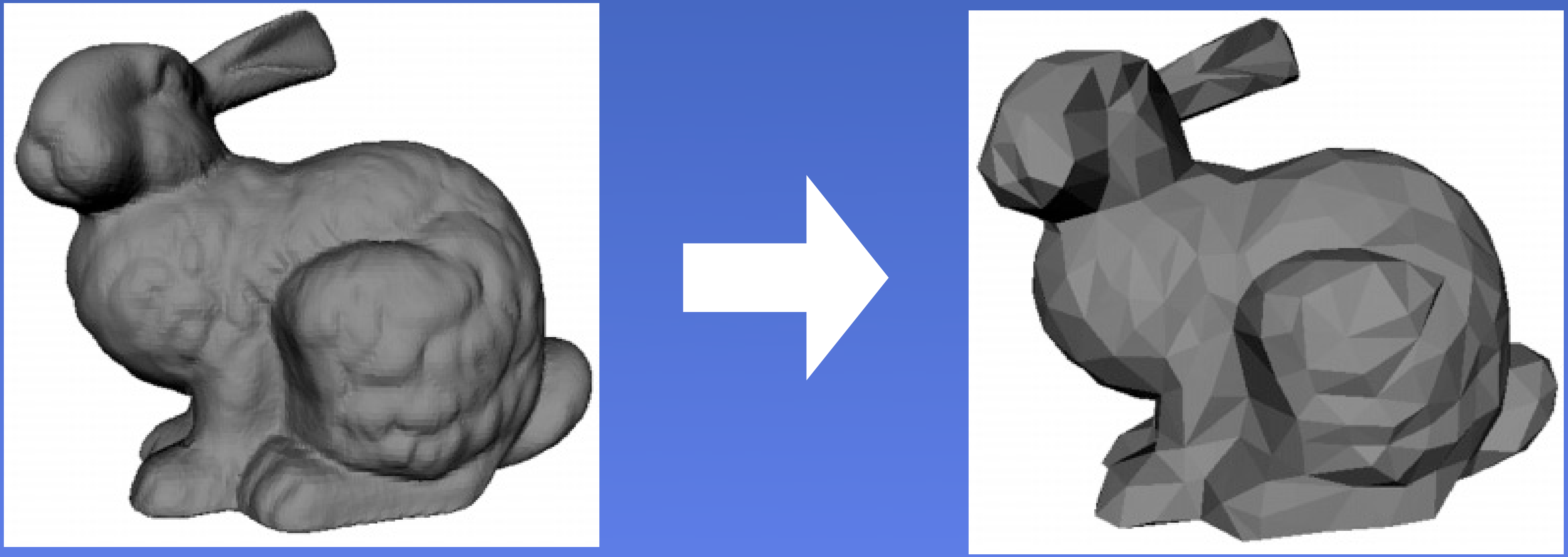
Surface Simplification Using Quadric Error Metrics

Michael Garland and Paul S. Heckbert

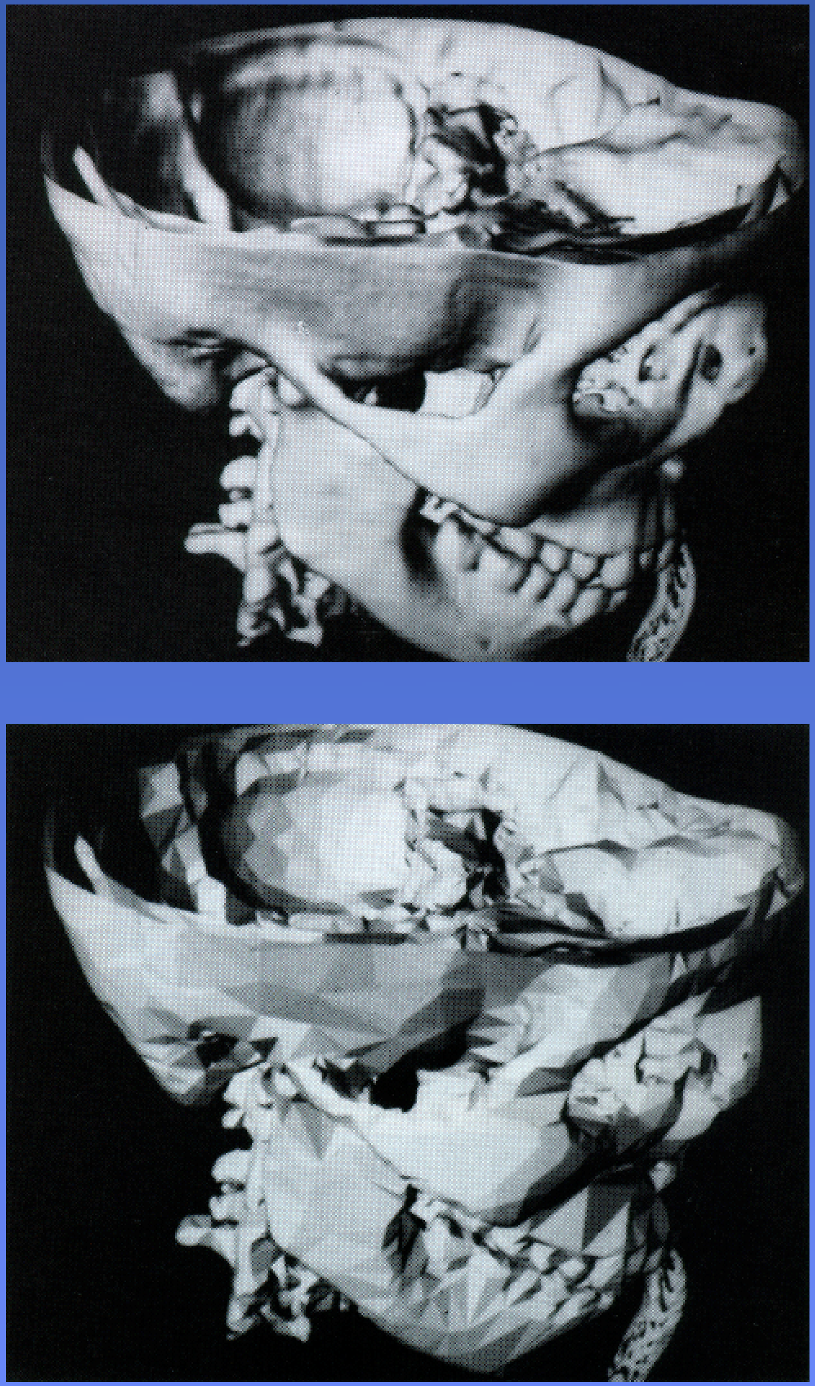
The Basic Idea



Vertex Decimation: [Schroeder92]

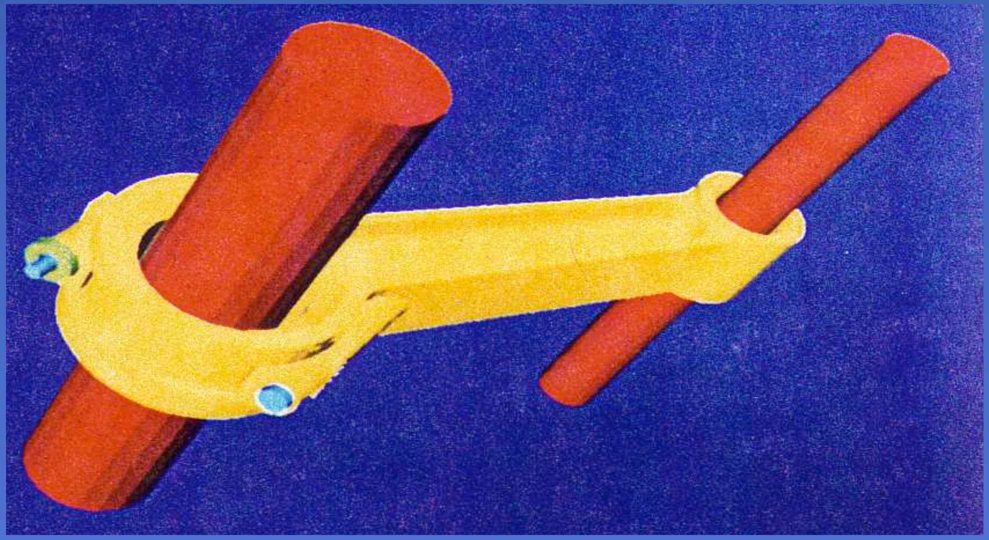
- Classify vertices as simple, complex, boundary, interior edge, or corner vertex.
- Iteratively remove vertices that meet some decimation criteria.
- Triangulate resulting holes.

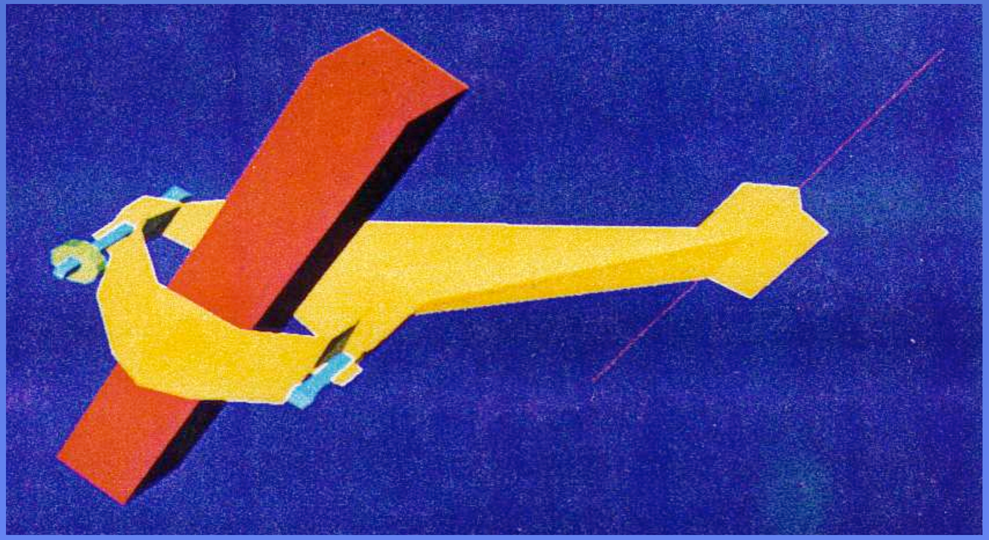
Restricted to manifold surfaces.
Carefully preserves topology.



Vertex Clustering: [Rossignac92]

- Weight vertices based on perceptual importance.
- Create bounding box and subdivide into grid.
- Perform weighted clustering of vertices in each cell.



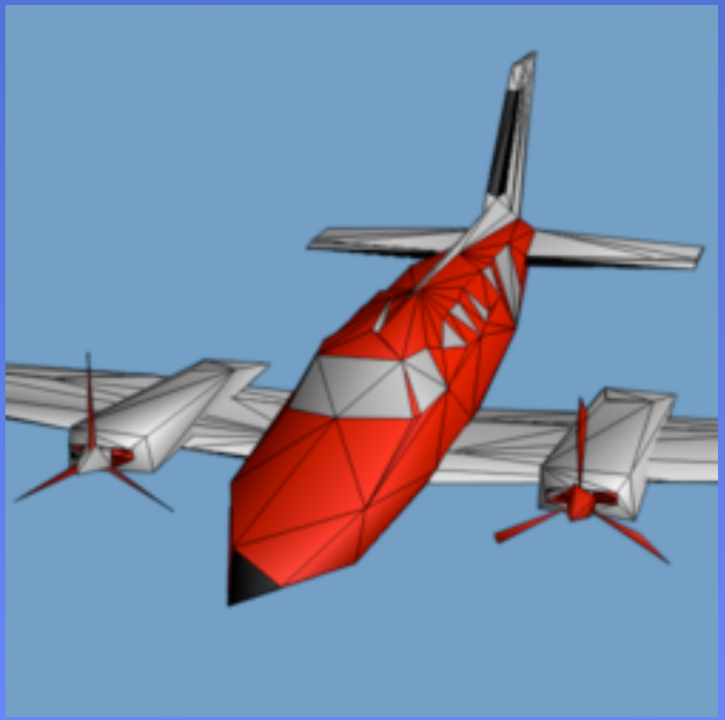


Very fast.
Works on non-manifold geometry.
May drastically alter topology.
Visually unappealing.
Difficult to produce models with N faces.

Iterative Edge Contraction: [Hoppe96] (and others)

- Define the cost of contracting an edge.
- Iteratively contract the edge with lowest cost.

High quality results.
Cost functions can be complex.
Can close holes.
Can't join disconnected components.



The Solution:

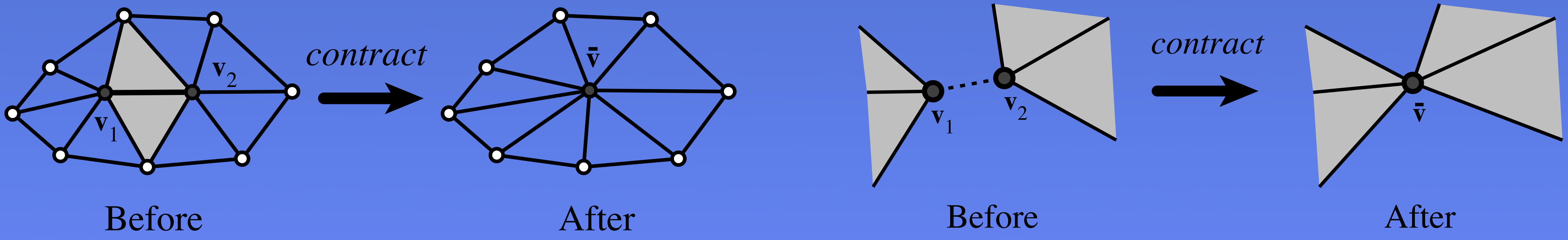
Iterative Pair Contraction with the Quadric Error Metric

- Works on non-manifold geometry
- Supports aggregation
- Can be implemented efficiently
- Produces high quality approximations

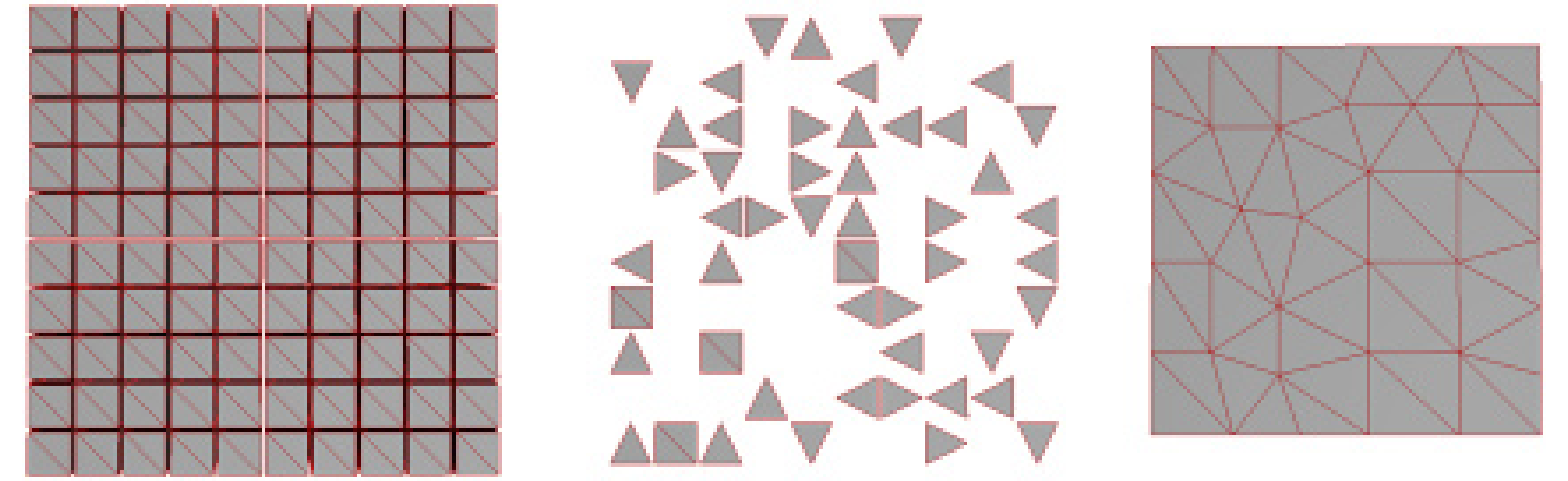
Iterative Pair Contraction

A pair of vertices $(\mathbf{v_1}, \mathbf{v_2})$ are **valid** for contraction if:

1. $(\mathbf{v_1}, \mathbf{v_2})$ is an edge, *or*
2. $||\mathbf{v_1} - \mathbf{v_2}|| < t$ for some threshold t



Benefits of Pair Contraction



- Can join unconnected components
- Can result in much nicer approximations

Error Metric

[Ronfard96] suggested the following:

- Each vertex is the intersection of a set of planes.
- Define the error at a vertex to be the sum of the squared distances to its planes.

$$\Delta(\mathbf{v}) = \Delta([v_x \; v_y \; v_z \; 1]^T) = \sum_{\mathbf{p} \in planes(\mathbf{v})} (\mathbf{p}^T \mathbf{v})^2$$

Where $\mathbf{p} = [a \; b \; c \; d]^T$ represents the plane $ax + by + cz + d = 0$

with $a^2 + b^2 + c^2 = 1$

Error Metric (2)

$$\begin{aligned}
\Delta(\mathbf{v}) &= \sum_{\mathbf{p}\in planes(\mathbf{v})} (\mathbf{p}^T\mathbf{v})^2 \\
&= \sum_{\mathbf{p}\in planes(\mathbf{v})} (\mathbf{p}^T\mathbf{v})^T(\mathbf{p}^T\mathbf{v}) \\
&= \sum_{\mathbf{p}\in planes(\mathbf{v})} (\mathbf{v}^T\mathbf{p})(\mathbf{p}^T\mathbf{v}) \\
&= \sum_{\mathbf{p}\in planes(\mathbf{v})} \mathbf{v}^T(\mathbf{p}\mathbf{p}^T)\mathbf{v} \\
&= \mathbf{v}^T\left(\sum_{\mathbf{p}\in planes(\mathbf{v})} (\mathbf{p}\mathbf{p}^T)\right)\mathbf{v}
\end{aligned}$$

Error Metric (3)

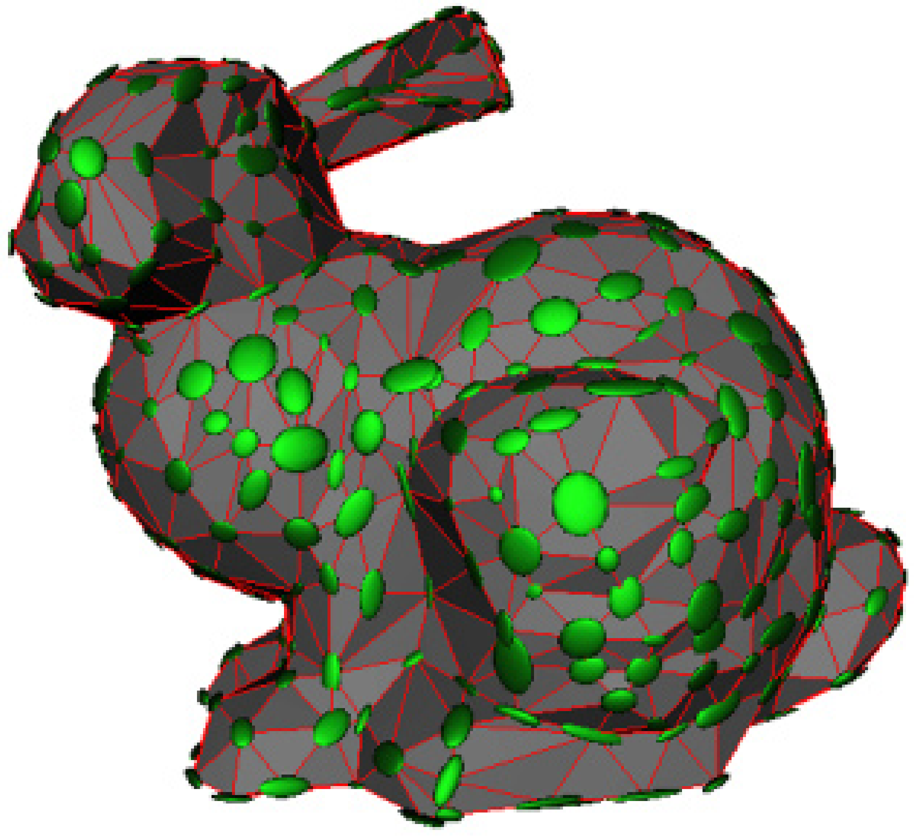
$$\Delta(\mathbf{v}) = \mathbf{v}^T \left(\sum_{\mathbf{p} \in planes(\mathbf{v})} (\mathbf{p}\mathbf{p}^T) \right) \mathbf{v}$$

$$= \mathbf{v}^T \left(\sum_{\mathbf{p} \in planes(\mathbf{v})} \mathbf{K}_\mathbf{p} \right) \mathbf{v}$$

Where $\mathbf{K}_\mathbf{p} = \mathbf{p}\mathbf{p}^T = \begin{bmatrix} a^2 & ab & ac & ad \\ ba & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix}$

$\mathbf{K}_\mathbf{p}$ is the **fundamental error quadric.**

Error Metric (4)



- For each vertex $\mathbf{v_i}$ store a symmetric 4x4 matrix $\mathbf{Q_i}$.
- For a given contraction $(\mathbf{v_1}, \mathbf{v_2}) \rightarrow \bar{\mathbf{v}}$, let $\bar{\mathbf{Q}} = \mathbf{Q_1} + \mathbf{Q_2}$.
- The matrices $\mathbf{Q_i}$ are called quadrics because the level sets of $\Delta(\mathbf{v}) = \epsilon$ form quadric surfaces (usually ellipsoids).

More on Quadrics

$$\mathbf{v_h} = [v_x \; v_y \; v_z \; 1]^T \quad \mathbf{p} = [a \; b \; c \; d]^T$$

$$\begin{aligned} D^2(\mathbf{v_h}) = (\mathbf{p}^T\mathbf{v_h})^2 &= (\mathbf{n}^T\mathbf{v} + d)^2 \quad \text{where} \quad \mathbf{n} = [a \; b \; c]^T \\ &= (\mathbf{v}^T\mathbf{n} + d)(\mathbf{n}^T\mathbf{v} + d) \\ &= (\mathbf{v}^T\mathbf{n}\mathbf{n}^T\mathbf{v} + 2d\mathbf{n}^T\mathbf{v} + d^2) \\ &= (\mathbf{v}^T(\mathbf{n}\mathbf{n}^T)\mathbf{v} + 2(d\mathbf{n})^T\mathbf{v} + d^2) \end{aligned}$$

$$\mathbf{X} = \mathbf{n}\mathbf{n}^T = \begin{bmatrix} a^2 & ab & ac \\ ba & b^2 & bc \\ ac & bc & c^2 \end{bmatrix} \quad \mathbf{y} = d\mathbf{n} = [da \; db \; dc]^T \quad z = d^2$$

More on Quadrics (2)

$$\mathbf{Q} = \begin{bmatrix} a^2 & ab & ac & ad \\ ba & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix} = Q(\mathbf{X}, \mathbf{y}, z)$$

$$\mathbf{X} = \mathbf{n}\mathbf{n}^T = \begin{bmatrix} a^2 & ab & ac \\ ba & b^2 & bc \\ ac & bc & c^2 \end{bmatrix} \quad \mathbf{y} = d\mathbf{n} = [da\ db\ dc]^T \quad z = d^2$$

$$\Delta(\mathbf{v}) = \mathbf{v}^T\mathbf{Q}\mathbf{v} = \mathbf{v}^T\mathbf{X}\mathbf{v} + 2\mathbf{y}^T\mathbf{v} + z$$

Performing Contractions

To perform a contraction $(\mathbf{v_1}, \mathbf{v_2}) \rightarrow \bar{\mathbf{v}}$, we must find $\bar{\mathbf{v}}$.

Specifically, we want $\nabla(\Delta(\bar{\mathbf{v}})) = 0.$

$$\nabla(\Delta(\bar{\mathbf{v}})) = 2\mathbf{X}\bar{\mathbf{v}} + 2\mathbf{y}$$

$$2\mathbf{X}\bar{\mathbf{v}} + 2\mathbf{y} = 0 \implies \bar{\mathbf{v}} = -\mathbf{X}^{-\mathbf{1}}\mathbf{y}$$

The associated minimum error is:

$$\Delta(\bar{\mathbf{v}}) = \mathbf{y}^T\bar{\mathbf{v}} + z = -\mathbf{y}^T\mathbf{X}^{-\mathbf{1}}\mathbf{y} + z$$

Algorithm Summary

- Compute initial quadrics for each vertex.
- Select all valid pairs.
- Compute the optimal contraction target for each pair and let its associated error be the **cost** of the contraction.
- Place all pairs in a keyed heap on cost with the minimum cost pair at the top.
- Iteratively remove the pair with least cost from the heap, contract the pair, and update the cost of all valid pairs involving this contracted vertex.

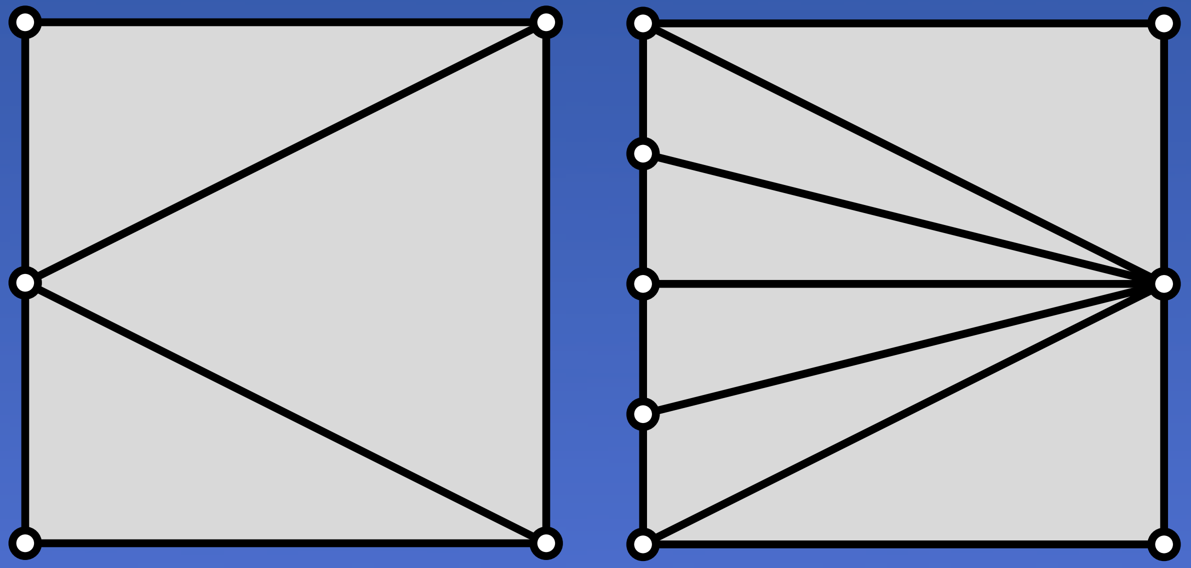
Considerations

- Implicitly tracking set of planes associated with a vertex via matrix addition.
- **Should** use set union instead:

$$(planes(v_1) \cup planes(v_2)) \neq (\mathbf{Q}_1 + \mathbf{Q}_2)$$

- But, a single plane can be counted at most 3 times.
- Resulting imprecision is largely tolerable.

Considerations (2)

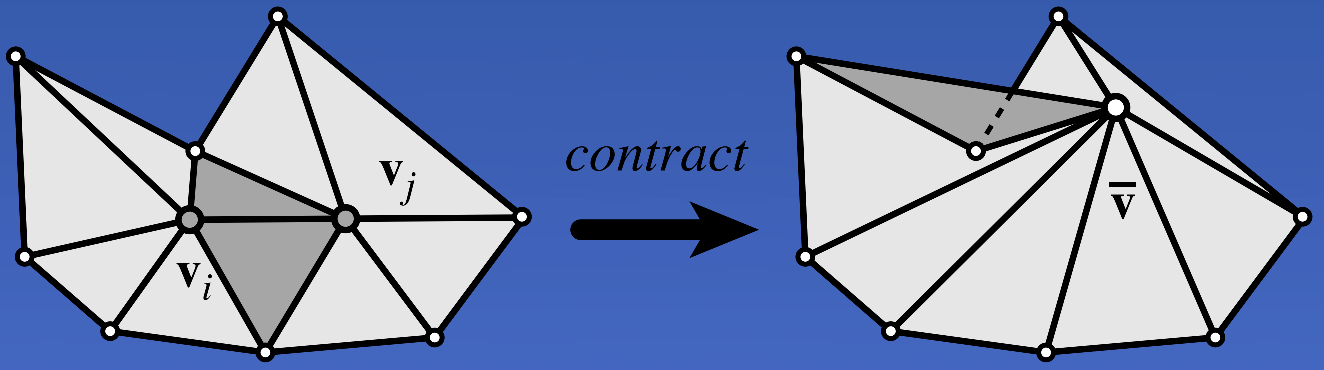


- As proposed in the paper, the algorithm is very sensitive to tessellation.
- In practice, weight each quadric according to area as in [Garland99].

Considerations (3)

- When we wish to preserve boundaries, we must create perpendicular planes to boundary edges.
- Then, weight the associated fundamental quadrics appropriately to penalize movement away from the boundary.

Considerations (4)



- Contractions may invert the mesh.
- The paper proposes penalizing contractions where the normal of a face changes by more than some threshold value.
- A better solution is described in [Garland99], which defines the region the contracted vertex may occupy without causing foldover.

Considerations (5)

$$\bar{\mathbf{v}} = -\mathbf{X}^{-1}\mathbf{y}$$

- Computing inverses is bad: use Cholesky decomposition (since $\mathbf{X} = \mathbf{L}\mathbf{L}^*$ is positive semidefinite, by construction).
- What if $\mathbf{X}$ is (nearly) singular?
  - Can use SVD to project vertex onto the solution space.
  - In practice, look along line between source vertices or just pick whichever source vertex minimizes the error.

Considerations (6)

Evaluating $\Delta(\mathrm{v})$ as proposed not stable with floats [Ju02].

Compute a sequence of givens rotations $\mathbf{G}$ s.t.:

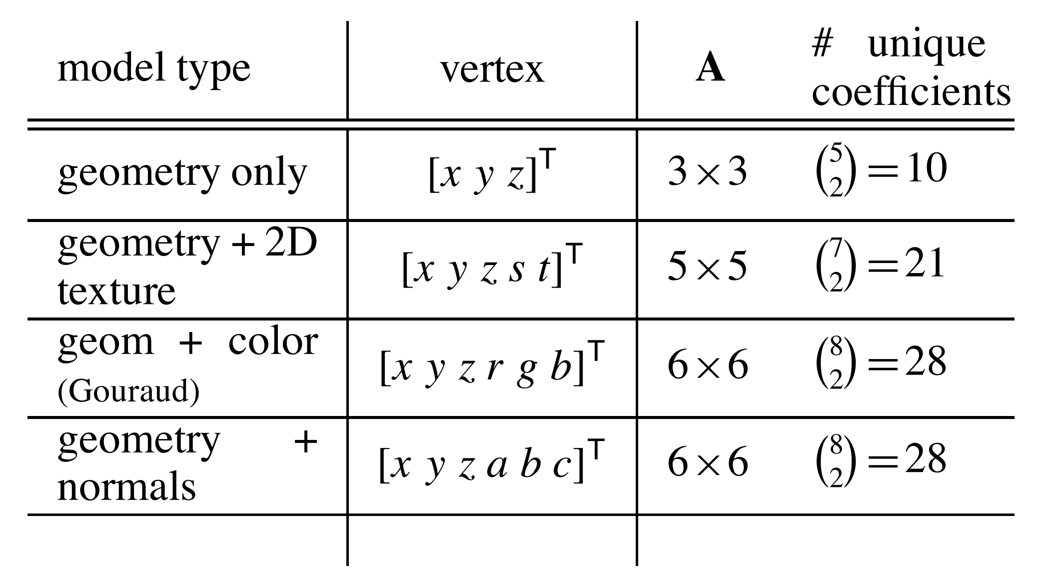
$$\mathbf{G}(\mathbf{X}\ \mathbf{y}) = \begin{pmatrix} x & x & x & x \\ 0 & x & x & x \\ 0 & 0 & x & x \\ 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots \end{pmatrix} = \begin{pmatrix} \hat{\mathbf{X}} & \hat{\mathbf{y}} \\ 0 & z \\ 0 & 0 \\ \dots & \dots \end{pmatrix}$$

$$\bar{\mathbf{v}} = -\mathbf{X}^{-1}\mathbf{y} = -\hat{\mathbf{X}}^{-1}\hat{\mathbf{y}}$$

Merging quadrics also accomplished via orthogonal transformations...

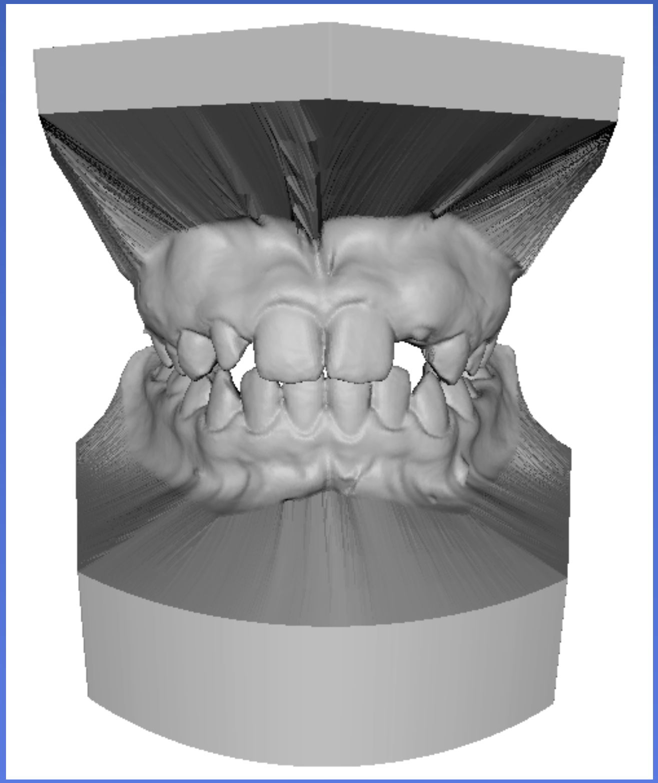
Considerations (7)

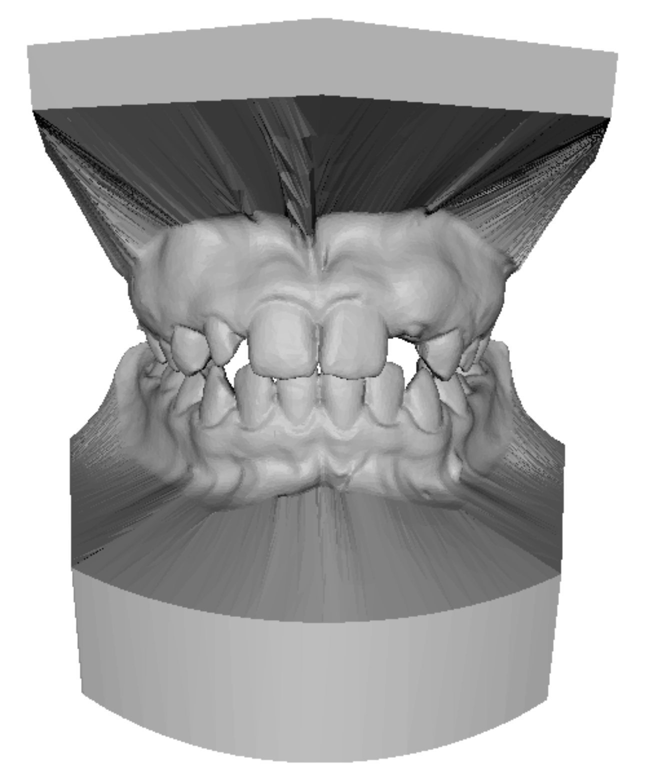
- Works only for geometric models.
- No support for color, texture, other vertex attributes.
- Could use segregated error quadrics for each feature.
- [Garland98] proposes generalization of error quadrics to higher dimensions.
- Increases storage/computational requirements:

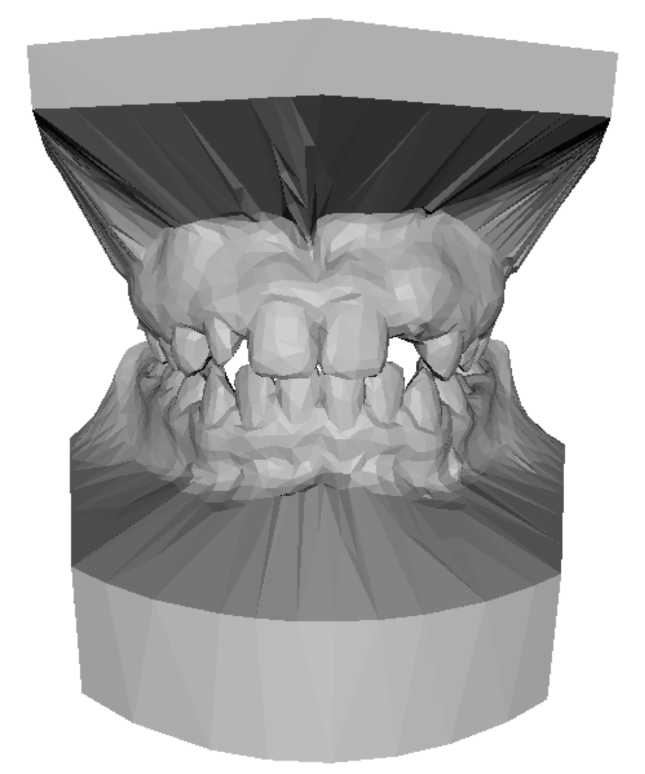


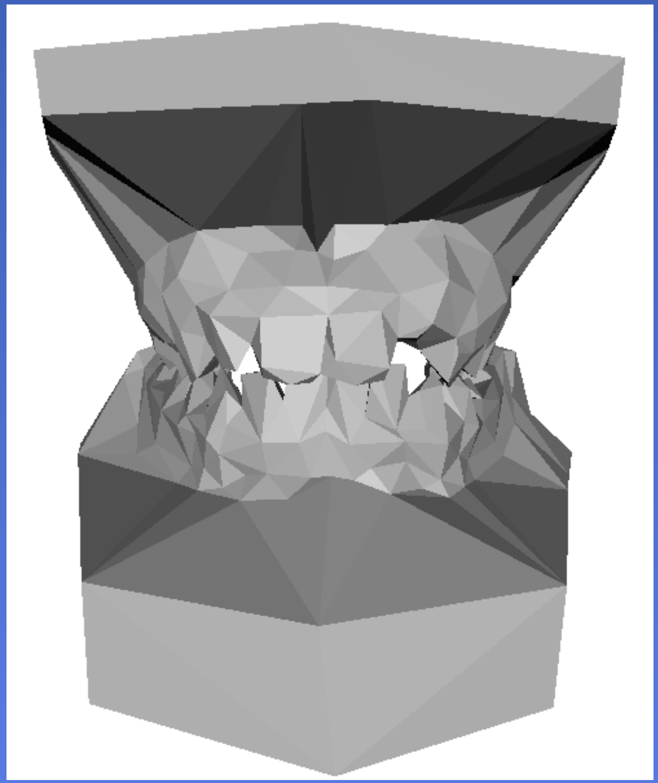
model type	vertex	**A**	# unique coefficients
geometry only	$[x\ y\ z]^\mathsf{T}$	3×3	$\binom{5}{2} = 10$
geometry + 2D texture	$[x\ y\ z\ s\ t]^\mathsf{T}$	5×5	$\binom{7}{2} = 21$
geom + color (Gouraud)	$[x\ y\ z\ r\ g\ b]^\mathsf{T}$	6×6	$\binom{8}{2} = 28$
geometry + normals	$[x\ y\ z\ a\ b\ c]^\mathsf{T}$	6×6	$\binom{8}{2} = 28$

Results

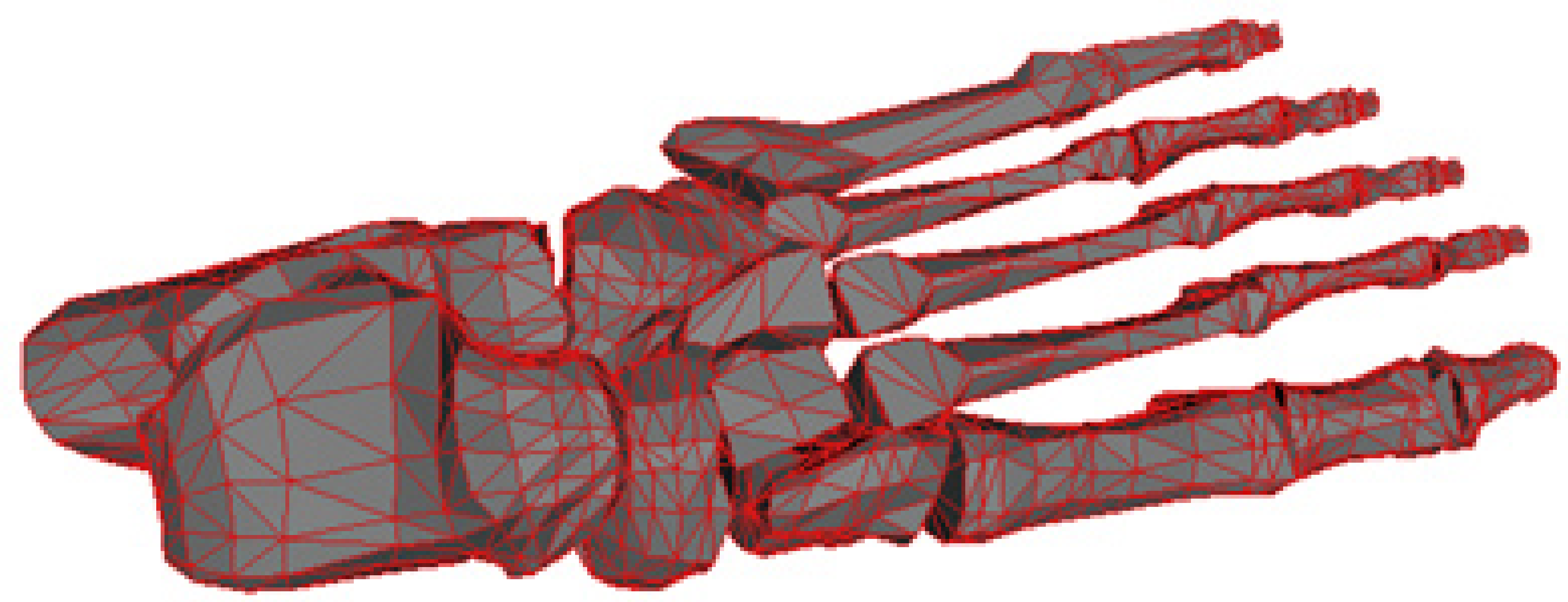

424,376

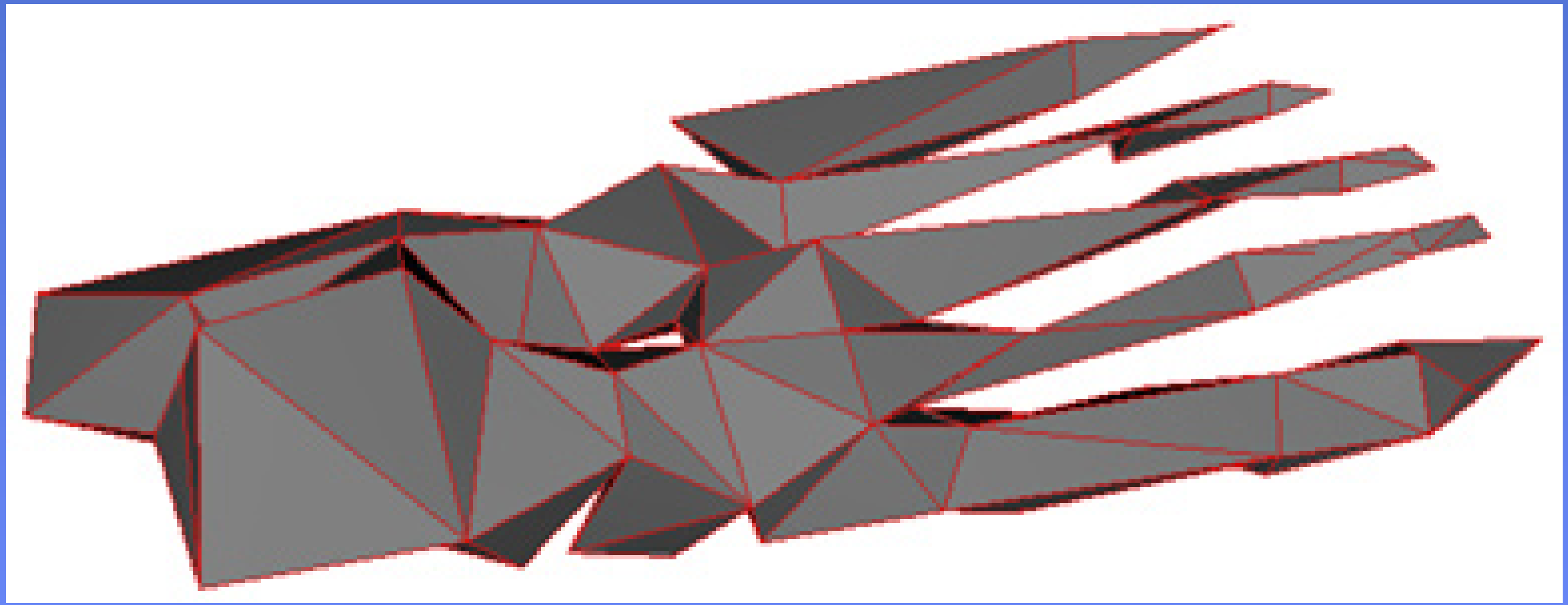

60,000


8,000

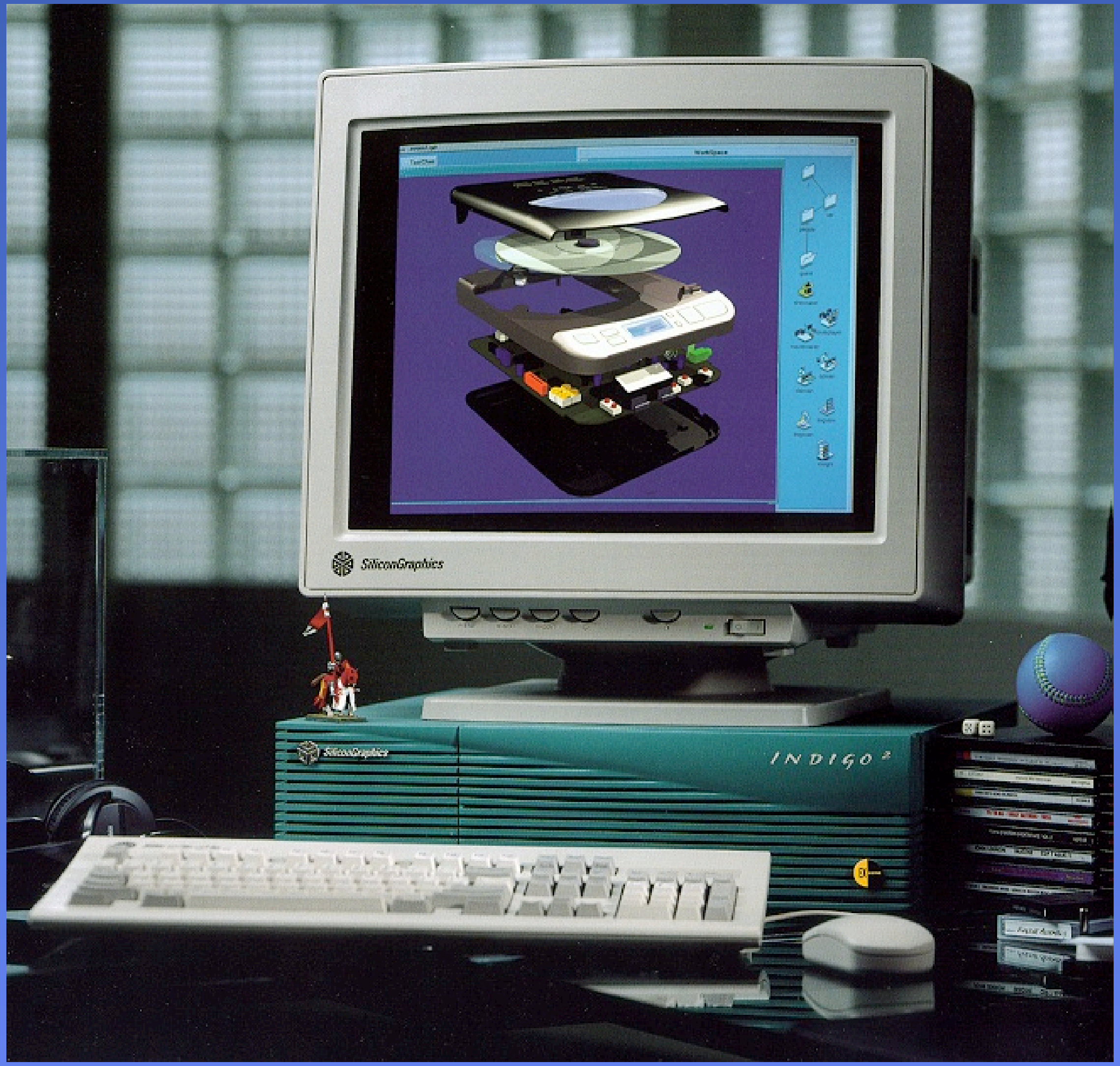

1,000

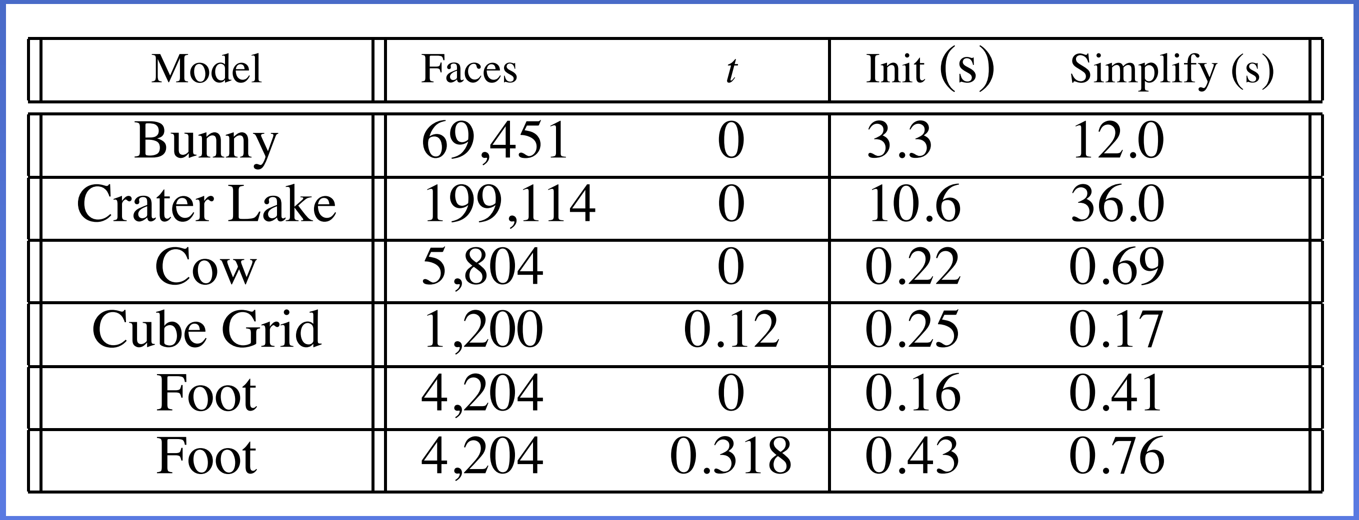
Results (2)





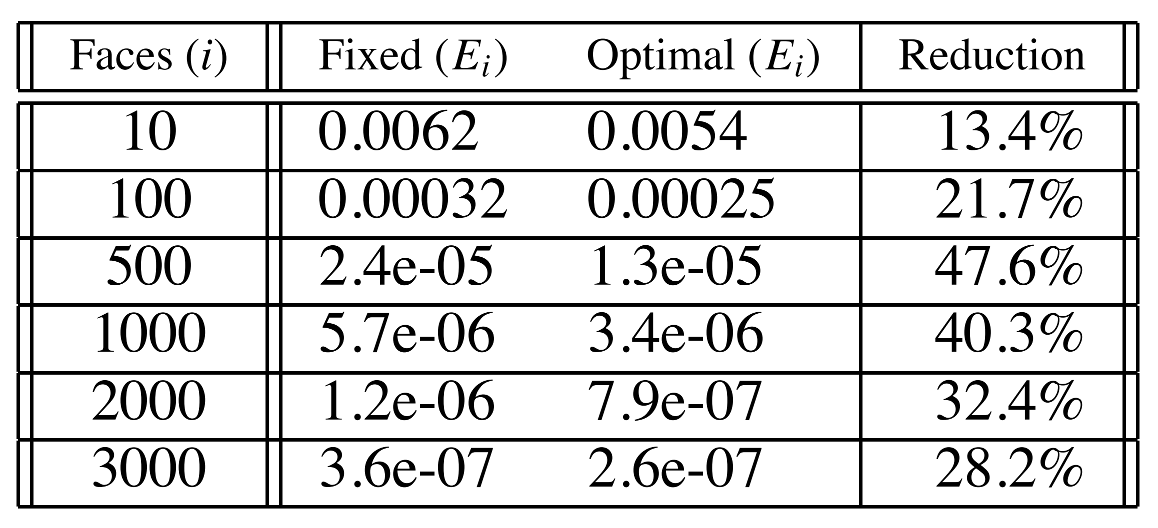
Results (3)





Model	Faces	*t*	Init (s)	Simplify (s)
Bunny	69,451	0	3.3	12.0
Crater Lake	199,114	0	10.6	36.0
Cow	5,804	0	0.22	0.69
Cube Grid	1,200	0.12	0.25	0.17
Foot	4,204	0	0.16	0.41
Foot	4,204	0.318	0.43	0.76

Results (3)



Faces (i)	Fixed (E_i)	Optimal (E_i)	Reduction
10	0.0062	0.0054	13.4%
100	0.00032	0.00025	21.7%
500	2.4e-05	1.3e-05	47.6%
1000	5.7e-06	3.4e-06	40.3%
2000	1.2e-06	7.9e-07	32.4%
3000	3.6e-07	2.6e-07	28.2%

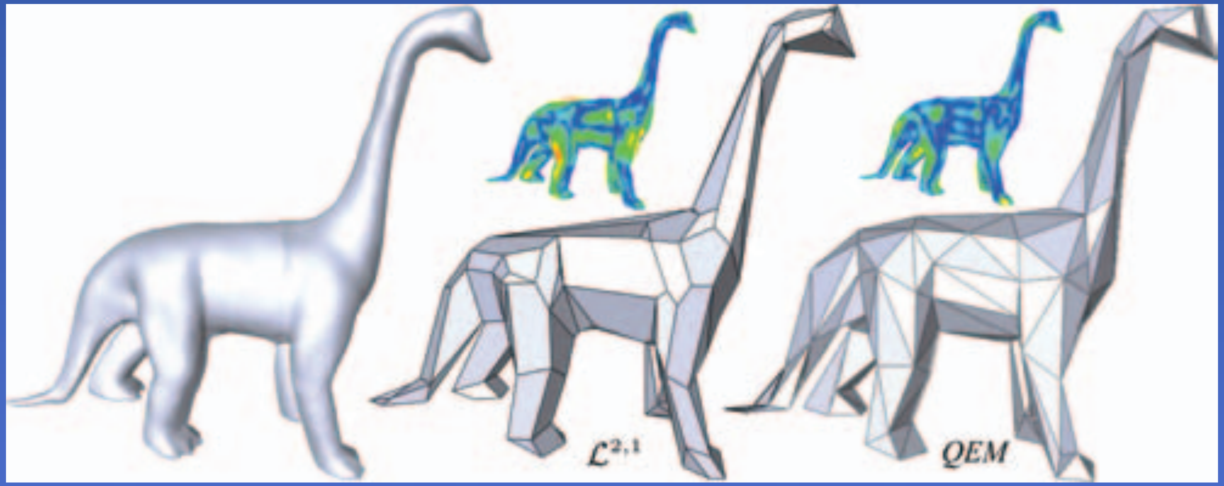
$$E_i = E(M_n, M_i) = \frac{1}{|X_n| + |X_i|} \left(\sum_{v \in X_n} d^2(v, M_i) + \sum_{v \in X_i} d^2(v, M_n) \right)$$

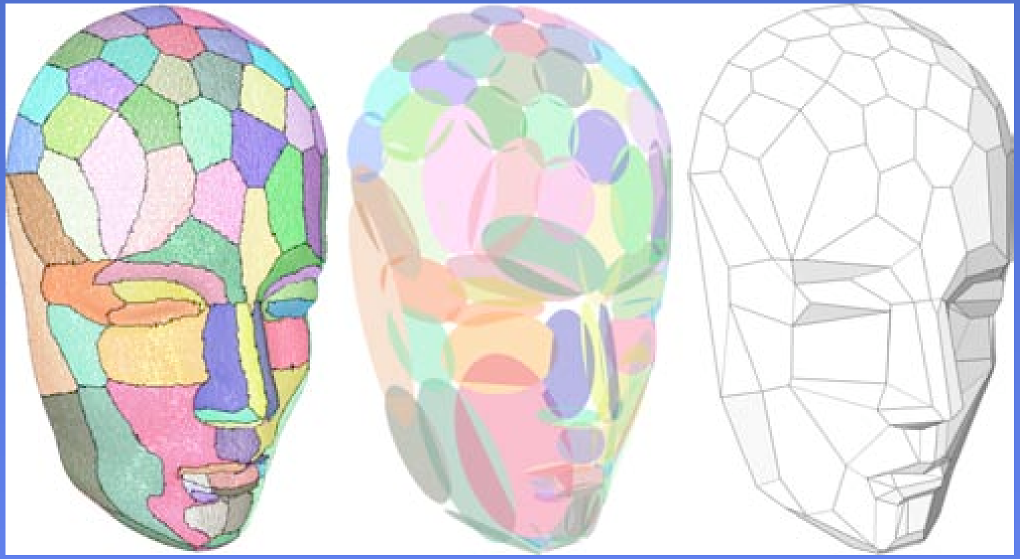
where $d(v, M) = \min_{p \in M} \|v - p\|$

(This is just the average squared distance between models)

Variational Shape Approximation: [Cohen-Steiner04]

- Formulate surface simplification as an optimization problem.
- Use clustering to fit local *shape proxies* to surface.
- Use these proxies to produce approximating surfaces.





Better approximation than QER.
Much slower than QER.

References

- M. Garland and P. Heckbert. Surface simplification using quadric error metrics. In Proceedings of SIGGRAPH 97, pp. 209–216, August 1997.
- W. Schroeder, J. Zarge, and W. Lorenson. Decimation of triangle meshes. Proceedings of SIGGRAPH 92, pp.65–70, August 1992.
- H. Hoppe. Progressive meshes. In Proceedings of SIGGRAPH 96, pp. 99–108, August 1996.
- J. Rossignax and P. Borrel. Multi-resolution 3D approximations for rendering complex scenes. In Modeling in Computer Graphics: Methods and Applications, pp. 455-465, 1993.
- Michael Garland. Quadric-Based Polygonal Surface Simplification. Ph.D. dissertation, Computer Science Department, Carnegie Mellon University, CMU-CS-99-105, May 1999.
- T. Ju, F. Losasso, S. Schaefer, and J. Warren. Dual contouring of Hermite data. In Proceedings of SIGGRAPH 02, pp. 339–346, 2002.
- D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In Proceedings of SIGGRAPH 04, pp. 905–914, August 2004.
- M. Garland and P. Heckbert. Simplifying Surfaces with Color and Texture using Quadric Error Metrics. In Proceedings of IEEE Visualization 98. 1998.