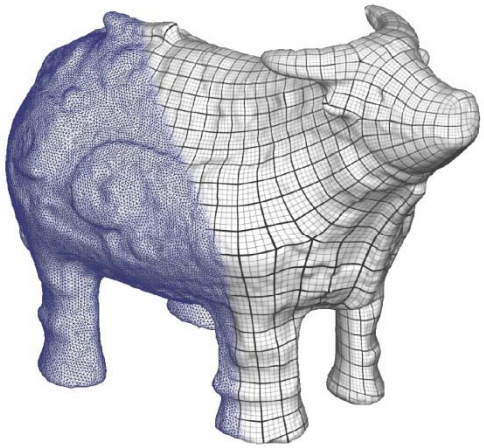
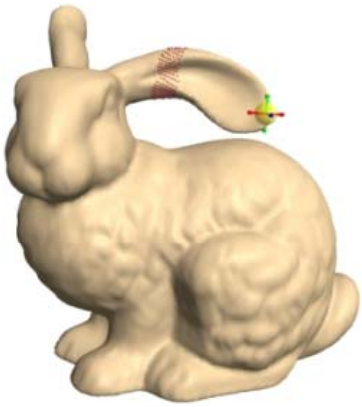
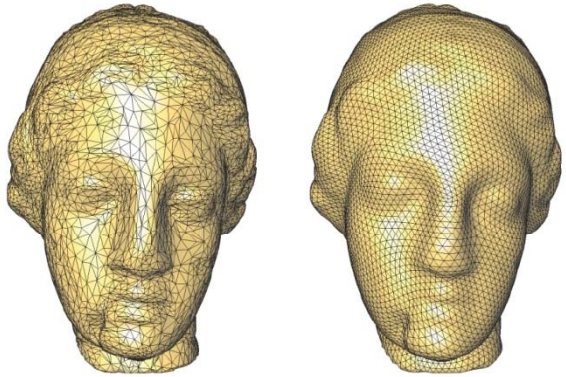
Geometry Processing Algorithms

CS468

http://cs468.stanford.edu/







Objective

- Theory and algorithms for efficient analysis and manipulation of complex 3D models

- Hands-on experience

Requirements

Prerequisites:

- Introduction to Computer Graphics
- Experience with C++ programming
- Background in geometry or computational geometry helpful, but not necessary.

Grade (3 units):

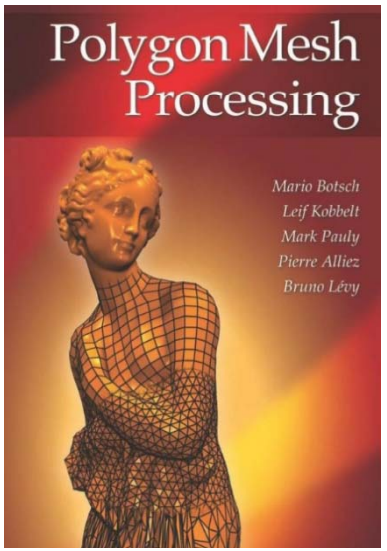
- Programming exercises
  - OpenMesh intro (10%)
  - Surface smoothing (20%)
  - Simplification (20%)
  - Parameterization (25%)
  - Remeshing (25%)

Work in pairs. Use *OpenMesh* API

References

- Book

  “Polygon Mesh Processing”
  by Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, Bruno Levy



- Eurographics 2008 course notes

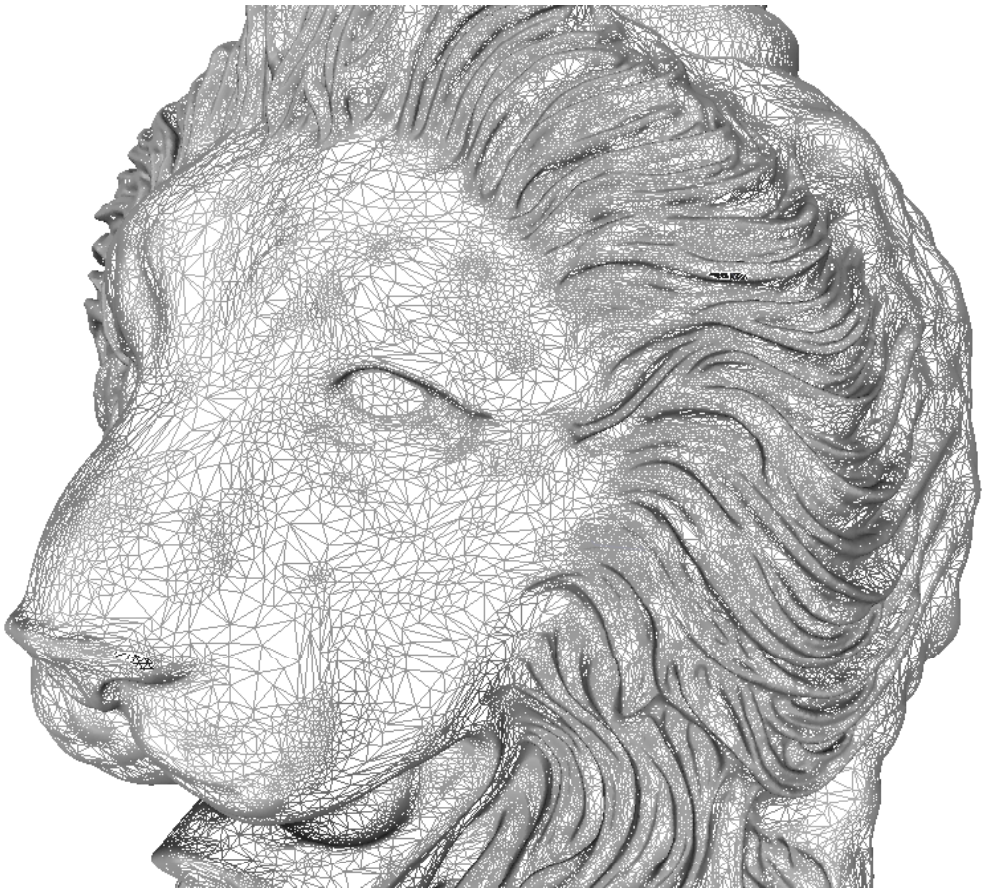
  “Geometric Modeling Based on Polygonal Meshes”

  by Mario Botsch, Mark Pauly, Leif Kobbelt, Pierre Alliez, Bruno Levy, Stephan Bischoff, Christian Rössl

- More links on web site

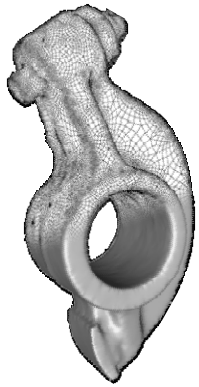
What is Geometry Processing About?

- Acquiring
- Analyzing
- Manipulating



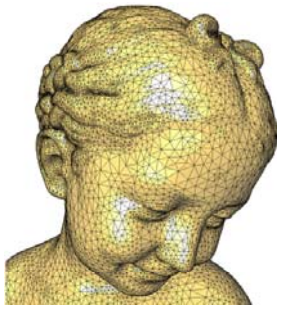
Applications

Medical

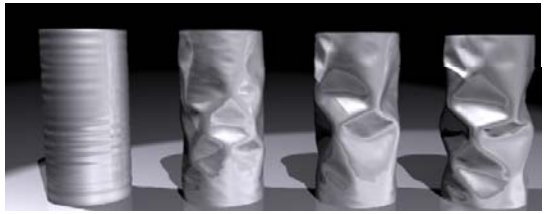


Engineering

E-Commerce



Culture



Simulation

Games & Movies

Architecture

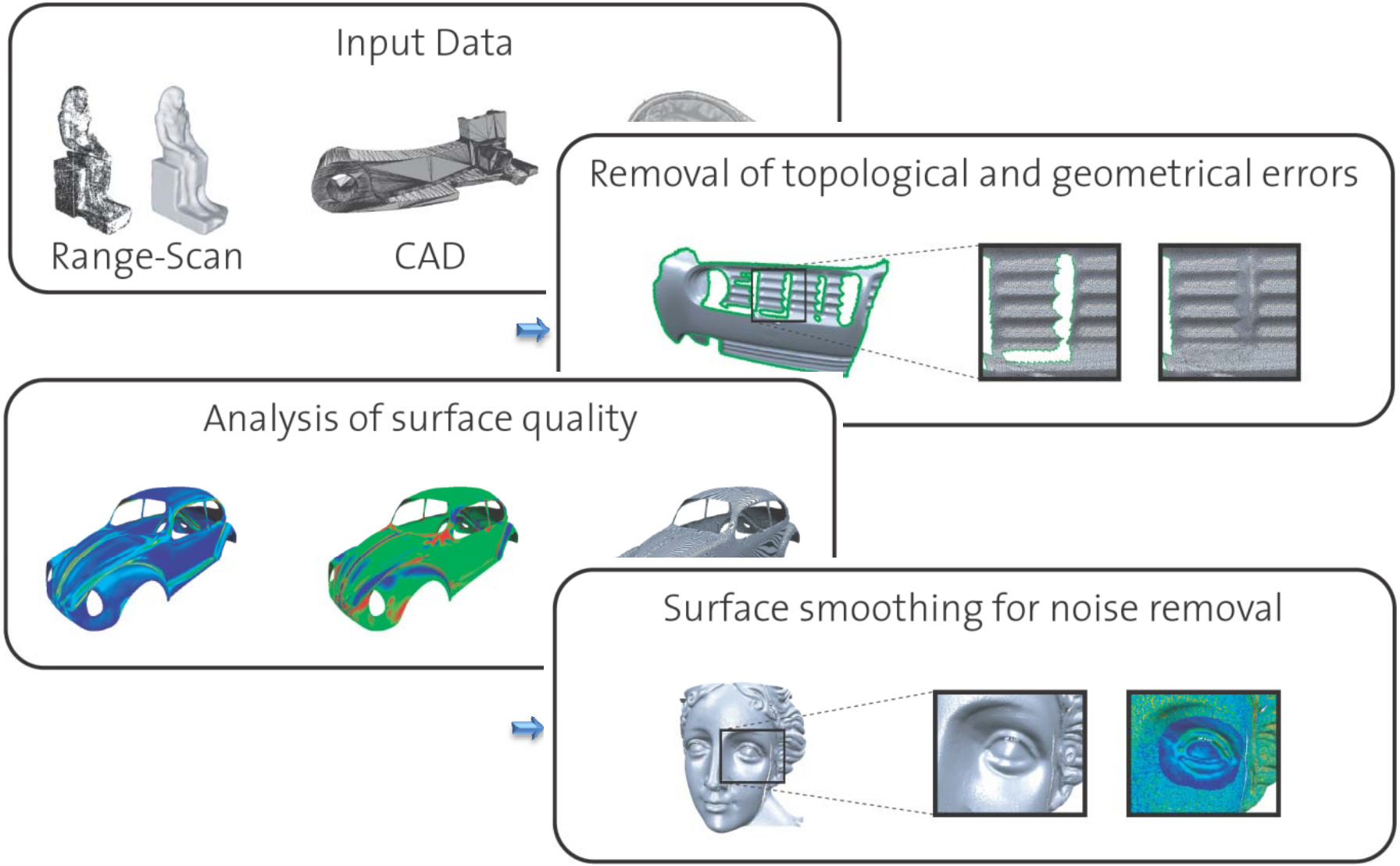
Creating

Architecture

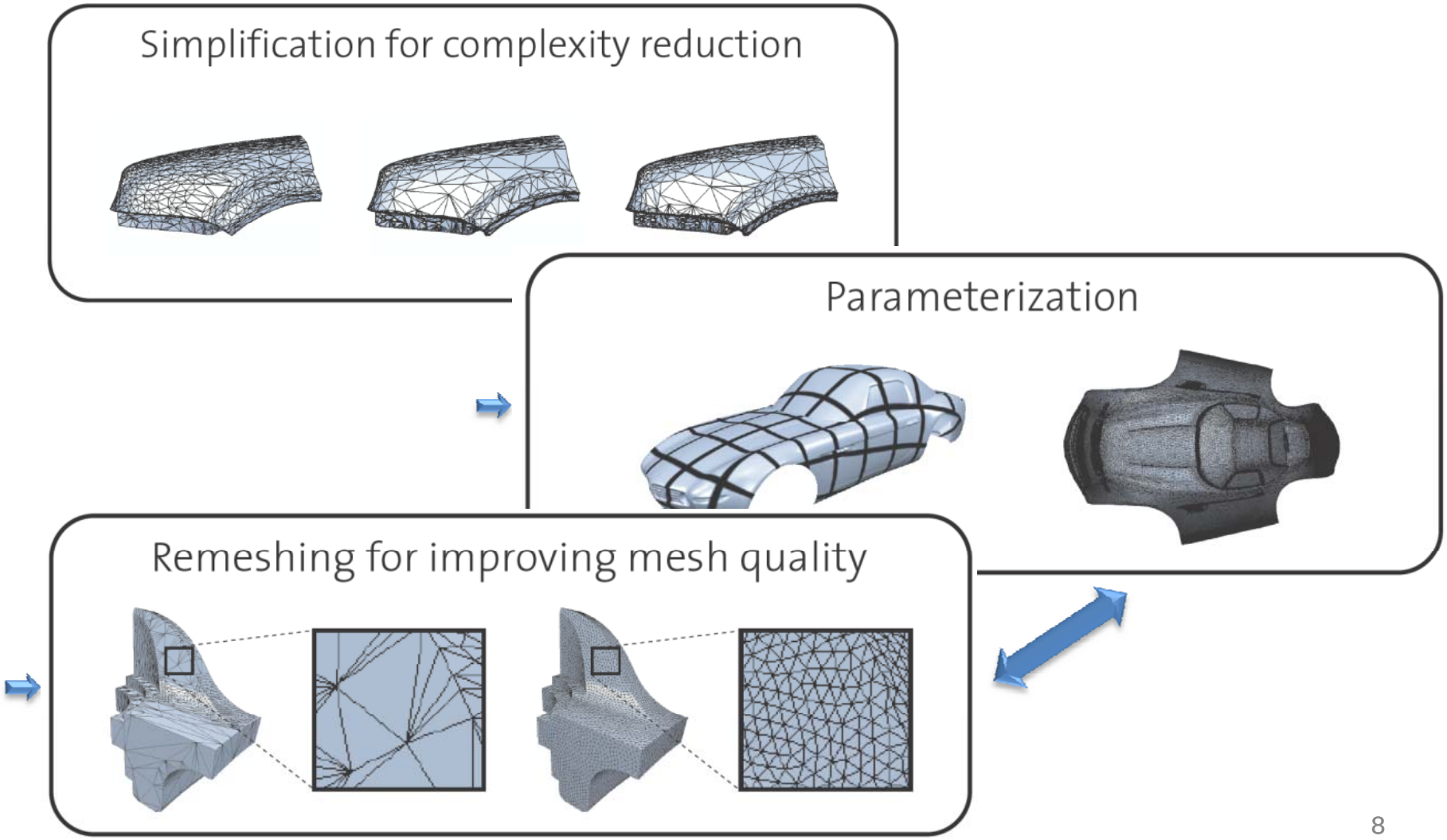
Reverse Engineering

A Geometry Processing Pipeline

Low Level Algorithms



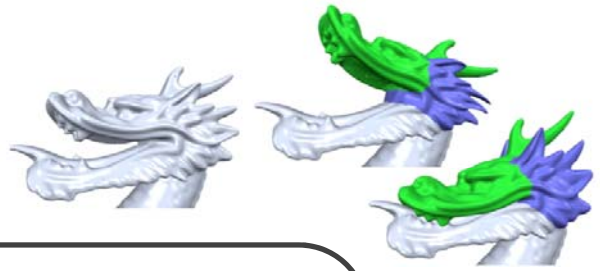
A Geometry Processing Pipeline



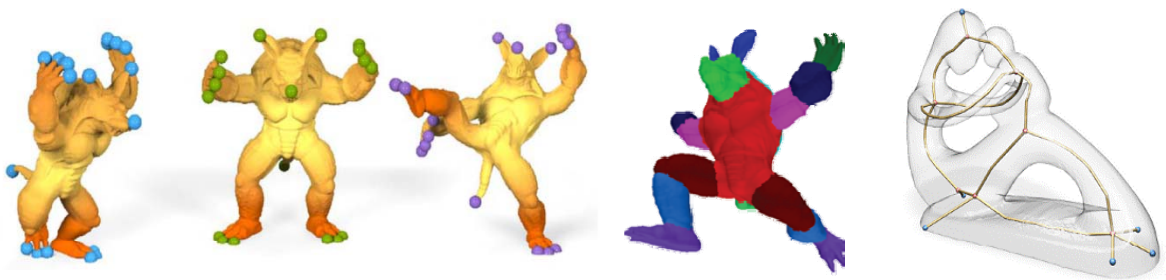
A Geometry Processing Pipeline
High Level Algorithms

Freeform and multiresolution modeling

Deformation and editing

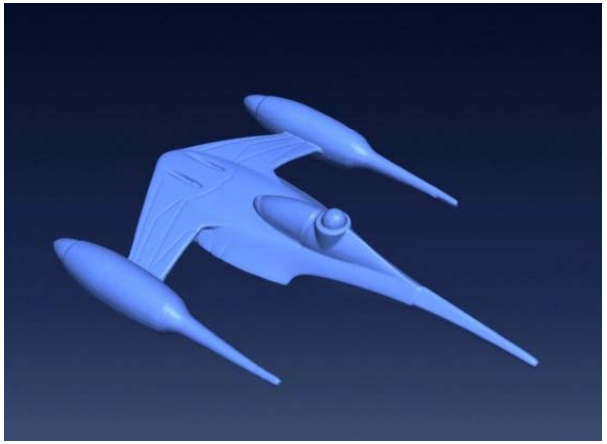


Extracting shape structure



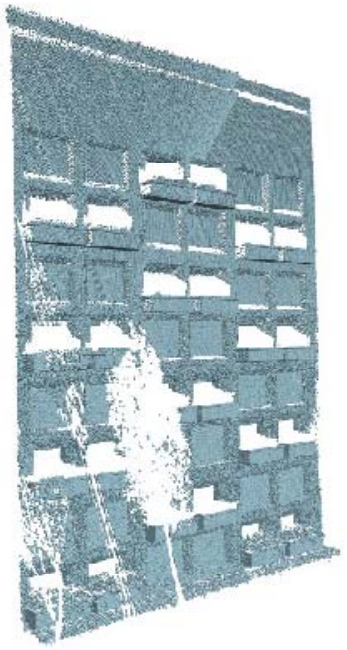
Acquiring 3D Geometry

Range Scanners



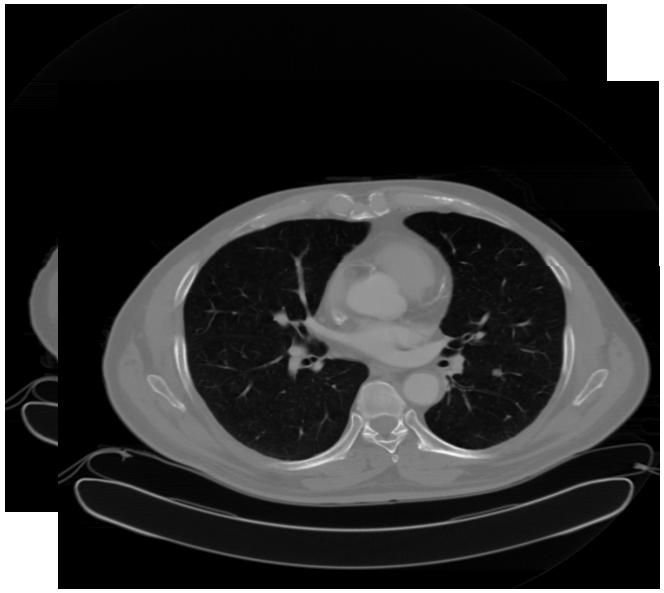
Acquiring 3D Geometry

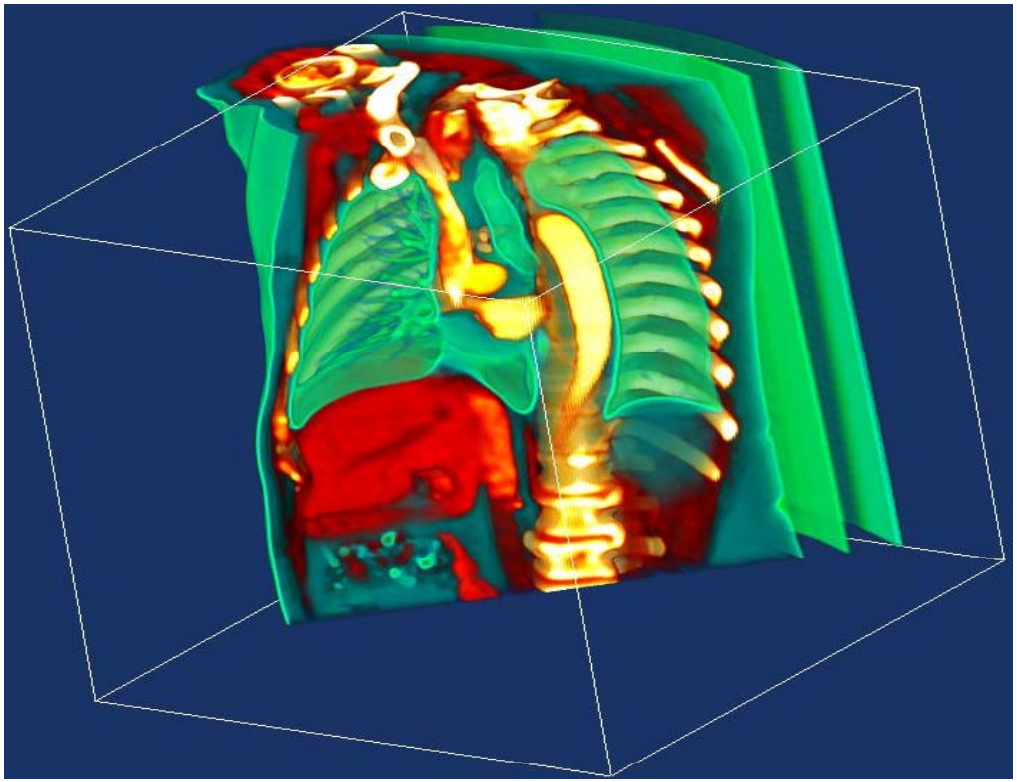
Range Scanners



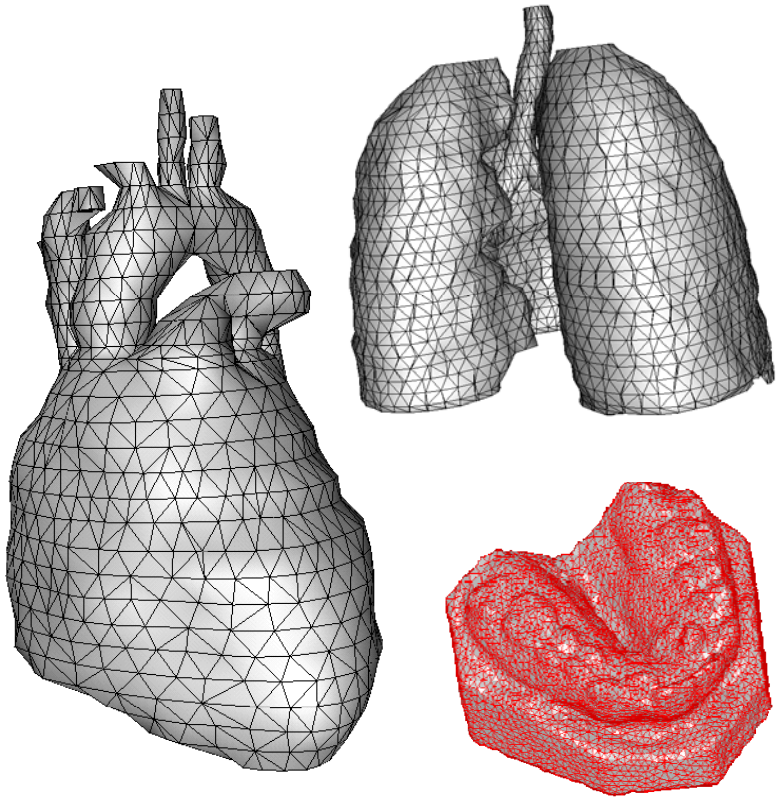
Acquiring 3D Geometry

Tomography





Mesh Construction



from contours

from point clouds

Simplification

20,000

8,000

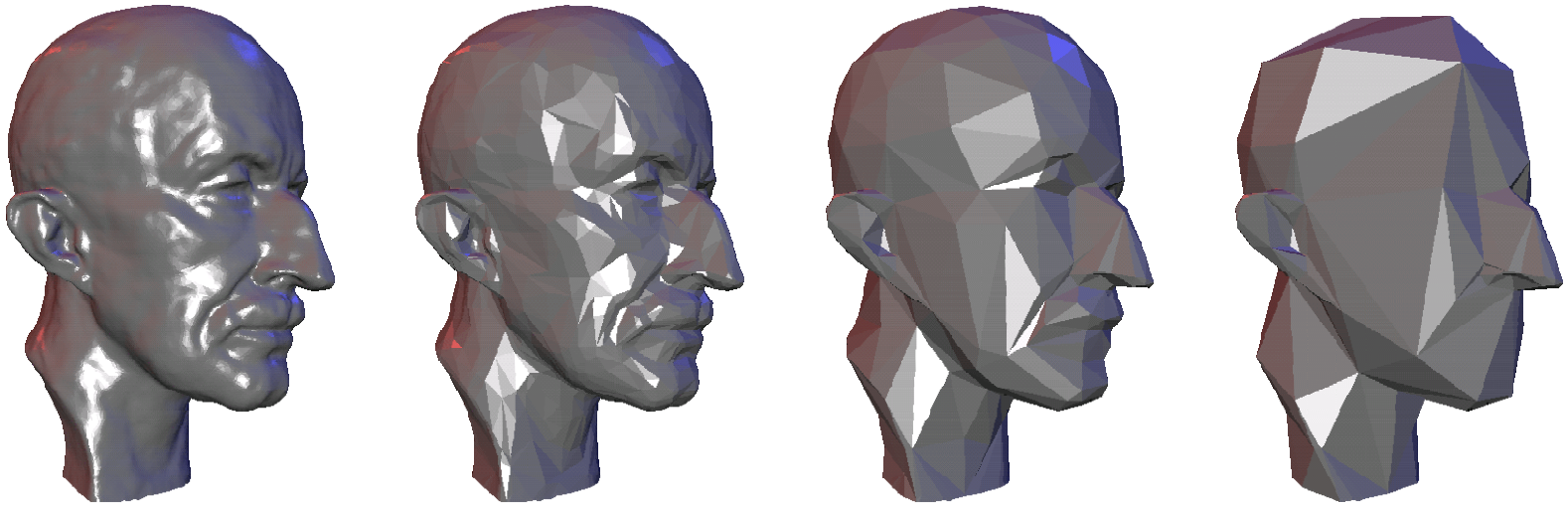
2,000

Demo

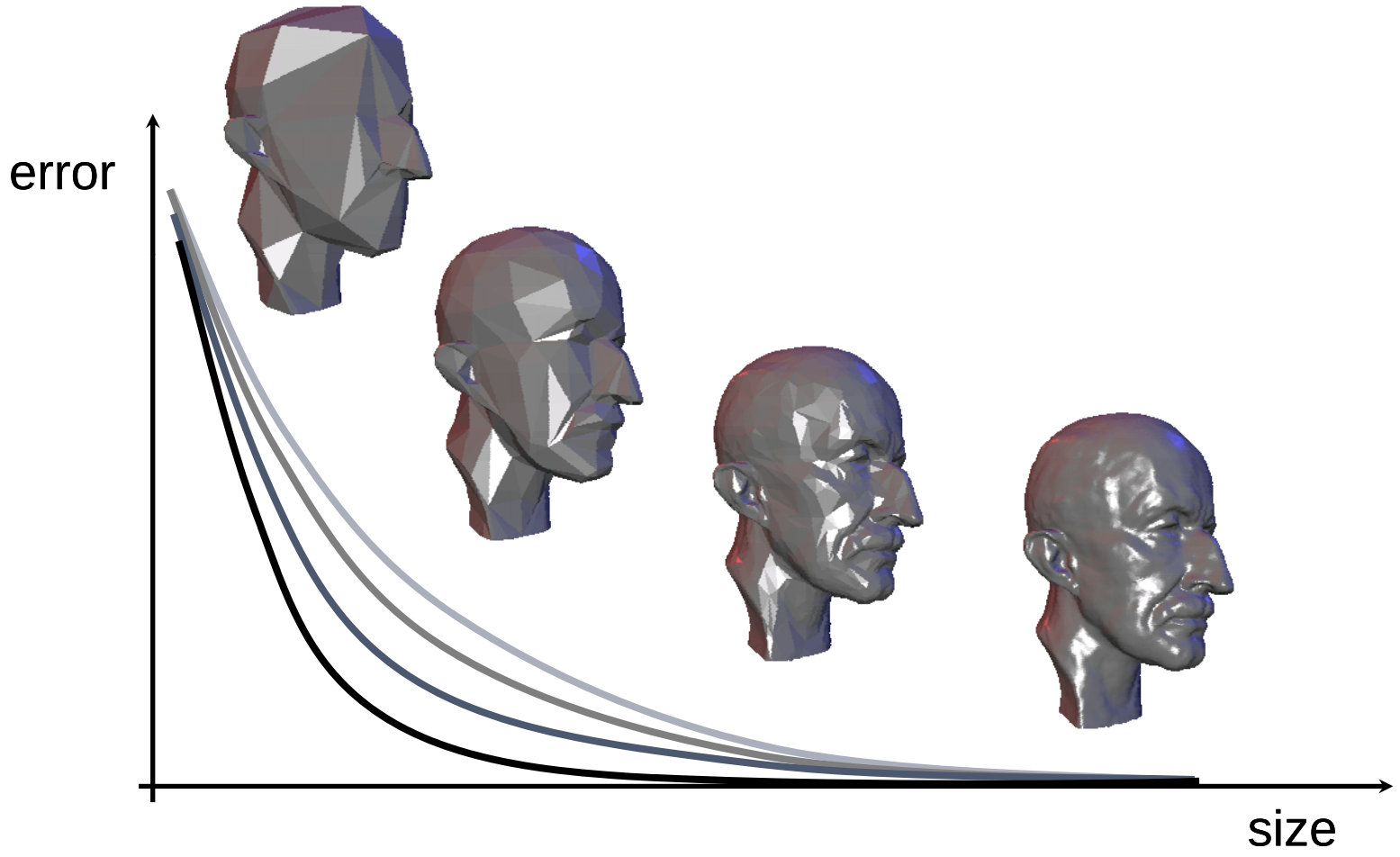
Applications

Multi-resolution hierarchies for

- efficient geometry processing
- level-of-detail (LOD) rendering



Size-Quality Tradeoff



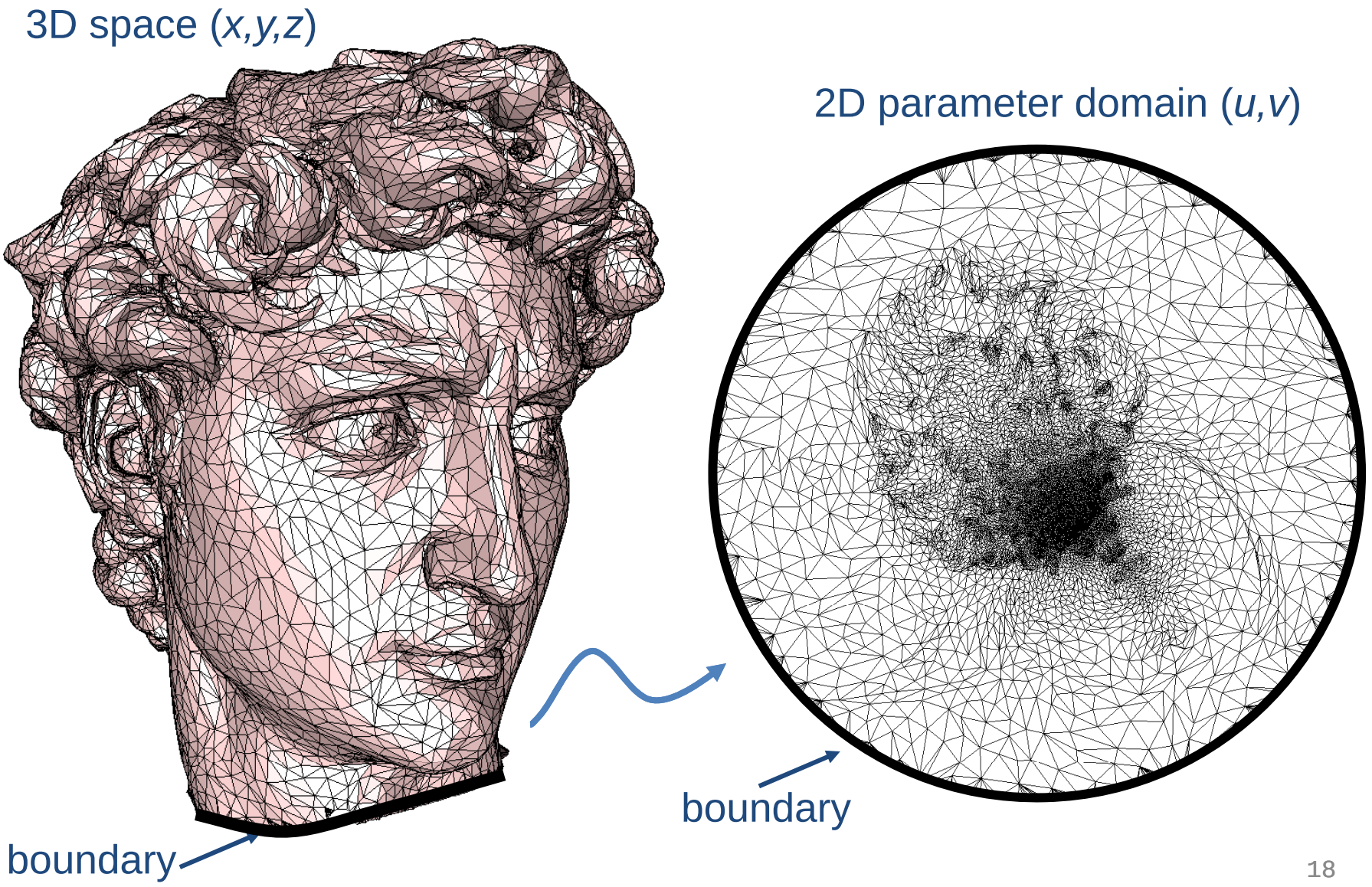
Compression

Surface Geometry Connectivity

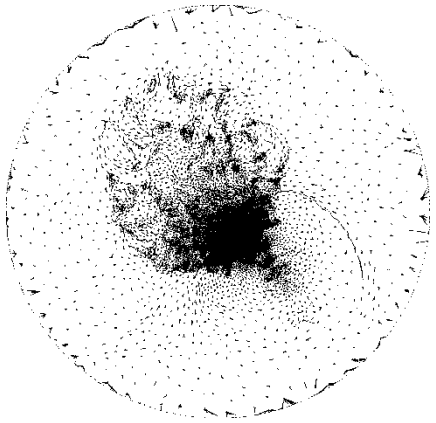
VRML = 200K, zipped VRML = 70K, compressed = 15K

demo

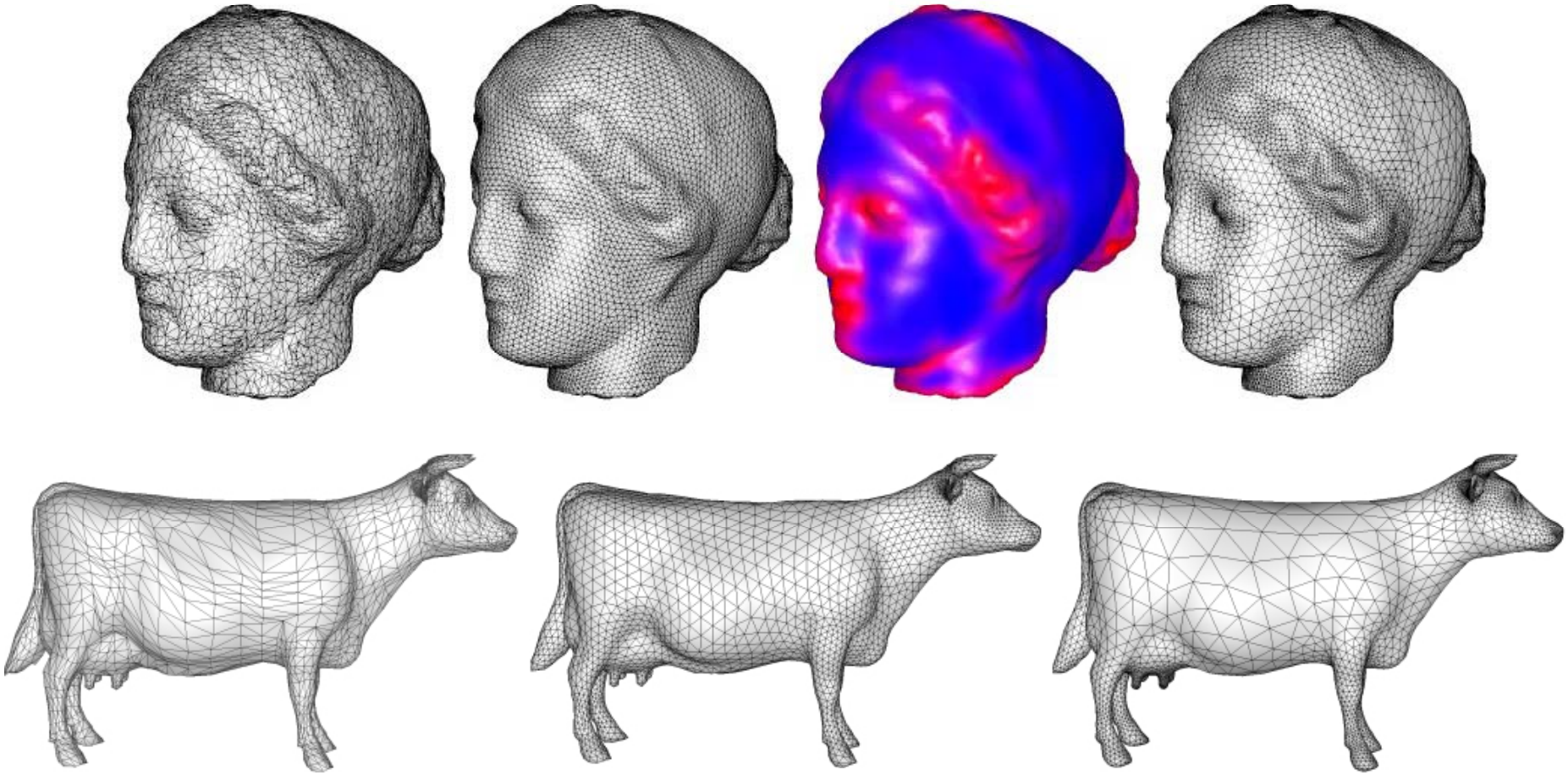
Parameterization



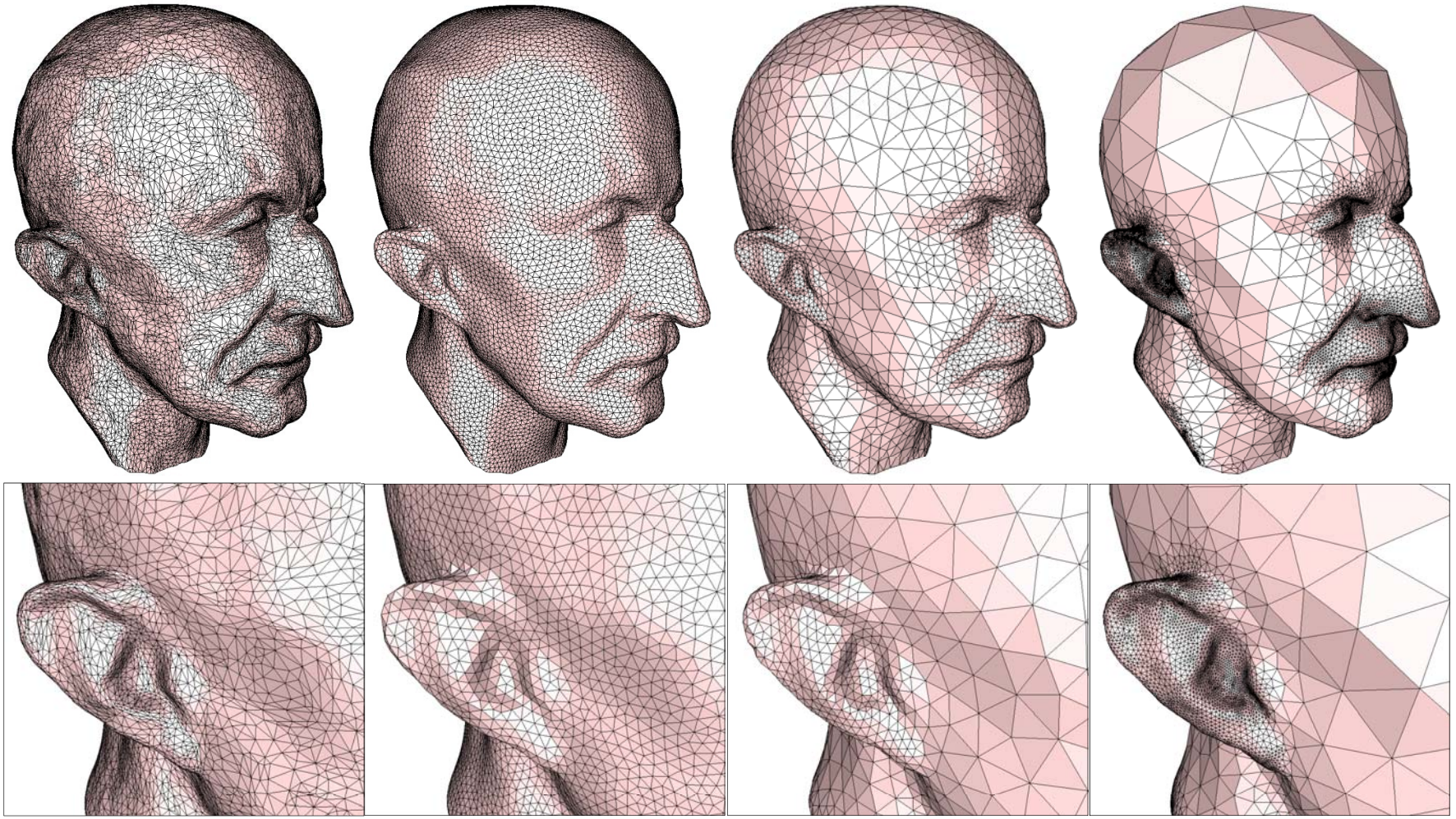
Application - Texture Mapping



Remeshing

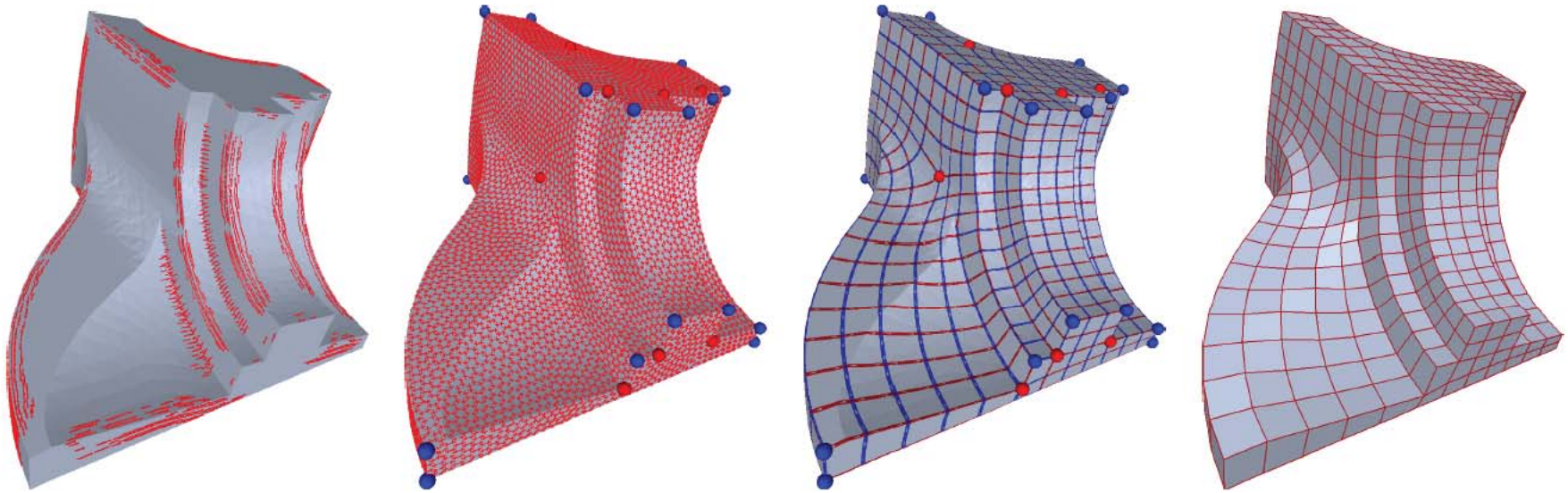


More Remeshing

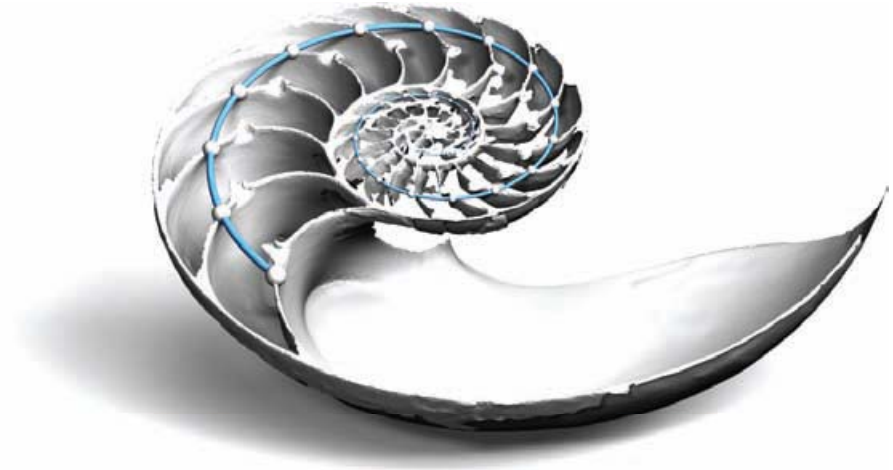


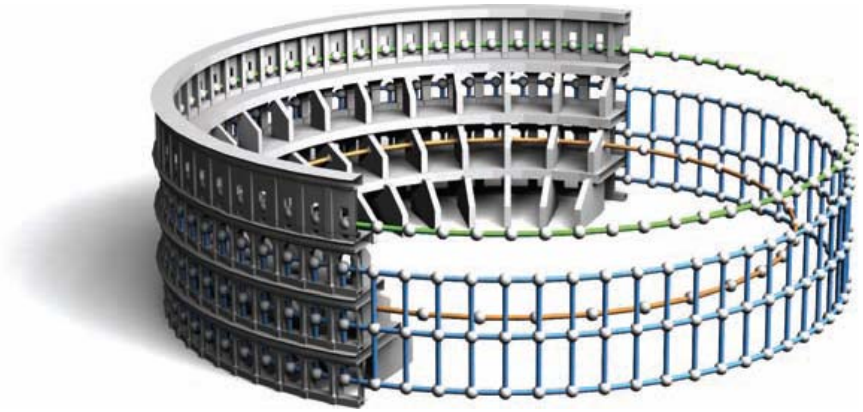
demo

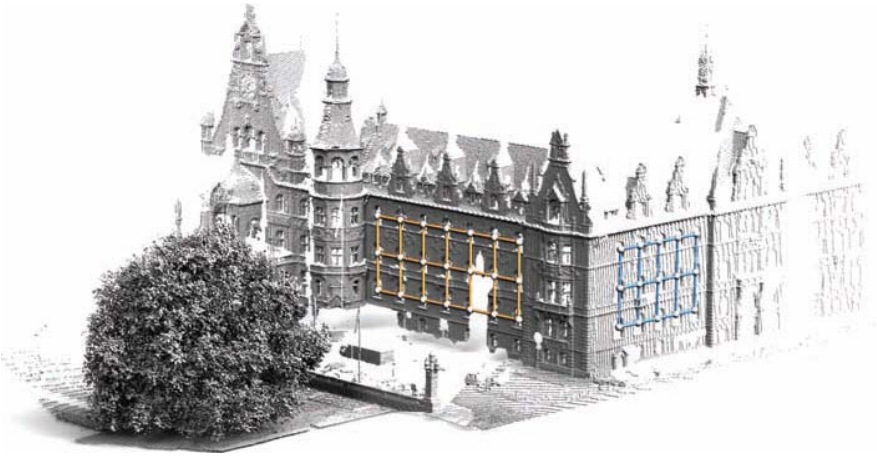
Quad Remeshing



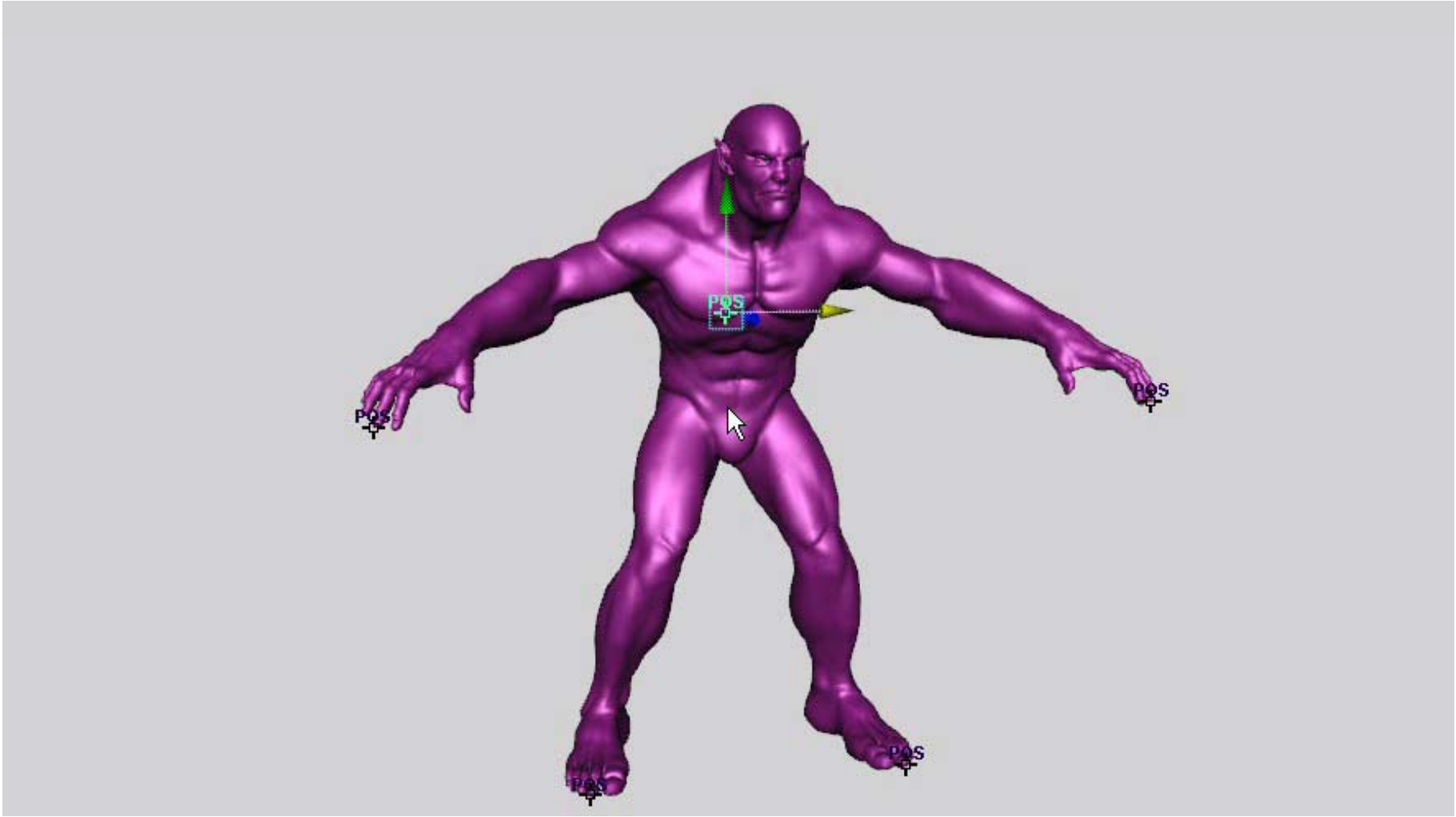
Symmetry Detection



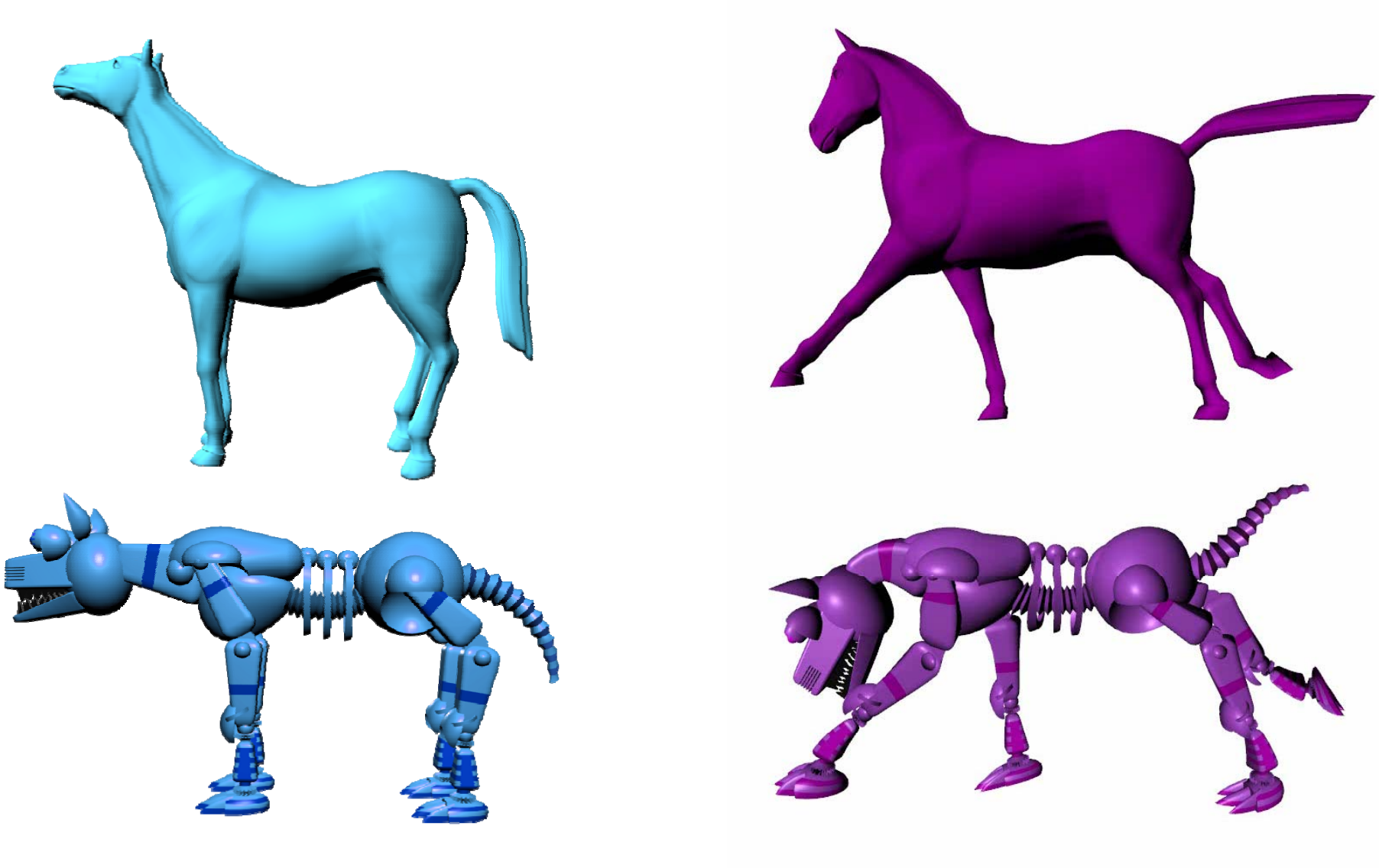




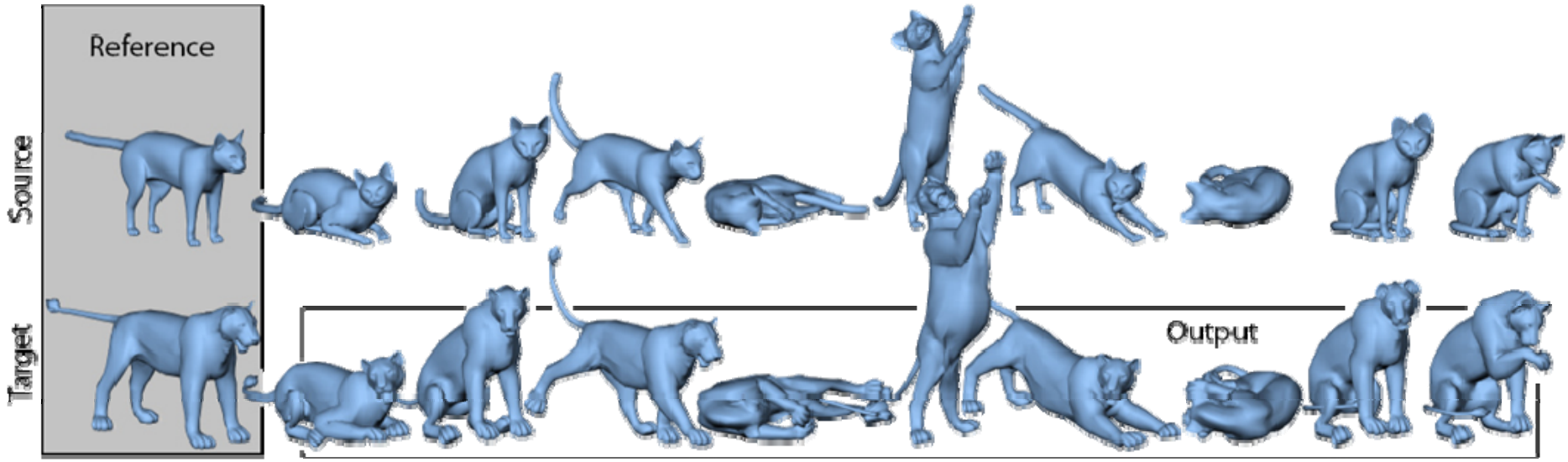
Deformation

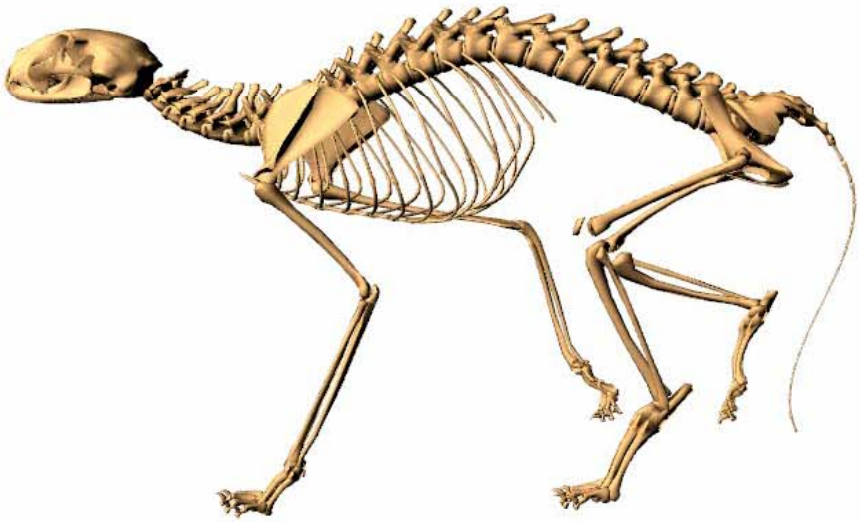


Deformation Transfer

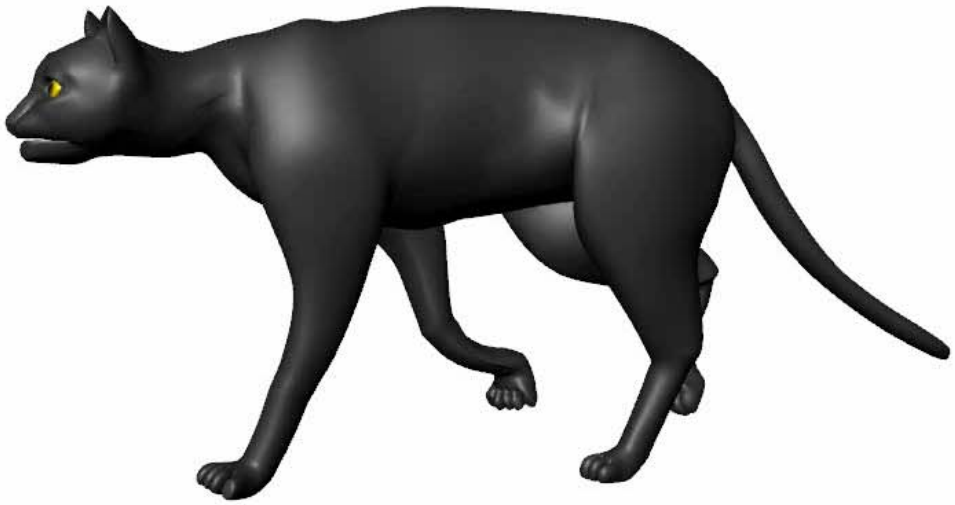


Deformation Transfer

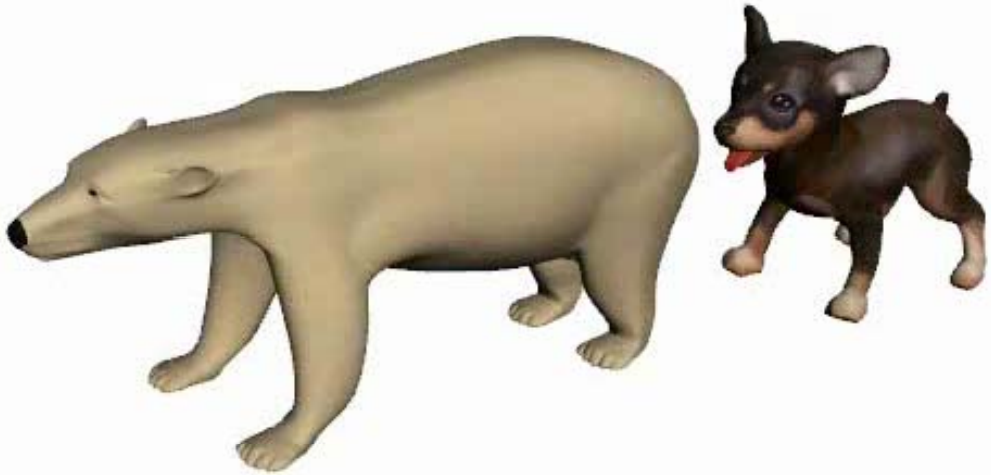




Target



Source



What's Next?

- Learn about geometry data structures
- Go into detail of a few representative algorithms
- Code them up!
- Have fun with geometry ☺