, the distance from it to its super plane can be taken as a measurement. As shown in Fig 4(a), as to the adjacent vertex union of vertex A, we at first triangulate that region, as Fig 4(b). For the reason that some triangles are generated after triangulation, we can use sum of them as area of super plane, and sum value of tetrahedrons comprised by triangle and vertex A as volume of super tetrahedron. So the height can be approximately be get by $SV(A)/SA(A)$.



(a)Vertex A and its adjacent vertex (b) Triangulation after deleting A

Figure 4 Computation of super height of vertex A

- Characteristic value of vertex to its adjacent triangle

We give every vertex a certain normal, which can be calculated by average from adjacent triangle's normal. As to the initial vertex V_i, algorithm need select a candidate vertex in the relative adjacent vertex union NV (V_i) as the direction to collapse. Our algorithm selects the edge which makes the least change. Regarding a candidate edge E (V_i, V_j), a new vertex V_k can be generated by interpolation. The position and normal is linear interpolated according to the vision characteristic value of V_i and V_j. Then compute the new CA(V_k), while changing the topology of V_i and V_j and adjacent vertex, as shown in Fig 5. Considering each edge of V_i, the one who has least vision characteristic is used.

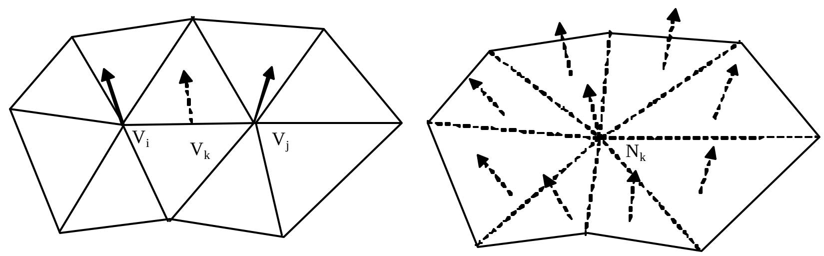


Figure 5 Process of edge collapse from E (V_i, V_j) to vertex V_k

The reason why we re-calculating the new CA(V_k) of new vertex generated is to avoid the folder-over problem of surface mentioned in [11]. We know that, expanding or contracting a vertex pair at random order could produce erroneous approximations. In order to decrease the influence of this problem, additional procedures must be used to check it when decimation. This operation is usually expensive and influences the performance of scheme. Our algorithm, by previously calculating the change of triangle normal, can prevent this in some cases. , As shown in Fig 6, we will delete vertex V_1, the candidate collapse edges are V_1 V_2 and V_1 V_3. If we select V_1 V_3, it will causes triangle V_2 V_4 V_6 to folder over, it will greatly increase the vision characteristic value CA (V_2). In our algorithm, this case will be ignored by using V_1 V_2 to collapse, which will produce less valued vision characteristic value CA (V_3).

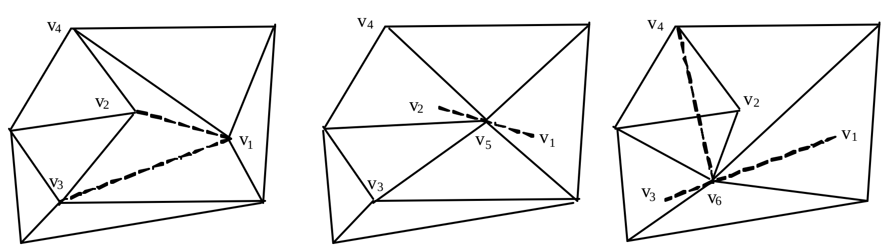


Figure 6 Procedure of folder-over triangle

- Extension of algorithm

Based on our algorithm, after small change, a new triangle decimation algorithm will be got. We can define triangle's vision characteristic value as the maximum product of angle between triangle normal and its adjacent ones with triangle area. Select triangle which has the least value to delete. Extended triangle decimation algorithm is described as below.

Triangle decimation algorithm based on vision characteristic:

(1) Compute every triangle's normal Nor(T_i) and vision characteristic CA(T_i);

(2) Sort these values from least to largest;

(3) Select the top triangle in the queue, and contract it to a single vertex. The position can be generated by interpolation according to vision characteristic value. Then change the adjacent vertex and triangle;

(4) Recalculate the affected triangles, and update their position in the queue;

(5) Repeat step (3) and (4), until the triangle number of decimated mesh is satisfied.

4. RESULT ANS ANALYSIS

We accomplished the presented algorithm in PC system, and experiment with some classic models. The used hardware environment is CPU PIII 550, RAM 256M. The selected models are bunny model (35947 vertexes, 69451 triangles) and horse model (48485 vertexes, 96966 triangles). The data structure used includes vertex list, adjacent edge list and triangle list. When edge collapse operator is processed, we only need update the responding vertex list and triangle list, which can accelerate the process speed. Now we try to analyze it from two ways: decimation effect and computation time.

- Decimation result

As shown in Fig 7 and 8, they are bunny model and horse model. From left to right are original model, decimation rate 90 %(surface rendering), 95 %(surface rendering), 95 %(mesh display). The vertex and triangle number of decimated model is listed in table 1. We can see that, when decimation rate is around 90%, it still contains the pretty good result; when 95%, it still can keep the major topology characteristic of initial model. As indicated in mesh model, in some smooth part of decimated mode, such as abdomen of bunny and horse model, a few large triangles is used to approximate it. In some coarse and protruding part, such as ear and foot of bunny, head of horse, large amount of small triangles is used to render. It shows that good balance is taken between details of model and area of triangles.

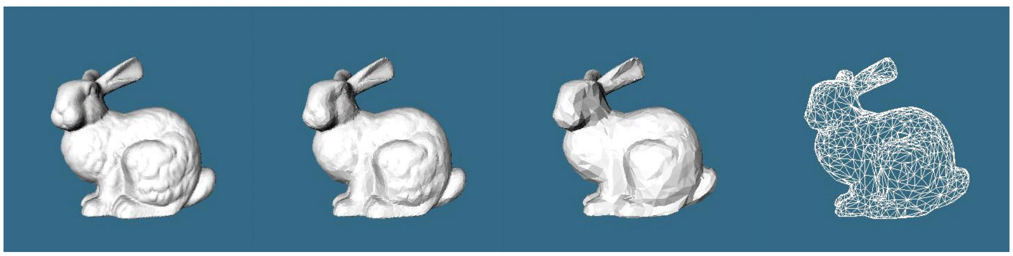


Figure 7 Decimated result of bunny model

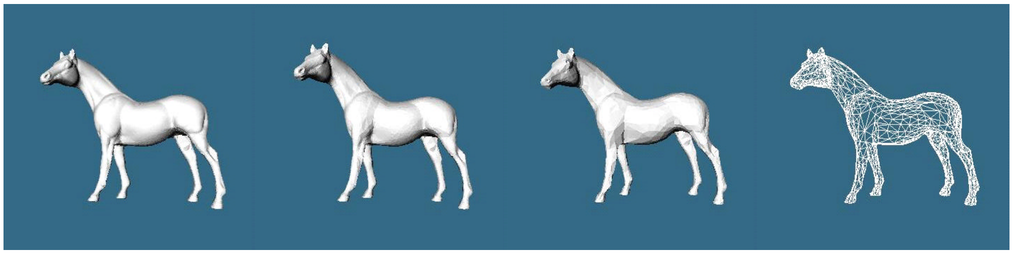


Figure 8 Decimated result of horse model

From left to right are: original model, decimated 90% model, decimated 95% model (surface rendering and mesh display)

	original	50%	80%	90%	95%
Bunny vertex	35947	17471	7053	3581	1844
Bunny triangle	69451	34725	13889	6945	3471
Horse vertex	48485	24243	9698	4850	2426
Horse triangle	96966	48482	19392	9696	4848

Table 1 Change of vertex and triangle number in decimation of bunny model and horse model

- Computation time

Large amount of float operator is needed for calculating of some normal info of vision characteristic, which make it a little slower that these algorithms which need not normal computing. A lot of operation is integrated to reduce time cost, such as two ways merging sort in step (3) and half search in step (5). Time needed in various decimation rate is shown in Fig 9. From it, we can see that, the time consume is linear, which means the algorithm computing time will not increase greatly as number of triangles increase.

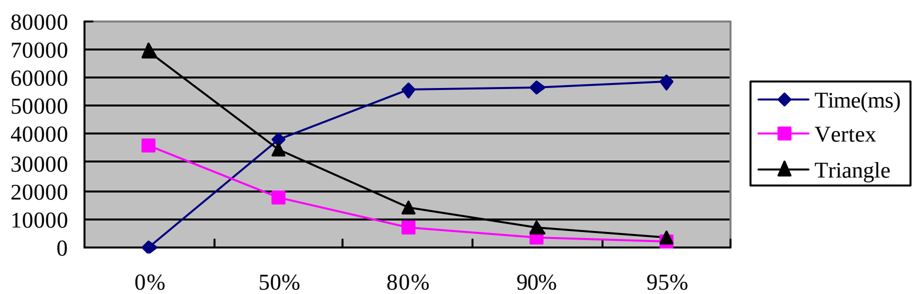


Figure 9 Time curve of mesh simplification algorithm

- Algorithm analysis

Algorithms which use normal in mesh controlling are presented in [Sch94a] [Nak95a] [Hin93a] Schroeder and Nakamae use the maximum angle of vertex normal and adjacent triangle normal as principle to decimate vertex. Our algorithm uses instead the maximum product value of angle with area of adjacent triangle, which can remove the effect brought by small faces. Other different ways of our algorithm with theirs is that our algorithm use edge collapse step instead of their re-triangulation and we use vision characteristic value to control the generated triangle's normal change, which can make more smooth result . And also this value is used in edge collapse step to select the candidate edge to contract. In Hinker and Kalvin's algorithm, plane with nearly parallel normal is merged[Kel96a]. And these algorithms need triangulate the hole generated when vertex is deleted.

5. CONCLUSION

The effect is obvious when vision characteristic value is introduced into edge collapse algorithm. It produces high quality decimation result, while easy to code. This algorithm can also automatically recognize and simplify the nearby co-planar triangle, while keeping the key vertex. This result can be seen from 95% decimated model, and our algorithm can be used to automatic generation of virtual reality and interactive visualization.

6. REFERENCE

[Gar97a] Garland, M., and Heckbert, P., Simplification Using Quadric Error Metrics, Computer Graphics, Vol31, pp.209-216, 1997

[Hin93a] Hinker P., et al. Geometric Optimization, IEEE Proceedings of Visualization, Los Alamites, pp.189-195, 1993

[Hop96a] Hoppe, H., Progressive Meshes, Computer Graphics, v.30 (SIGGRAPH 96)

[Jia99a] Jiang SP., Zhou MT., Dai Y., Mesh Simplification based on normals, Chinese Journal of Computers, Vol22(10), pp.1074-1079, 1999

[Kel96a] Kelvin A, et al, Super face: Polygonal Mesh Simplification with Bounded Error, IEEE Computer Graphics and Application, Vol3, pp.64-77, 1996

[Nak95a] Nakamae E, et al. Simplification of Polygonal Surface with Attributes, Proceedings of ICSC95, Hong Kong, pp. 132-139, 1995

[Ros93a] Rossignac, J., and Borrel, P. Multi-Resolution 3D Approximation for Rendering Complex Scenes, pp.455-465 in Geometric Modeling in Computer Graphics, Springer-Verlag(1993), Genova, Italy

[Sch92a] Schroeder, W., Zarge, J. and Lorenson, W. Decimation of Triangle Meshes, Computer Graphics, Vol26(2), pp.65-70, 1992

[Sch94a] Schroeder, et al. Managing the Complexity of Digital Terrain Models, Computer Graphics, Vol18(6), pp.775-783, 1994

[Xia96a] Xia J., Varshney A. Dynamic view-dependent simplification for polygonal models. In: Proceedings of IEEE Visualization'96, pp.327-334, 1996