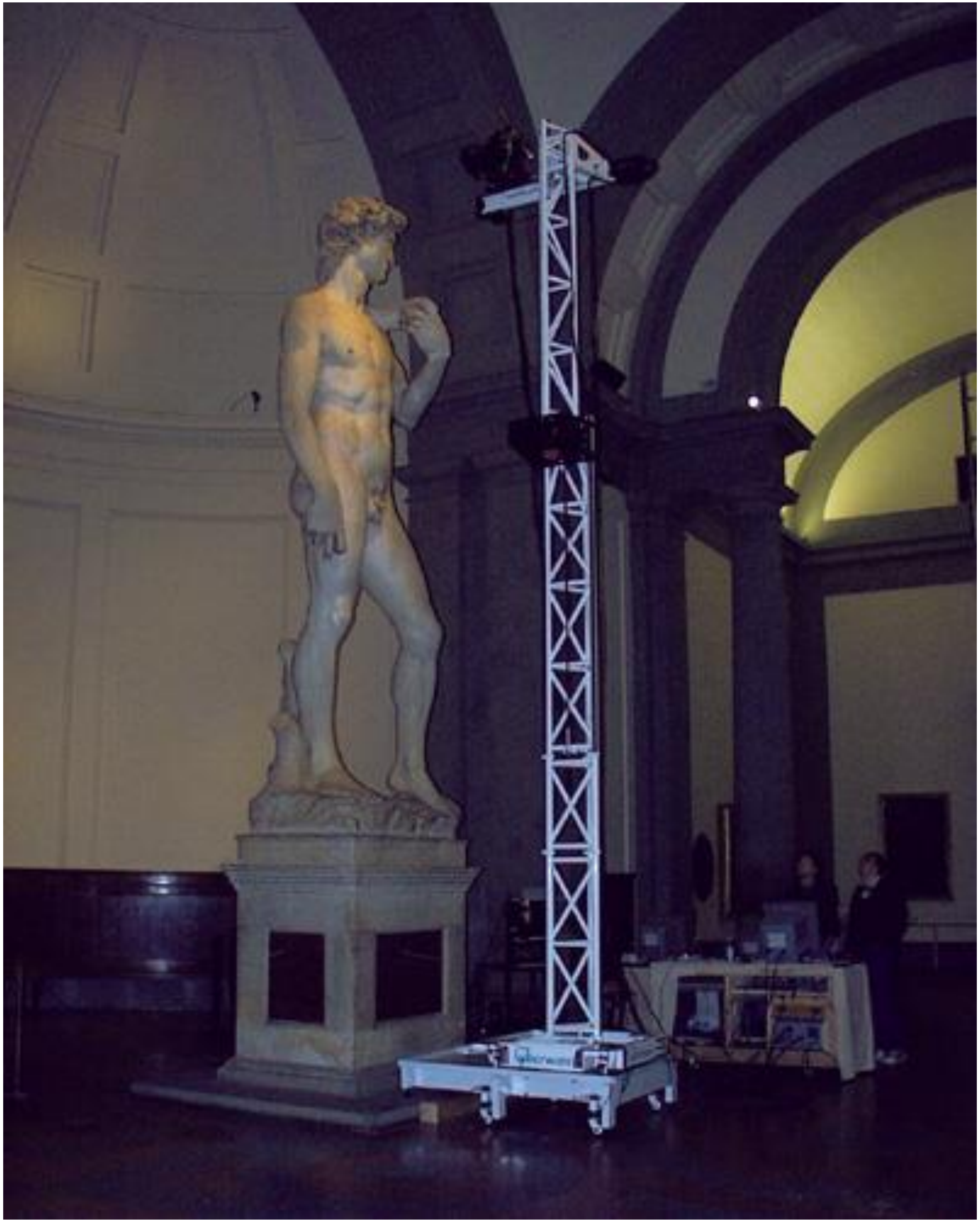
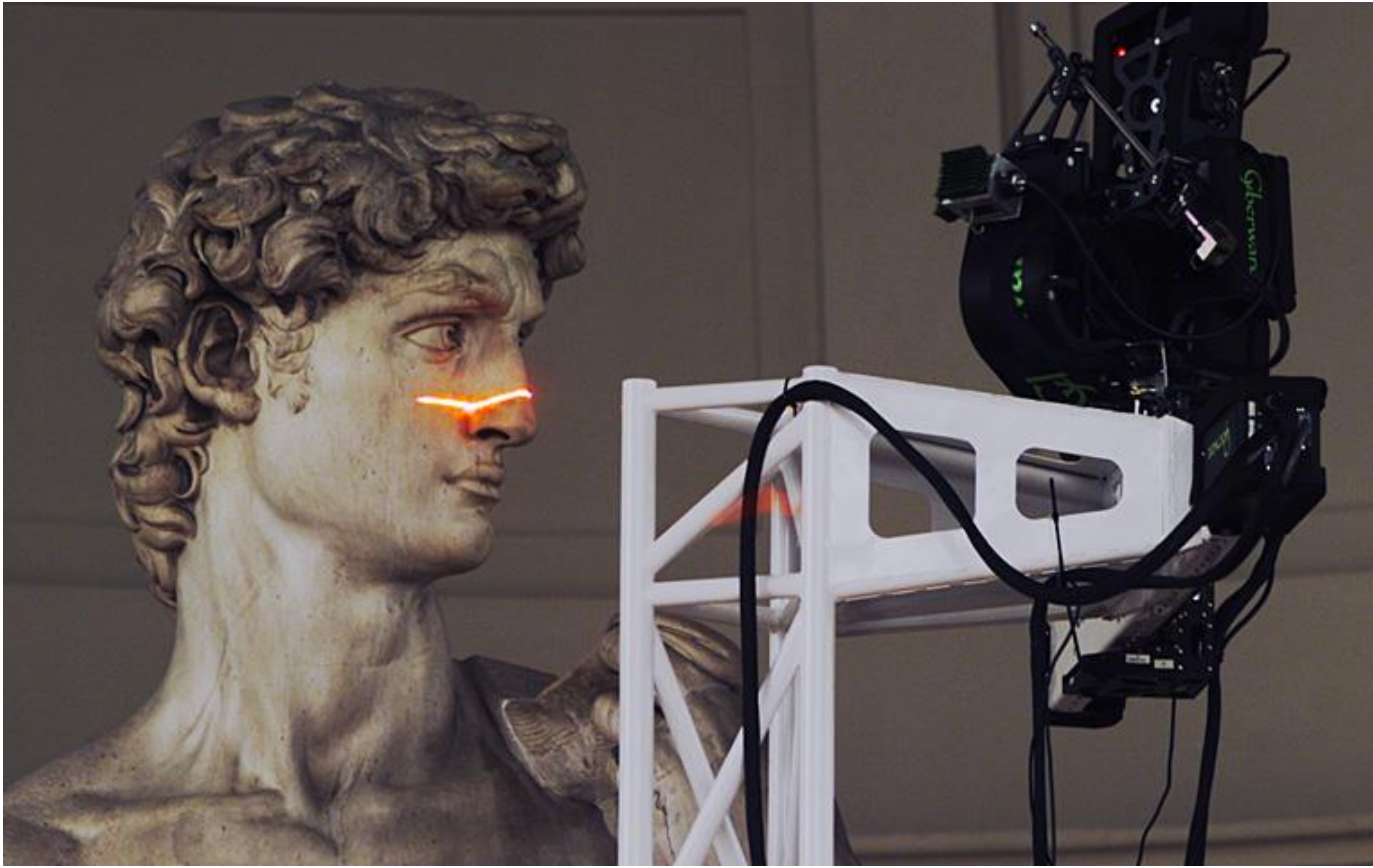
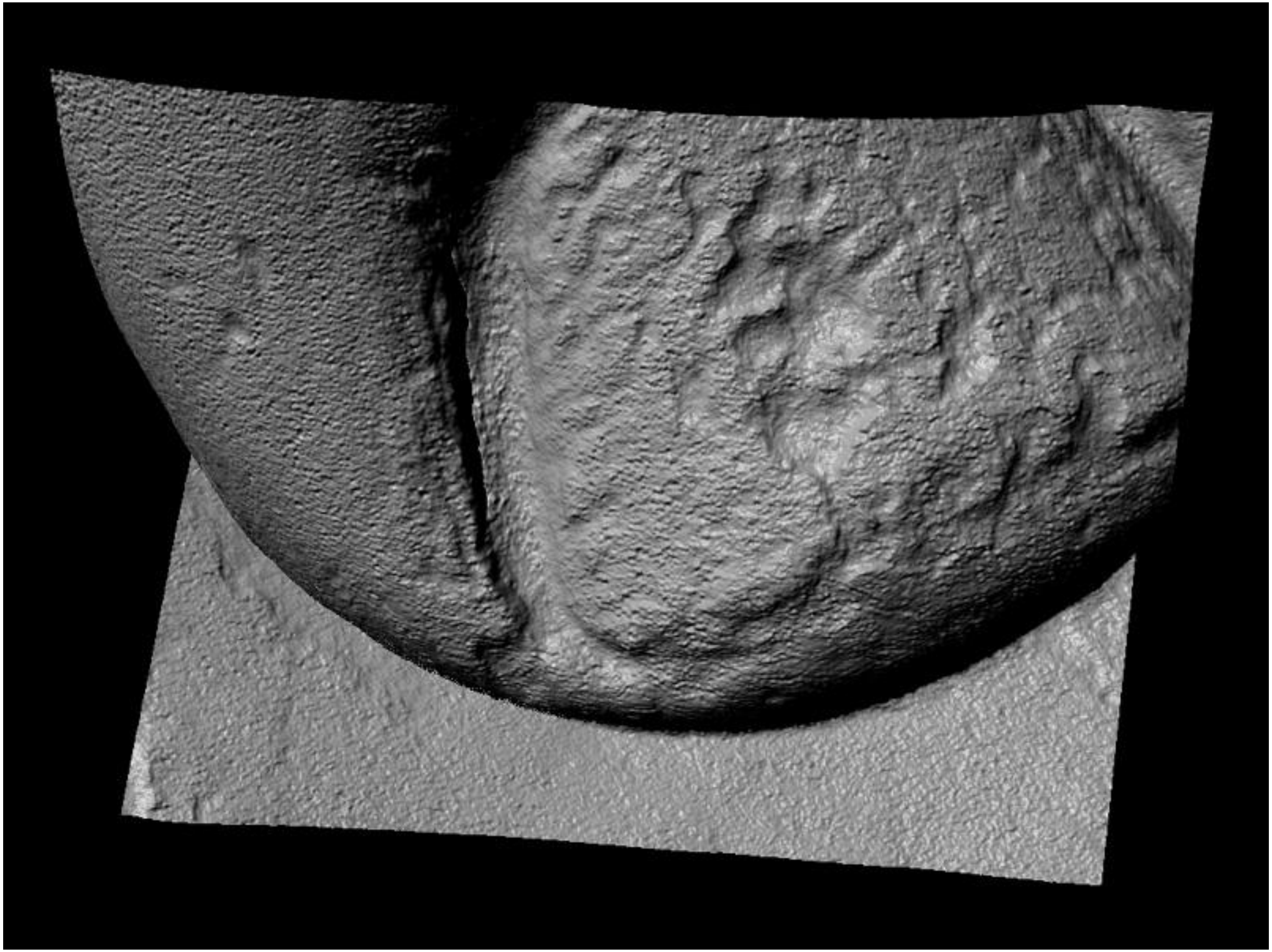
Mesh Simplification

COMP 557
Paul Kry







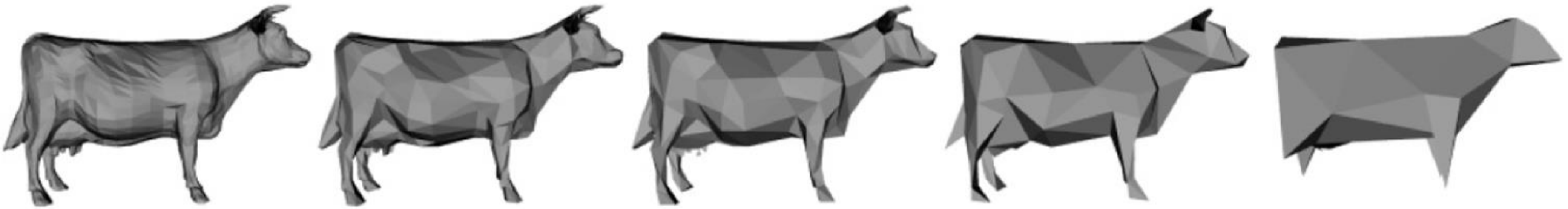
Level of Detail

- LOD (level of detail)
  - “loosely” the size of the polygons, e.g., length of shortest edge
- Resolution at which a model is displayed could be too coarse or too fine
  - aliasing problems
- David model has 1 billion polygons
- Another important example: terrain

Level of Detail

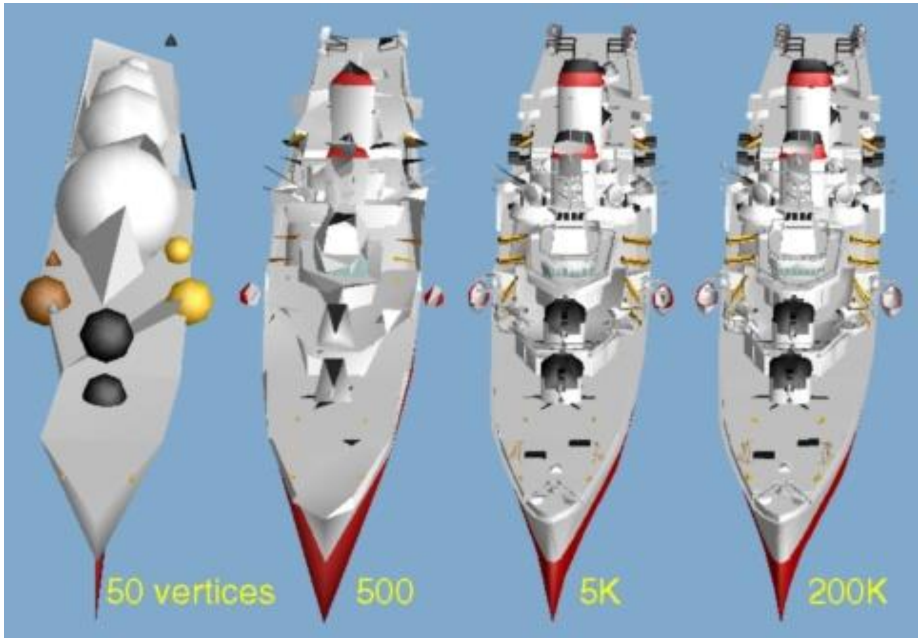
- Solution: compute several different coarse approximations
- How to choose which model to use?
  - # pixels use in screen space?
  - Continuous levels of detail (progressive meshes) to reduce popping when switching models
  - Viewpoint dependence?
  - Might need multiple levels simultaneously
    - e.g., terrain, coarse geometry for far, fine geometry for near.

Mesh Simplification



[Garland and Heckbert 1997]

- Reduce number of polygons
  - Faster rendering
  - Less storage
  - Simpler manipulation
- Find “good” approximation
  - Visual approximation
  - Geometric approximation
  - Data approximation
- Other desirable qualities
  - Applicability (works on all meshes?)
  - Efficiency, Scalability,
  - Preservation of attributes (texture coordinates, normals, etc.)



[Hoppe 1996]

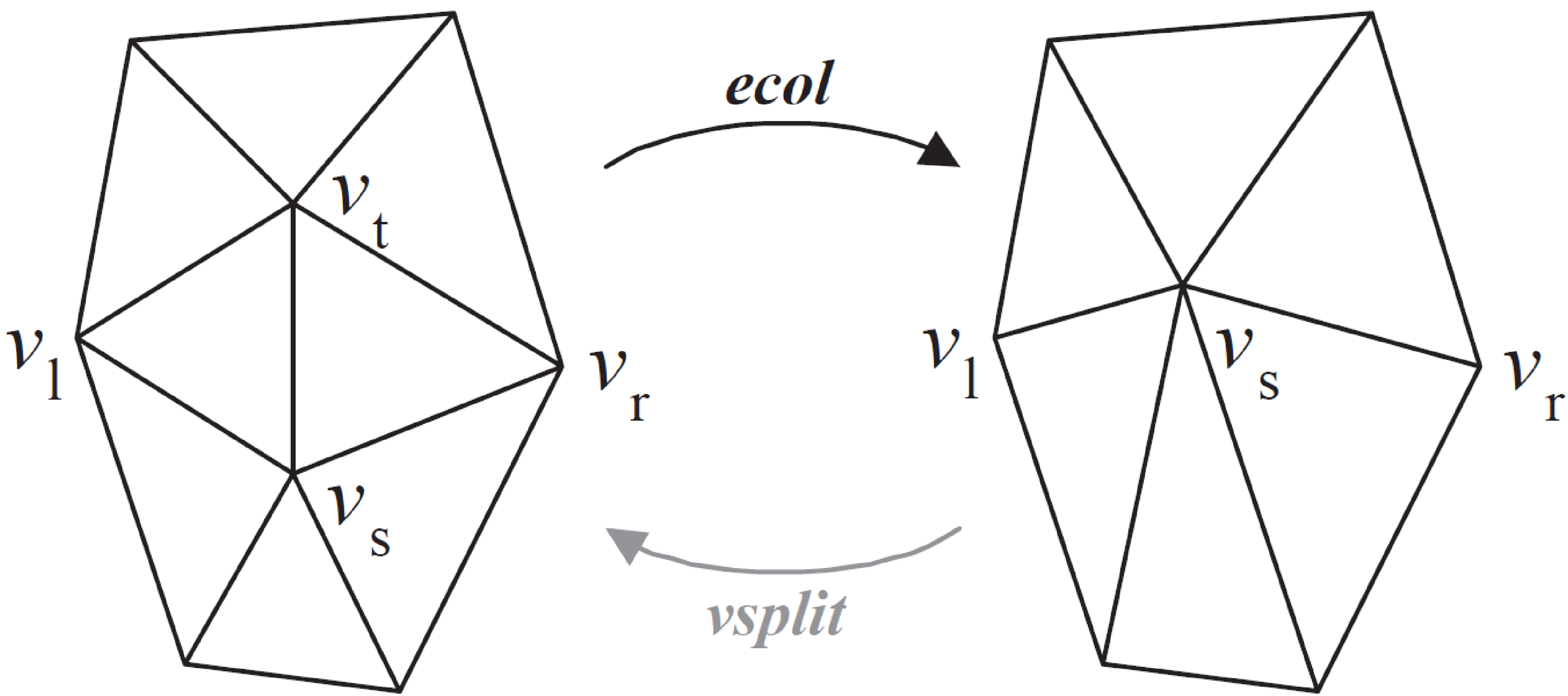
Data Sources

- Measurement
  - Models from laser range finder
  - Iso-surface generation from 3D MRI or CT
  - Terrain from Satellite, Radar, Sonar

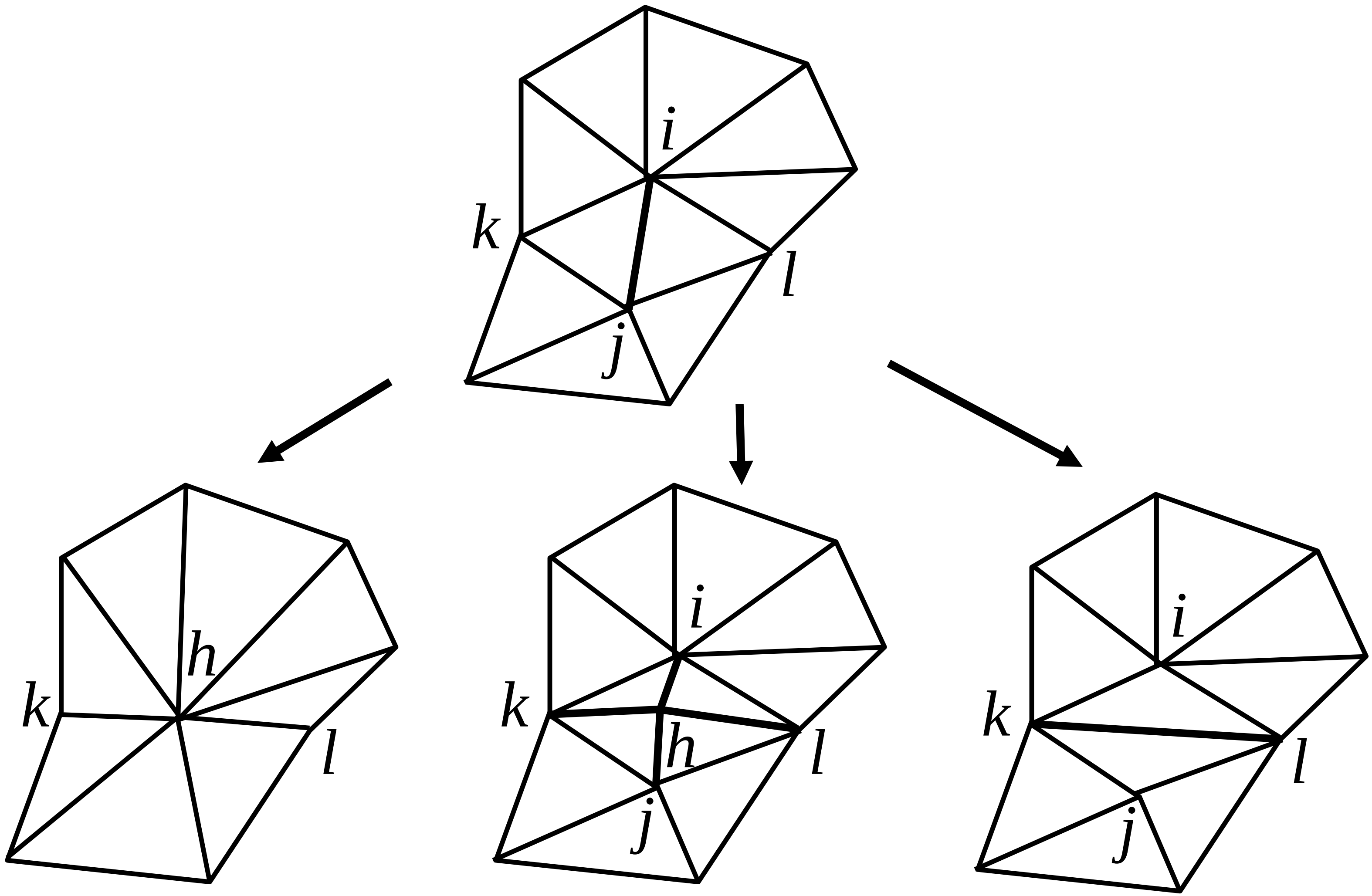
Simplification Approaches

- Geometry refinement
  - Adaptive subdivision
- Geometry resampling
  - Mesh re-tiling
  - Variational Shape Approximation
    - Find a set of geometric proxies that fit the data
- Geometry decimation
  - Vertex decimation
    - remove vertices in planar regions and fill hole with triangles
  - Edge contraction
  - Vertex merging

Local Modification (invertible)



Other Local Modifications



edge collapse
edge split
edge swap

Global Modifications

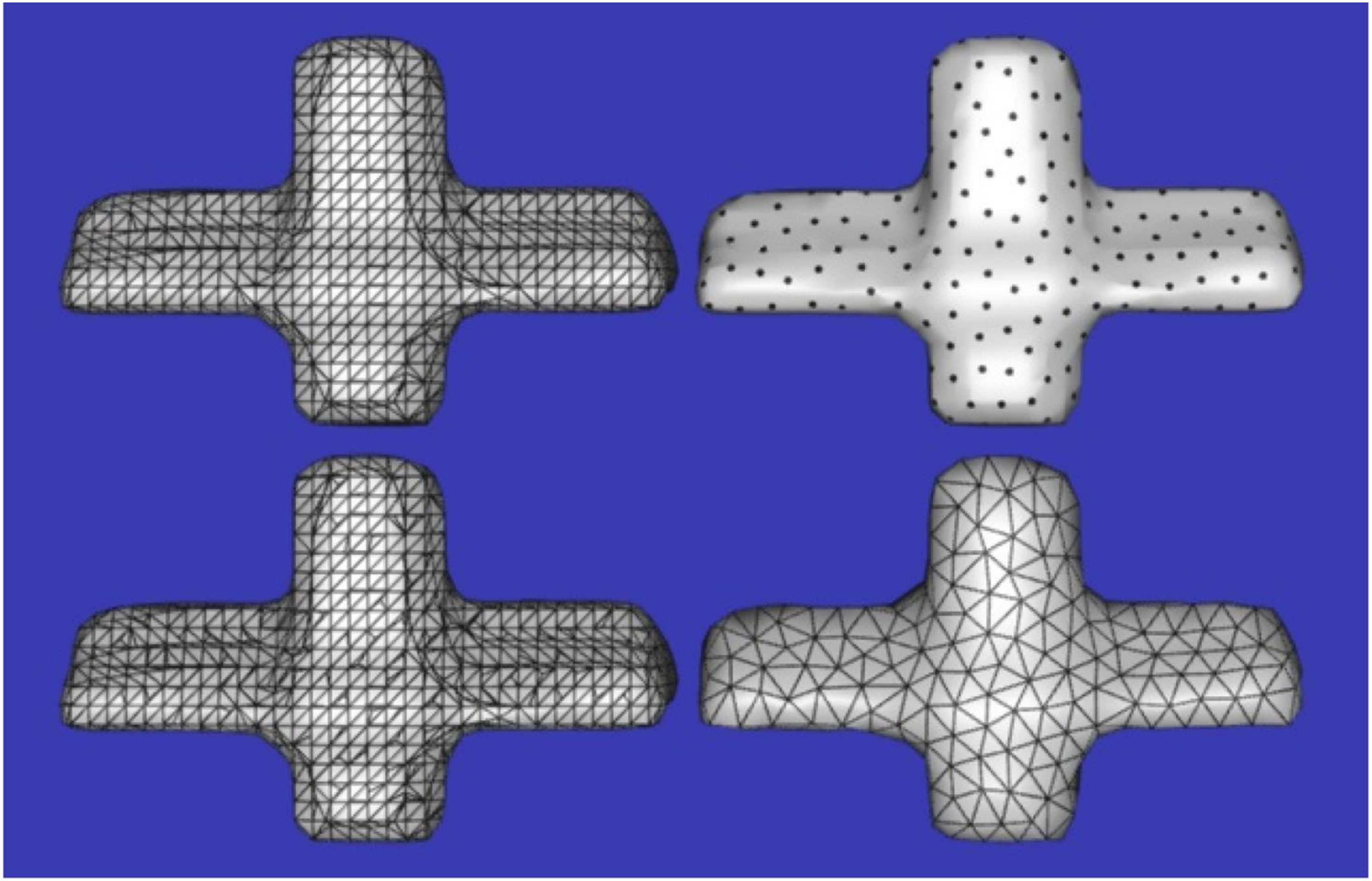
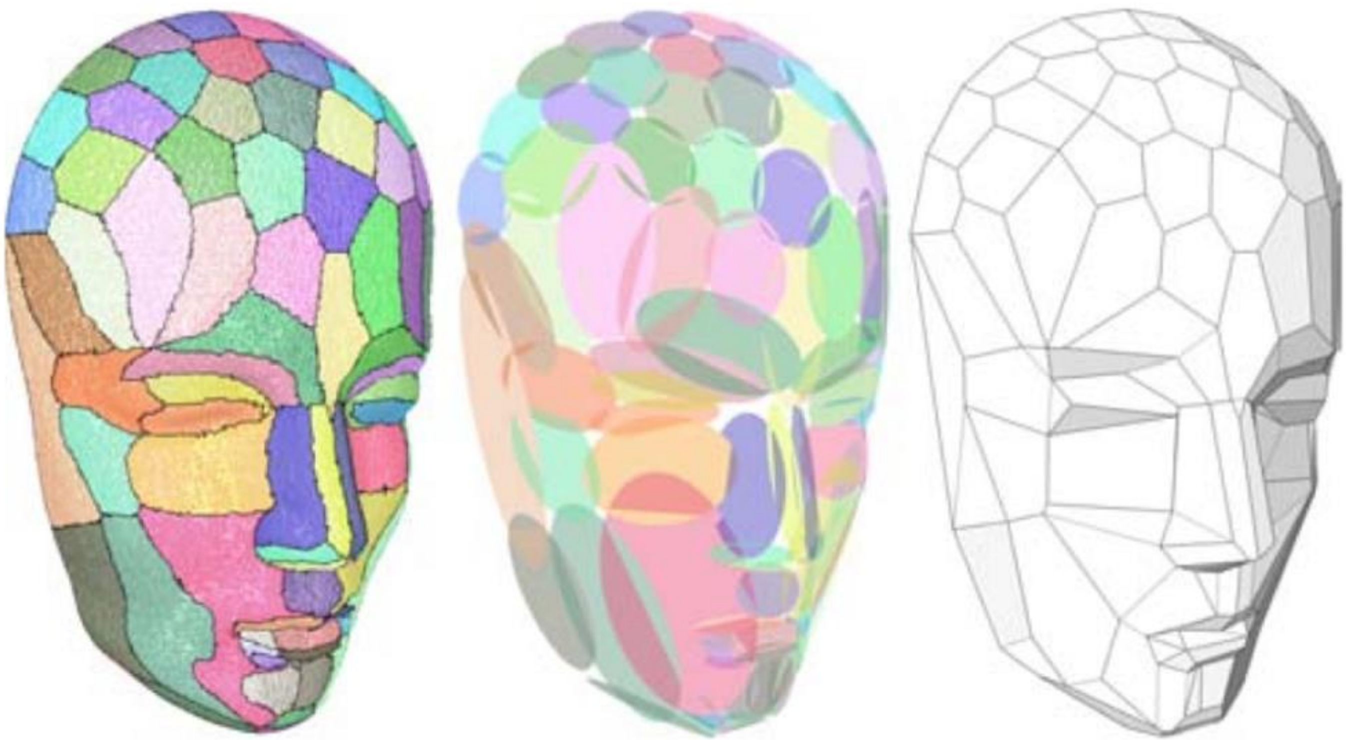


Figure 1: Re-tiling of a radiation iso-dose surface. Upper left: Original surface. Upper right: Candidate vertices after point-repulsion. Lower left: Mutual tessellation. Lower right: Final tessellation.

Global Modificaitons



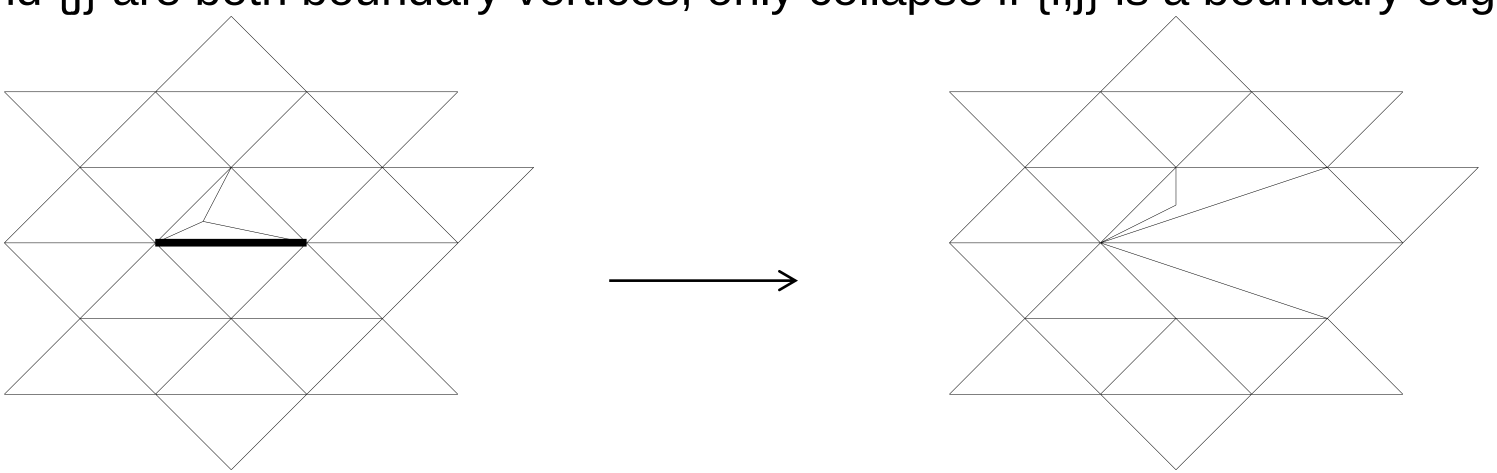
Variational shape approximation from [Cohen-Steiner et al. 2004].

Characteristics

- Speed vs quality?
- Type of Mesh
  - Height field or parametric
  - Manifold
  - Polygon soup
- Modifies topology?
- Continuous LOD?
- View-Dependent refinement?
- Simplify topology?
  - David head with genus 340?

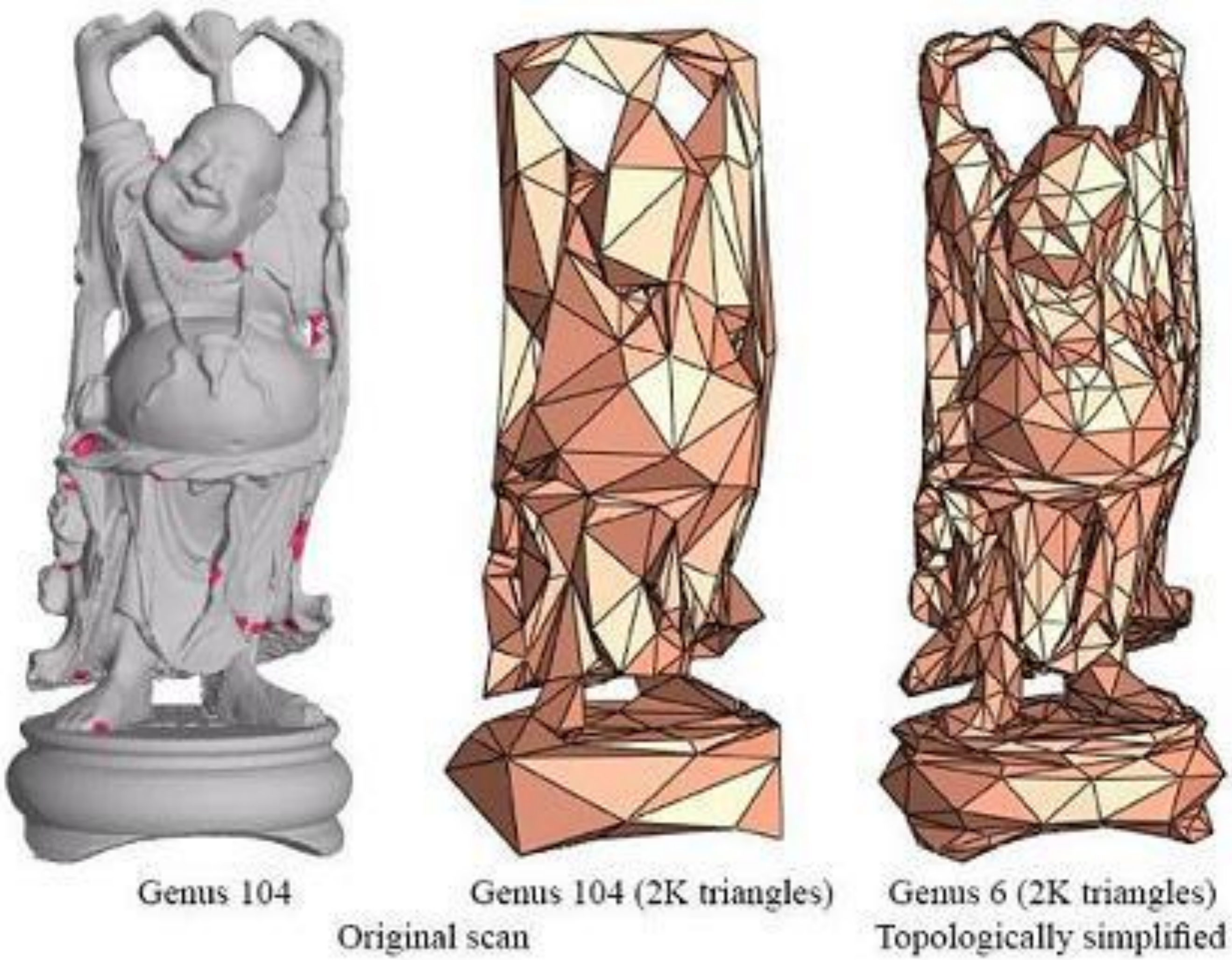
Edge Collapse Problems

- Preserve topology? Preserve the manifold?
- Avoid creating self-intersections in geometry
- Avoid creating non-manifold topology (can use heuristics)
  - Number of common adjacent vertices to collapsing edge should be 2
  - If {i} and {j} are both boundary vertices, only collapse if {i,j} is a boundary edge



Fixing Topology

- Preserving topology during simplification is not always a good idea

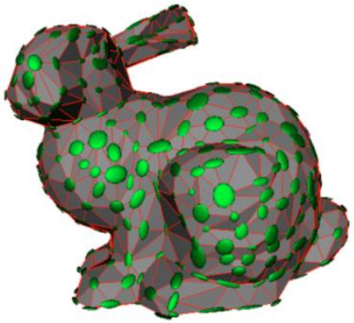


Quadric Error Metric

- Each face defines a plane
- Each vertex of a face lies in the planes of its adjacent faces
- Consider moving a vertex to position v
  - How well does vertex v lie in a set of planes? $p=[a\ b\ c\ d]^T$

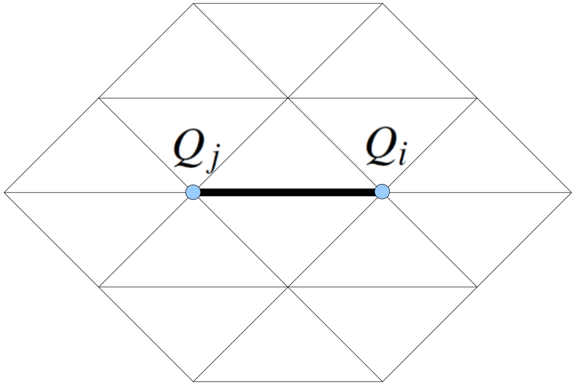
$$
\begin{aligned}
\Delta(\mathbf{v}) &= \sum_{\mathbf{p}\in \text{planes}(\mathbf{v})} (\mathbf{v}^{\mathsf{T}}\mathbf{p})(\mathbf{p}^{\mathsf{T}}\mathbf{v}) \\
&= \sum_{\mathbf{p}\in \text{planes}(\mathbf{v})} \mathbf{v}^{\mathsf{T}}(\mathbf{p}\mathbf{p}^{\mathsf{T}})\mathbf{v} \\
&= \mathbf{v}^{\mathsf{T}}\left(\sum_{\mathbf{p}\in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

Distance squared from v to plane $||p^T v||^2$



Cost of collapsing an edge is related to error for best choice of position v for both ends.
(we will want to regularize this)

$$e = \min_{v} v^T (Q_i + Q_j) v$$



Quadratic Error Metric Implementation Issues

- Let us discuss the following issues
  - Regularization of the minimization problem using distance squared to point half way along an edge
  - Computation of the error for each edge by solving the minimum of the quadratic equation (i.e., take the derivative and solve for the zero)
  - The use of a priority queue to keep track of which edge should be collapsed next
    - Each collapse requires a number of adjacent edges to be removed and their error recomputed

Quadratic Error Metric Mesh Simplification Steps

- Compute planes equation for each face, **p**
- Compute quadratic function coefficient for each face, **K**
- Compute quadratic function coefficient for each vertex, **Q**
- Solve optimal vertex position and error for each edge (i,j) using **Q**i + **Q**j and possibly including a regularization term.
- Insert all edge errors into a priority queue, with the minimum error at the top of the queue
- Pop top off queue until we find an edge collapse that does not cause problems (avoid bad topology, perhaps check geometry)
- Collapse the edge to optimal vertex position, set quadratic function coefficient of this vertex as **Q**i + **Q**j,
- Remove adjacent edges from queue, re-compute edge collapse error, and reinsert into queue
- Repeat until desired level of detail is reached.

More information

- Surface simplification using quadric error metrics
  - Garland and Heckbert, 1997
  - http://dl.acm.org/citation.cfm?id=258849
- Progressive meshes
  - Hoppe, 1996
  - http://research.microsoft.com/en-us/um/people/hoppe/proj/pm/