Surface Mesh Simplification

Stefano Falangone N97/304

Computer Graphics Course
Master's Degree in Computer Science
Department of Electrical and Information Engineering
University of Naples, Federico II

March 2020

Introduction I

Definition

Surface Mesh Simplification is the process for reducing the polygons a Mesh is formed of based on some criteria.

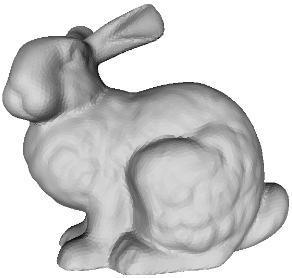


Figure: Bunny formed of 28576 Triangles

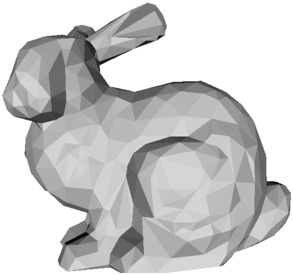


Figure: Bunny formed of 1500 Triangles (95% reduction)

Introduction II

Problem

Scanners keep generating meshes with larger numbers of faces.
Mesh Simplification Alghoritms aim to **reduce the number of faces** of a Mesh to speed up graphics computation.

Tradeoff

- Efficency: time needed to produce a solution
- Quality: measure of approximation found
- Generality: categories of meshes that can be processed

Common strategies adopted

Vertex Decimation

Generate a hole and fill it linking a vertex (Schroeder's approach)

Vertex clustering

Build a bounding box around the model, grid it into cells. All vertices of a cell collapse into one (Rossignac and Borrel)

Edge collapse

Bring one edge of the graph to collapse (Roosignac, Hoppe, Garland, Lindstrom and Turk)

Modeling a Mesh

$M = \langle V, E \rangle$ where V is a set of vertices, $E \subseteq VxV$

Topological entities

- v_i 0-simplex
- e_i 1-simplex
- t_i 2-simplex

Simplex operators

- $\lfloor s \rfloor$ denotes the $n-1$ faces of a n-simplex
- $\lceil s \rceil$ denotes the $n+1$ simplexes that include s

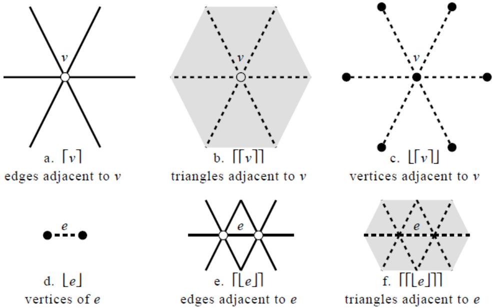


Figure: examples of $\lfloor s \rfloor$ and $\lceil s \rceil$

General idea of a iterative alghoritm

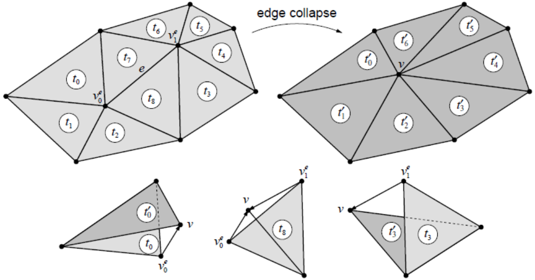
Preprocessing

For each $e_i = (v_1, v_2) \in E$ compute:

- position for a new vertex
- cost for the edge reduction

Iterate

- Collapse v_2 on v_1
- Transfer all $\lceil v_2 \rceil$ and $\lceil \lceil v_2 \rceil \rceil$ to v_1
- Recompute position and cost of $\lfloor \lceil v_1 \rceil \rfloor$



Mid-point solution

Mid-point cost and solution

Given $e_i = (\mathbf{v}_1, \mathbf{v}_2)$ let:

1. $f_c(e) = dist(\mathbf{v}_1, \mathbf{v}_2)$
2. $solution = (\frac{(\mathbf{v}_x^1 + \mathbf{v}_x^2)}{2}, \frac{(\mathbf{v}_y^1 + \mathbf{v}_y^2)}{2}, \frac{(\mathbf{v}_z^1 + \mathbf{v}_z^2)}{2})$

Pro

- Simple to implement
- Stable
- Computationally cheap

Con

- Does not preserve any geometric property
- too much locally based

Lindstrom-Turk strategy

Lindstrom-Turk approach opposed to mid-point solution. Compute first the position of new vertex, then the cost.
Position for vertex **v** after collapse of edge e is built by three non planar planes in $\mathbb{R}^3$. Planes equations describe a linear system of constraints

$$\mathbf{Av} = \mathbf{b}$$

with $\mathbf{A} \in \mathbb{R}^{3x3}$, $\mathbf{v}, \mathbf{b} \in \mathbb{R}^3$. Let $\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3)^T$, then shall be

1. $\mathbf{a}_1 \neq \mathbf{0}$
2. $(\mathbf{a}_1^T \mathbf{a}_2)^2 < (\| \mathbf{a}_1 \| \| \mathbf{a}_2 \| \, cos(\alpha))^2$ (I choose $(\frac{a_1^T a_2}{\|a_1\|\|a_2\|})^2 < 0.97$)
3. $((\mathbf{a}_1 \times \mathbf{a}_2)^T \mathbf{a}_3)^2 > ((\| \mathbf{a}_1 \times \mathbf{a}_2 \| \| \mathbf{a}_3 \| \, sin(\alpha))^2$

Lindstrom-Turk strategy

Linear indipendent Constraints

Constraints are found imposing the minimization of geometric properties.

Several constraints can be obtained from the previous $a_i, i \in 0..3$ and a quadratic objective function $f(\mathbf{x})$ that we wish to minimize (quadratic programming Problem).
Given $f(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T\mathbf{A}\ \mathbf{x} - \mathbf{b}^T\mathbf{x} + \frac{1}{2}c$ subject to n constraints $(\mathbf{a}_i, b_i)$ then the additional 3-n constraints can be found solving

$$\mathbf{Q}(\mathbf{A}\mathbf{x} - \mathbf{b}) = \mathbf{0} \tag{1}$$

With $Q \in \mathbb{R}^{(3-n),3}$ formed by rows orthogonal to each other and a_i

Volume Preservation

Generating a new position for **v** usually changes the Volumes of the tethraedrons formed by $\lceil\lceil \mathbf{v}_1 \rceil\rceil \cup \lceil\lceil \mathbf{v}_2 \rceil\rceil$. In particular, the passage from $t = (\mathbf{v}_i^e, \mathbf{v}_1^t, \mathbf{v}_2^t)$ to $t' = (\mathbf{v}, \mathbf{v}_1^t, \mathbf{v}_2^t)$ moves v_i^e to a new position. During this movement, a Volume is swept. If this passage crosses the plane of t, then it is negative, positive otherwise. The costraint we build is

$$\sum_i V(\mathbf{v}, \mathbf{v}_0^{t_i}, \mathbf{v}_1^{t_i}, \mathbf{v}_2^{t_i}) = \frac{1}{6} \sum_i \begin{vmatrix} v_x & v_{0x}^{t_i} & v_{1x}^{t_i} & v_{2x}^{t_i} \\ v_y & v_{0y}^{t_i} & v_{1y}^{t_i} & v_{2y}^{t_i} \\ v_z & v_{0z}^{t_i} & v_{1z}^{t_i} & v_{2z}^{t_i} \\ 1 & 1 & 1 & 1 \end{vmatrix} = 0$$

$$\iff \sum_i (\mathbf{v}_0^{t_i} \times \mathbf{v}_1^{t_i} + \mathbf{v}_1^{t_i} \times \mathbf{v}_2^{t_i} + \mathbf{v}_2^{t_i} \times \mathbf{v}_0^{t_i})\mathbf{v} = \sum_i \mid \mathbf{v}_0^{t_i}, \mathbf{v}_1^{t_i}, \mathbf{v}_2^{t_i} \mid$$

Boundary Preservation

2D analog of previous constraint for boundary edges. We wish to minimize $\sum_i A(\mathbf{v}, \mathbf{v}_0^{e_i}, \mathbf{v}_1^{e_i})$.

$$\sum_i A(\mathbf{v}, \mathbf{v}_0^{e_i}, \mathbf{v}_1^{e_i}) = \sum_i \frac{1}{2} \parallel \mathbf{v} \times \mathbf{v}_0^{e_i} + \mathbf{v}_0^{e_i} \times \mathbf{v}_1^{e_i} + \mathbf{v}_1^{e_i} \times \mathbf{v} \parallel = 0$$

defining $\mathbf{e}_{1_i} = \mathbf{v}_1^{e_i} - \mathbf{v}_0^{e_i}$, $\mathbf{e}_{2_i} = \mathbf{v}_1^{e_i} \times \mathbf{v}_0^{e_i}$, $\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2$, then the solution space for the optimization problem can be written as:

1. $\mathbf{e}_1^T \mathbf{e}_1 \mathbf{e}_3 \mathbf{v} + \mathbf{e}_3^T \mathbf{e}_3 = 0$
2. $(\mathbf{e}_1 \times \mathbf{e}_3)\mathbf{v} = 0$

Volume Optimization

We minimize the sum of unsigned volumes:

$$f_V(e, \mathbf{v}) = \sum_i V(\mathbf{v}, \mathbf{v}_0^{t_i}, \mathbf{v}_1^{t_i}, \mathbf{v}_2^{t_i})^2$$

This constraint can be rewritten in a quadratic optimization problem of the form

$$\frac{1}{18}\left[\mathbf{v}^T(\sum_i \mathbf{n}_i\mathbf{n}_i^T)\mathbf{v} - \sum_i[\mathbf{v}_0^{t_i}, \mathbf{v}_1^{t_i}, \mathbf{v}_2^{t_i}]\mathbf{n}_i^T v + \frac{1}{2}\sum_i[\mathbf{v}_0^{t_i}, \mathbf{v}_1^{t_i}, \mathbf{v}_2^{t_i}]^2\right] \qquad (2)$$

Boundary Optimization

2d analog of previous constraint for boundary edges. We minimize the sum of unsigned areas $f_S(e, \mathbf{v}) = \sum_i A(\mathbf{v}, \mathbf{v}_0^{e_i}, \mathbf{v}_1^{e_i})^2$. Trasforming into a quadratic programming problem

$$\frac{1}{2}\left[\frac{1}{2}\mathbf{v}^T \sum_i (\times \mathbf{e}_{1_i})(\mathbf{e}_{1_i} \times)\mathbf{v} + \sum_i (\mathbf{e}_{1_i} \times \mathbf{e}_{2_i})\mathbf{v} + \frac{1}{2}\sum_i (\mathbf{e}_{2_i}^T \mathbf{e}_{2_i})\right] \tag{3}$$

where

$(\mathbf{v}\times) = \begin{pmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{pmatrix}$ and $(\times\mathbf{v}) = (\mathbf{v}\times)^T$.

Triangle Shape Optimization

It is possible that after the previous constraints, there are not yet three linear indipendent ones. Another geometric property in this situations that can be imposed is to form as much equilater possible triangles with the new solution $\mathbf{v}$. This can be writter minimazing $\sum_i L(\mathbf{v}, \mathbf{v}_i)^2$, namely the sum of quadratic distances from the solution to adjacent vertices.
The corresponding quadratic programming problem is

$$\left[\frac{1}{2}\mathbf{v}^T \sum_i \mathbf{I}\mathbf{v} - \sum_i \mathbf{v}_i^T \mathbf{v} + \frac{1}{2} \sum_i \mathbf{v}_i^T \mathbf{v}_i\right] \tag{4}$$

Lindstrom-Turk strategy

Order to consider the constraints:

	method	*constraints*
1.	volume preservation	≤ 1
2.	boundary preservation	≤ 2
3.	volume/boundary optimization	≤ 3
4.	triangle shape optimization	≤ 3

Edge Collapse Priority

Edge collapse priority associated with vertex placement v
$f_c(e, \mathbf{v}) = \lambda f_V(e, \mathbf{v}) + (1 - \lambda) L^2(e) f_S(e, \mathbf{v})$ where L(e) is the length of the edge e, $\lambda \in [0, 1]$.

Garland-Heckbert strategy

Squared distance point - plane

Given a point $\mathbf{p} = (x, y, z, 1)^T$ and plane $\mathbf{q} = (a, b, c, d)^T$, we consider

$$dist(\mathbf{q}, \mathbf{p})^2 = (\mathbf{q}^T \mathbf{p})^2 = \mathbf{p}^T (\mathbf{q}\ \mathbf{q}^T) \mathbf{p}$$

Where

$$\mathbf{q}\mathbf{q}^T = \mathbf{Q} = \begin{pmatrix} a^2 & ab & ac & ad \\ ba & b^2 & bc & bd \\ ca & cb & c^2 & cd \\ da & db & dc & d^2 \end{pmatrix}$$

Garland-Heckbert strategy

Quadric Error Metrics

Quadric Error Metrics chooses the point p that minimizes the squared distances to the planes of adjacent triangles.

Minimize $f(\mathbf{p}) = \sum_i dist(\mathbf{q}_i, \mathbf{p})^2 = \mathbf{p}^T(\sum_i \mathbf{Q}_i)\mathbf{p} = \mathbf{p}^T \mathbf{Q}_q \mathbf{p}$
Thus

$$\nabla f = (\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}) = 0 \iff \begin{pmatrix} a^2 & ab & ac & ad \\ ba & b^2 & bc & bd \\ ca & cb & c^2 & cd \\ 0 & 0 & 0 & 1 \end{pmatrix} p^* = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Garland-Heckbert strategy

- $\mathbf{Q}'$ might be singular, therefore a solution could not be found for edge contraption
- Switch to a simple strategy like mid-point or discard the edge

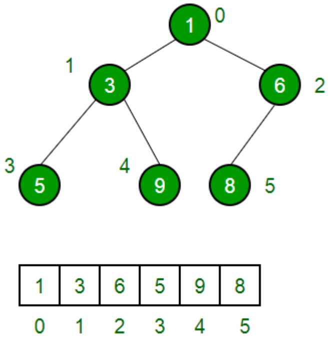
Edge collapse Priority

The cost associated to the solution p* found is

$$\Delta(\mathbf{v}) = \sum_{\mathbf{p} \in planes(\mathbf{v})} (\mathbf{p}^T \mathbf{v})^2$$

Implementation Overview

1. Select a **strategy** and a mesh file
2. Build a graph containing vertices, edges, triangles
3. Compute a position for each edge and a cost
4. Build a min-heap for edges with priority equal to cost (quicksort)
5. Simplify a edge linking incident objects from $\mathbf{v}_2$ to $\mathbf{v}_1$
6. Update coordinates of $\mathbf{v}_1$
7. **Relax** adjacent edges



Hand of Bones (98.5% reduction)

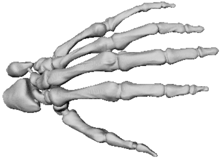


Figure: 654.666 Triangles

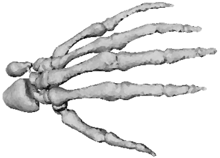


Figure: Mid-point: 9878 Triangles

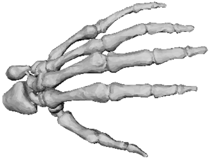


Figure: L-T: 9839 Triangles

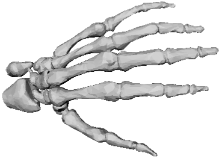


Figure: Garland: 9538 Triangles

Dragon (83% reduction)

Figure: 871.414 Triangles

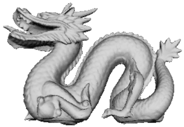


Figure: Mid-point: 151.419 Triangles

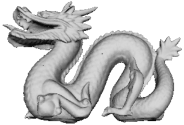


Figure: L-T: 147.972 Triangles

Figure: Garland: 144.106 Triangles

Cow (95% reduction)

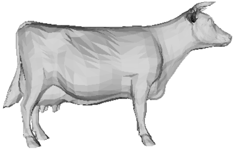


Figure: 5520 Triangles

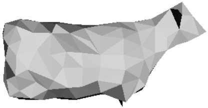


Figure: Mid-point: 294 Triangles

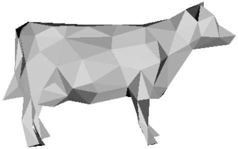


Figure: L-T: 294 Triangles

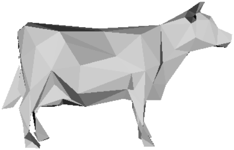


Figure: Garland: 295 Triangles

Bibliography I

P. Lindstrom, G. Turk
Fast and Memory Efficient Polygonal Simplification.
Proceedings Visualization'98, 279-286, 1998.

P. Lindstrom, G. Turk
Evaluation of Memoryless Simplification.
IEEE Transactions on Visualization and Computer Graphics 5 (2), 98-115, 1999.

M. Garland, P. Heckbert
Surface Simplification Using Quadric Error Metrics.
Proceedings of the 24th annual conference on Computer graphics and interactive techniques, 209-216, 1997.

F.Cacciola, Surface Mesh Simplification,
Computational Geometry Algorithm Library