3D Object Representation

Connelly Barnes

CS 4810: Graphics

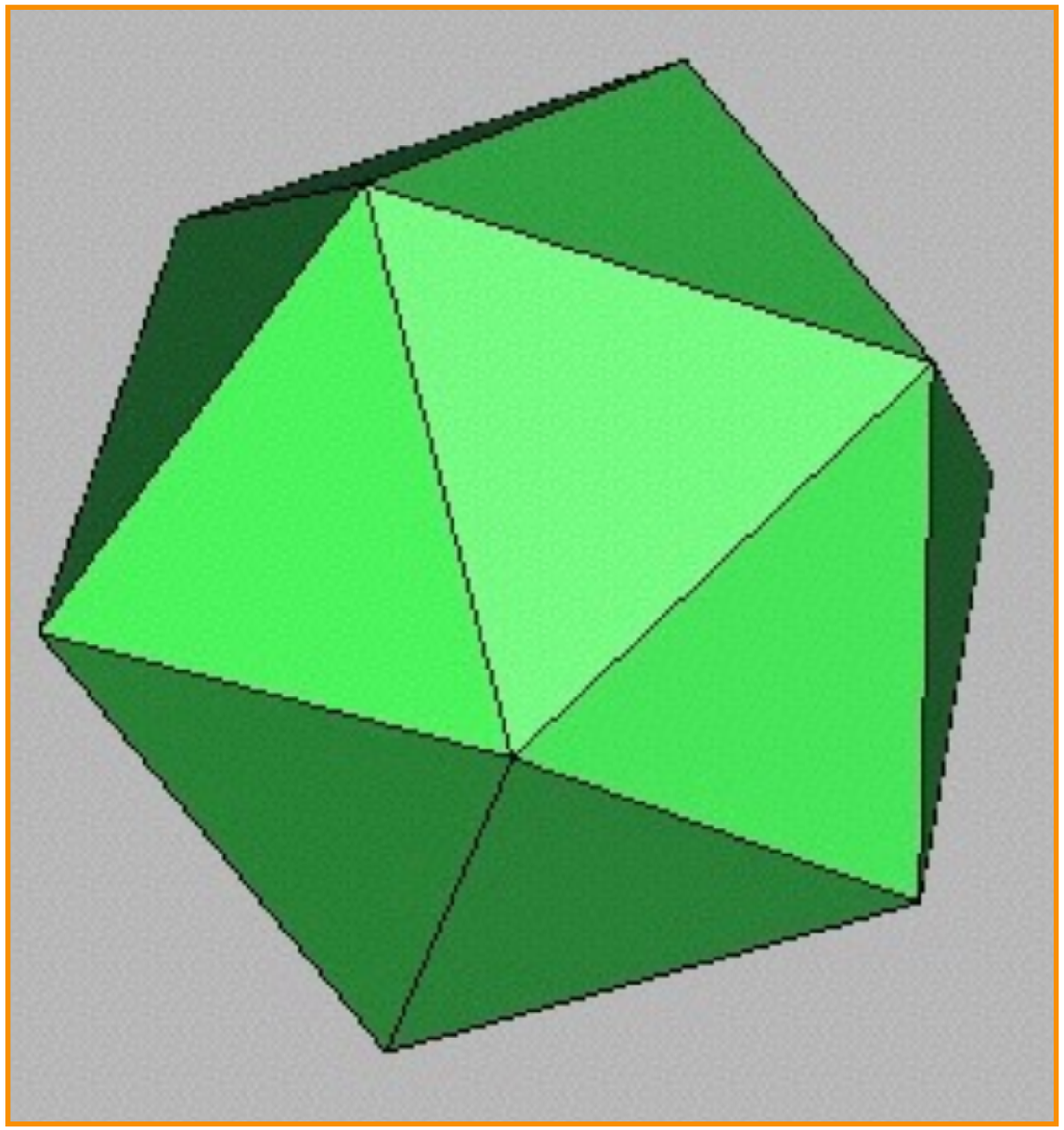
Acknowledgment: slides by Jason Lawrence, Misha Kazhdan, Allison Klein, Tom Funkhouser, Adam Finkelstein and David Dobkin

3D Object Representation

- How do we ...
  - Represent 3D objects in a computer?
  - Construct such representations quickly and/or automatically with a computer?
  - Manipulate 3D objects with a computer?

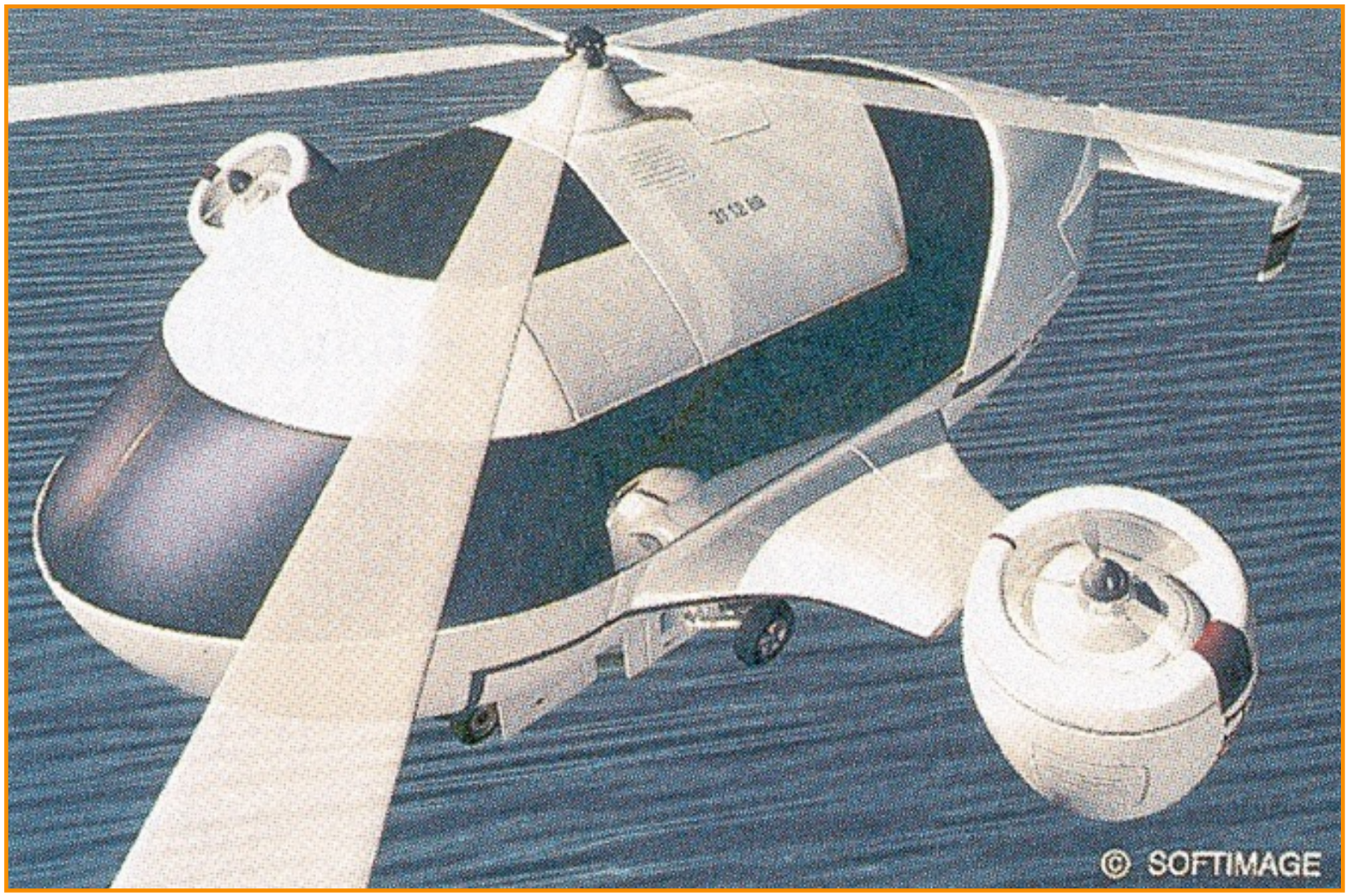
Different methods for different object representations

3D Objects



How can this object be represented in a computer?

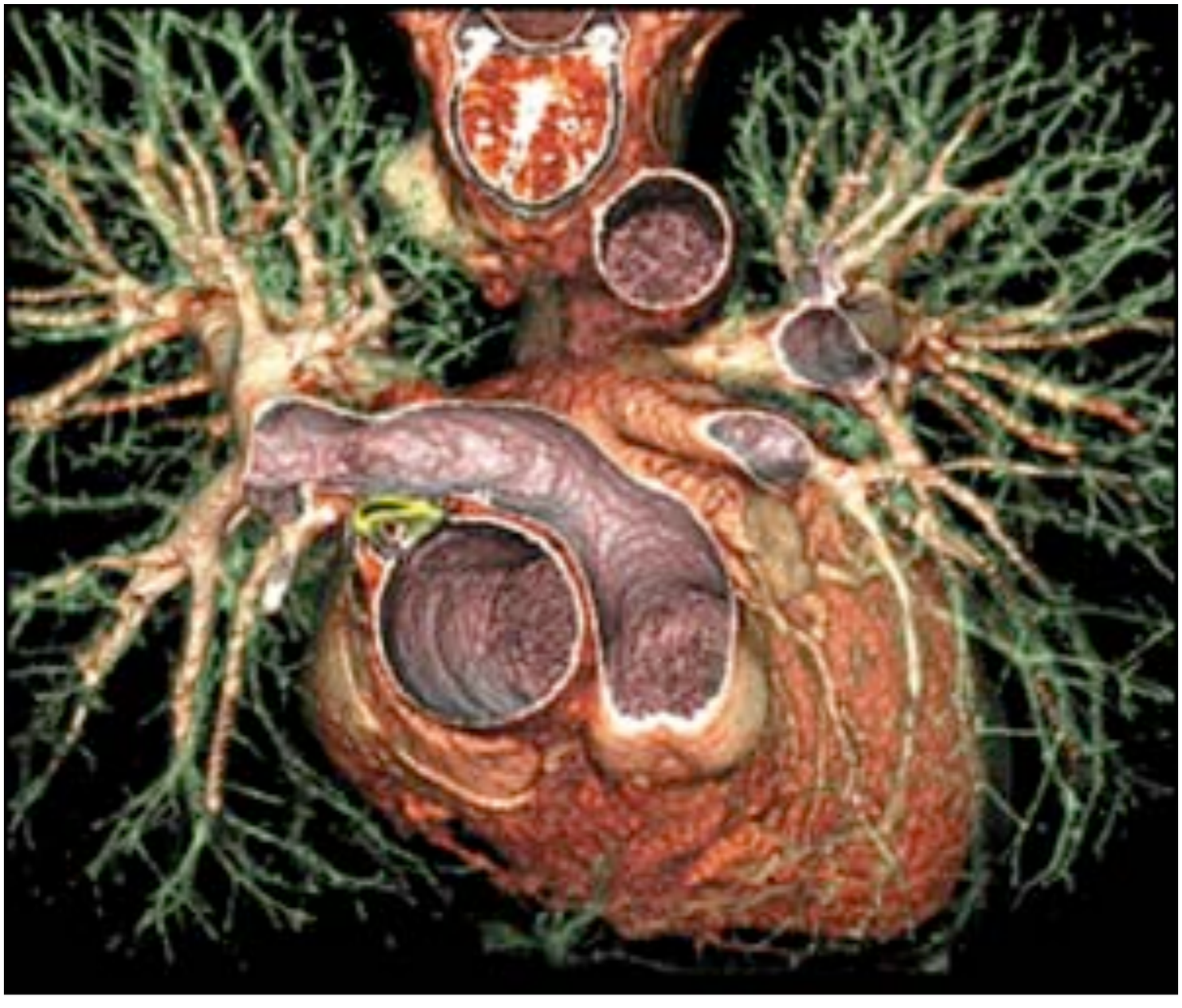
3D Objects



H&B Figure 10.46

This one?

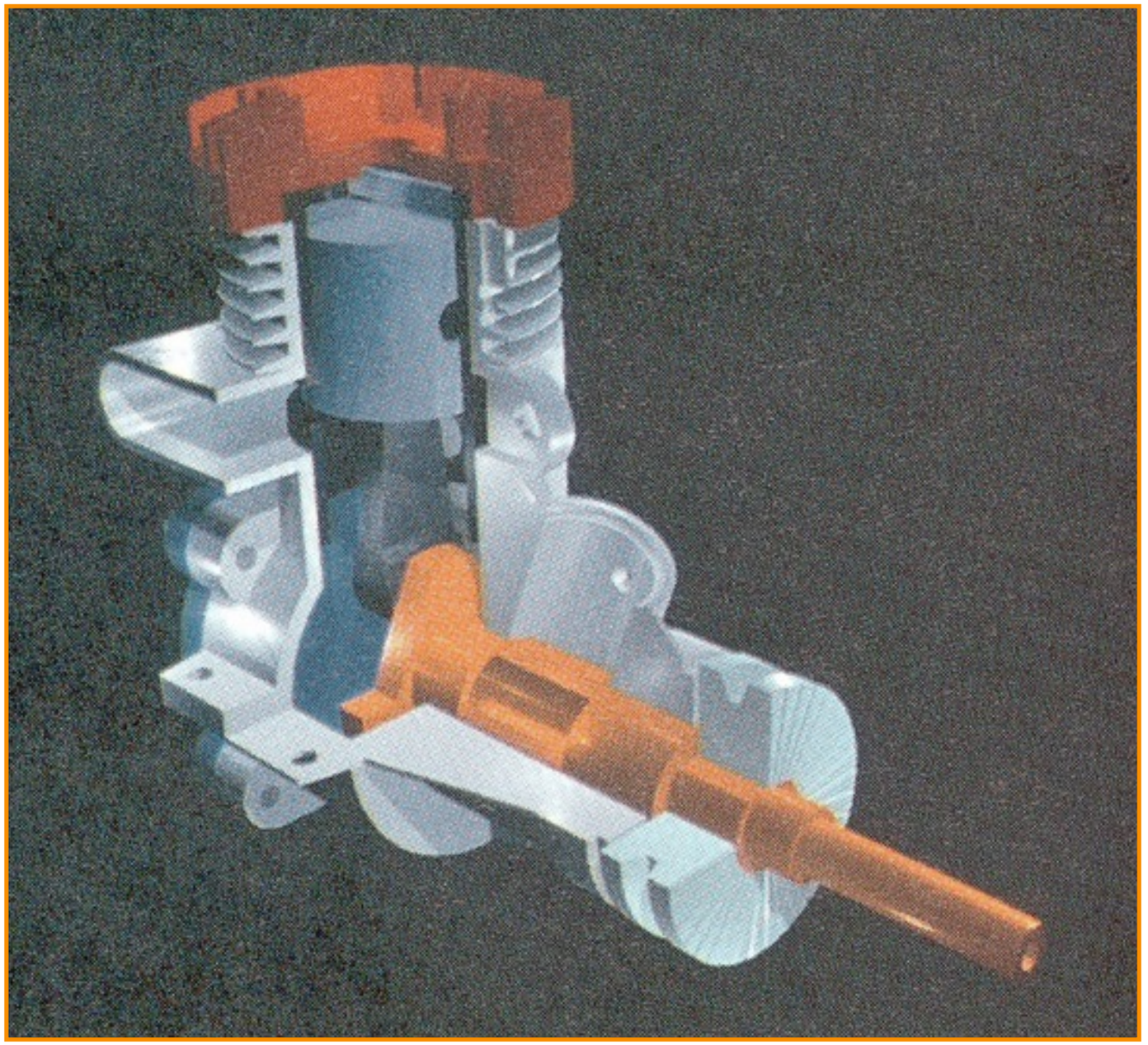
3D Objects



Imaging Economics

How about this one?

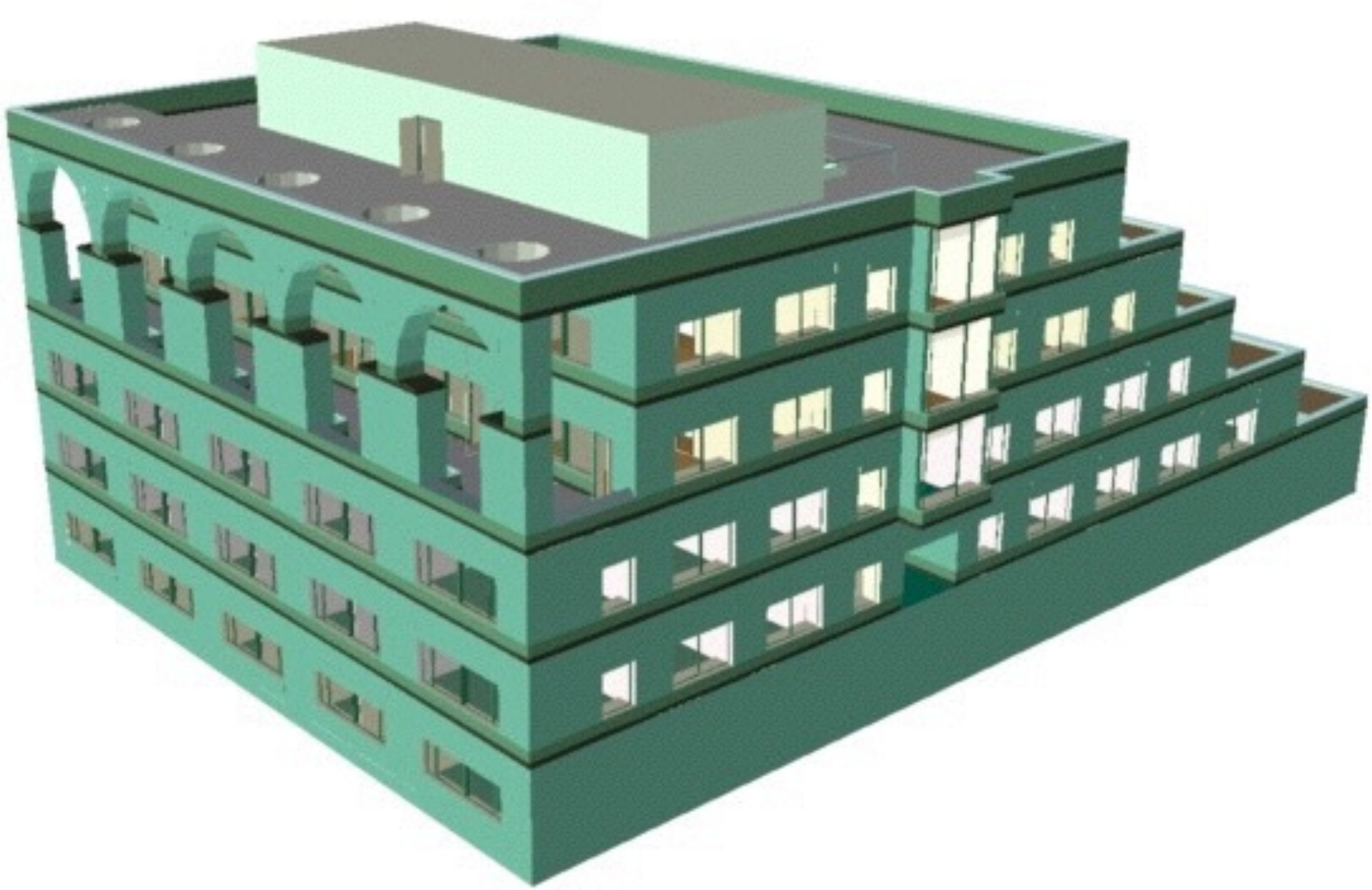
3D Objects



H&B Figure 9.9

This one?

3D Objects



This one?

Representations of Geometry

- 3D Representations provide the foundations for
  - Computer Graphics
  - Computer-Aided Geometric Design
  - Visualization
  - Robotics

- They are languages for describing geometry

  data structures **algorithms**

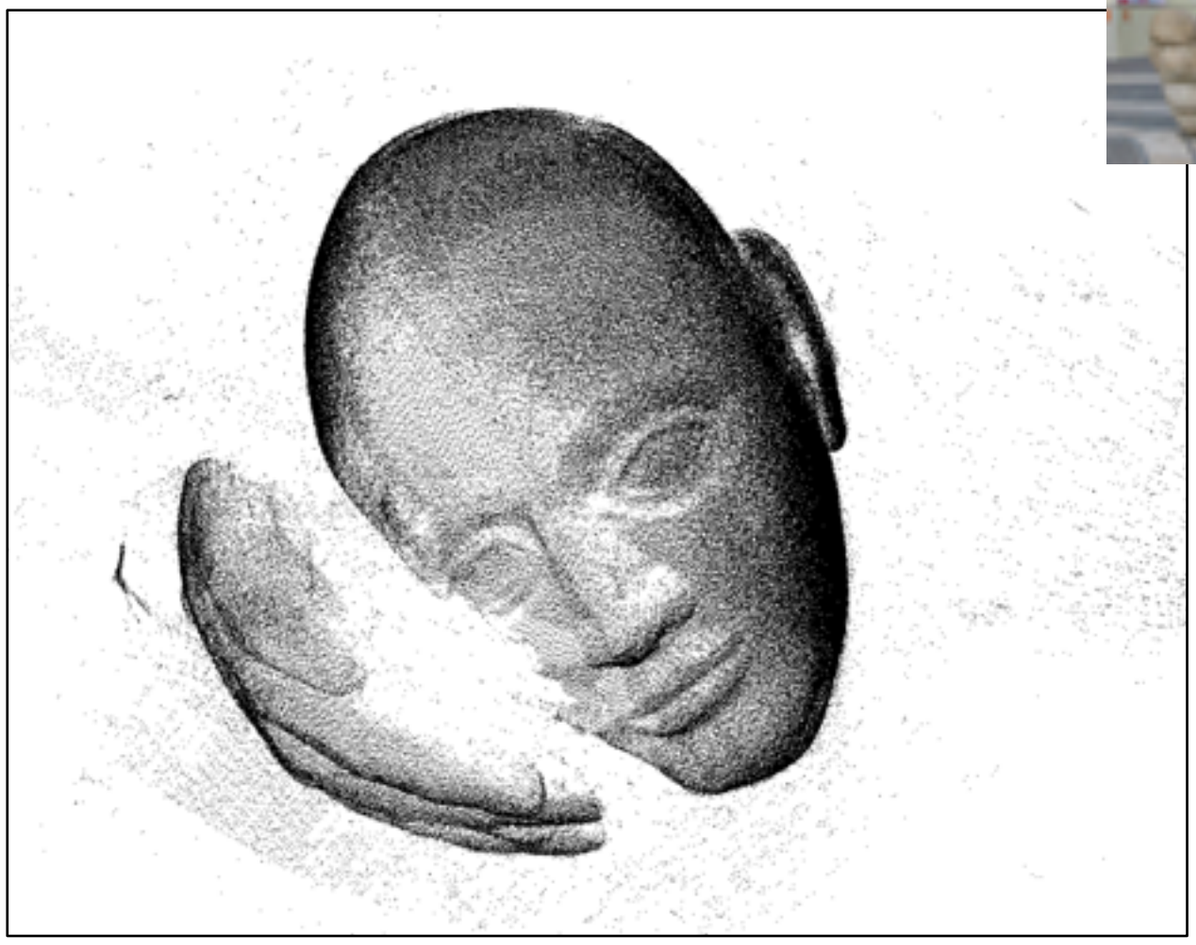
- Data structures determine algorithms!

3D Object Representations

- Raw data
  - Point cloud
  - Range image
  - Polygon soup
- Surfaces
  - Mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Skeleton
  - Application specific

Point Cloud

- Unstructured set of 3D point samples
  - Acquired from range finder, random sampling, particle system implementations, etc

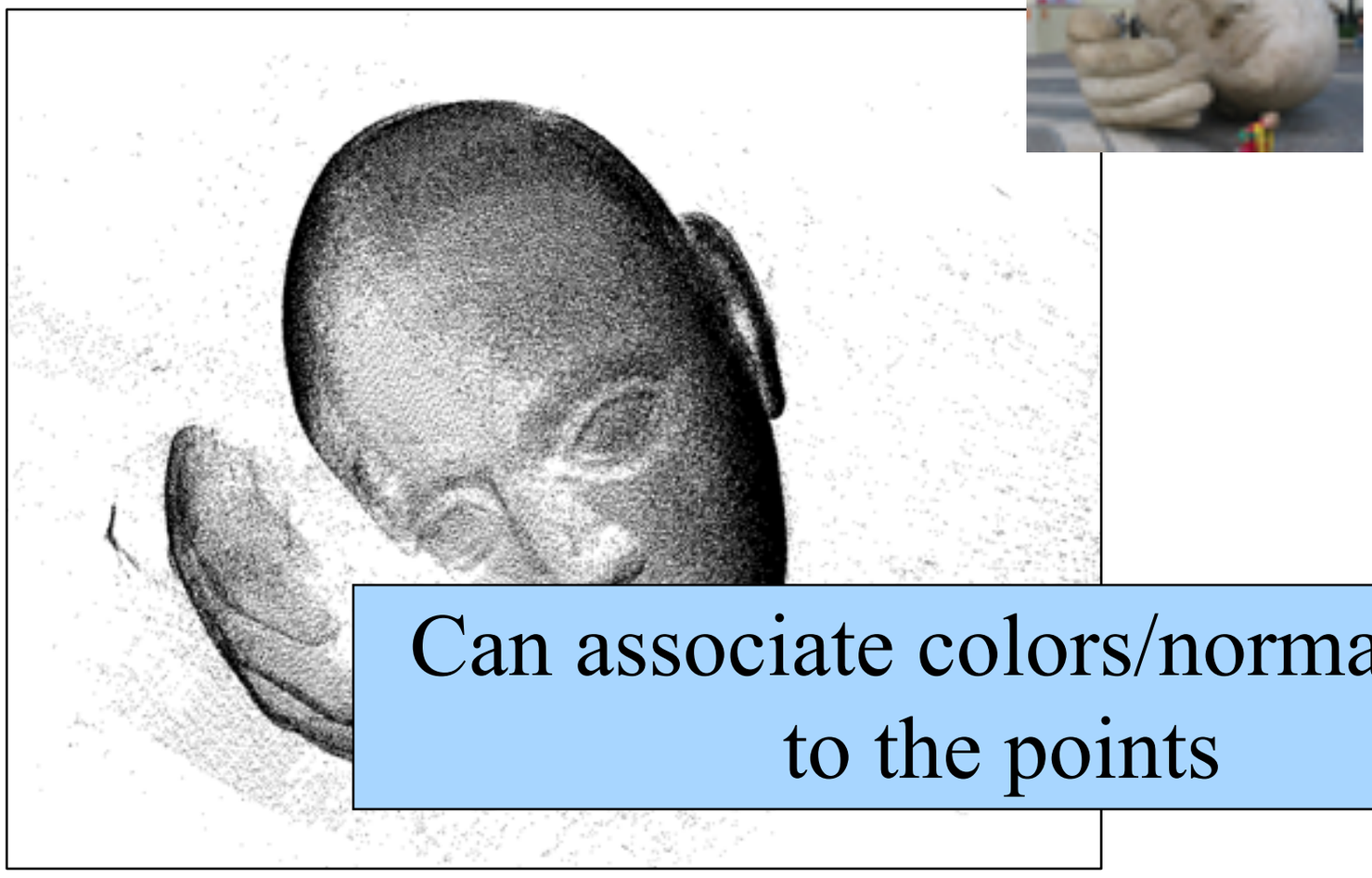


Hoppe

Czech Academy of Sciences

Point Cloud

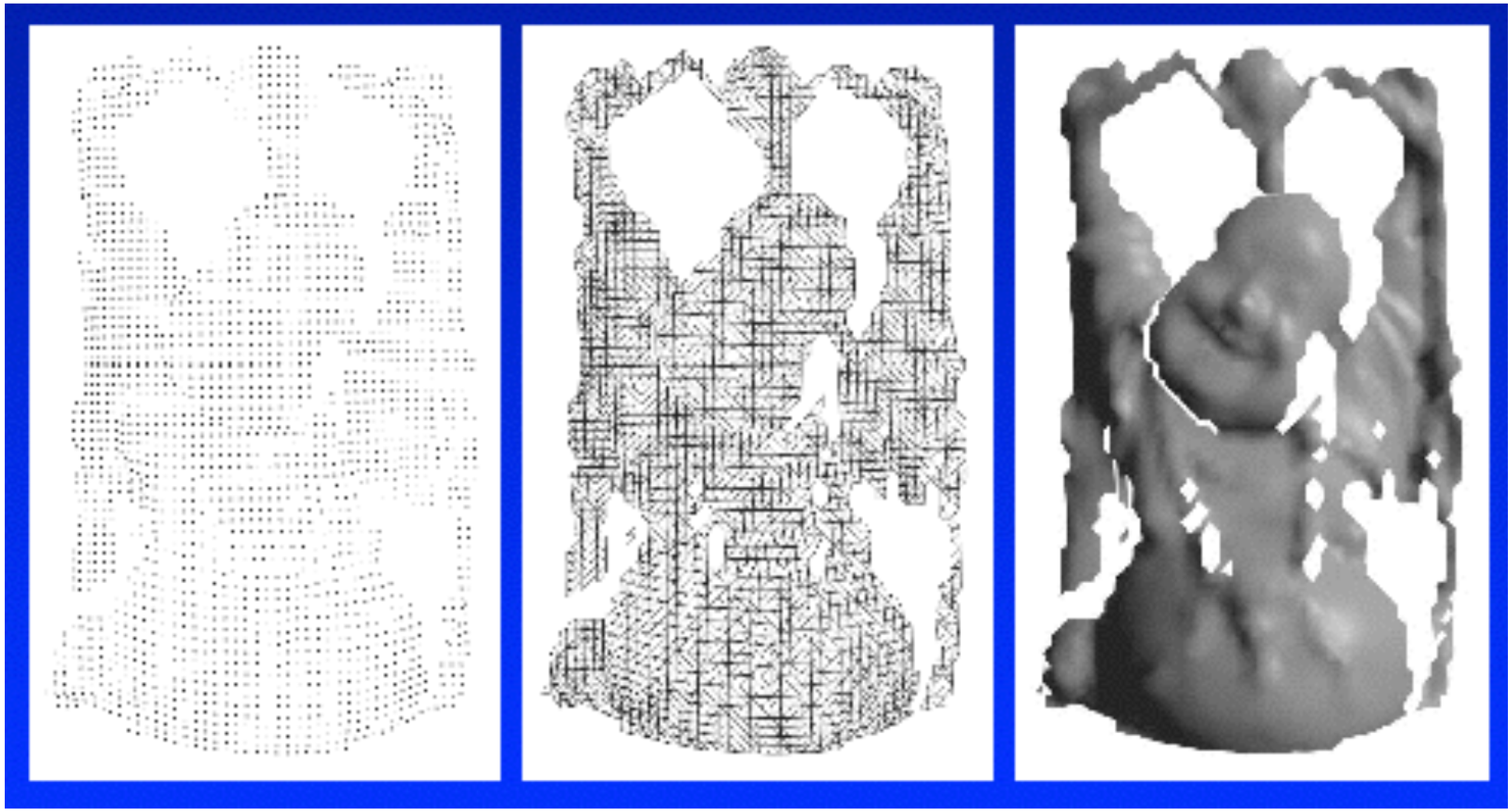
- Unstructured set of 3D point samples
  - Acquired from range finder, random sampling, particle system implementations, etc



Can associate colors/normals/etc. to the points

Range Image

- An image storing depth instead of color
  - Acquired from range scanners — e.g. Microsoft Kinect

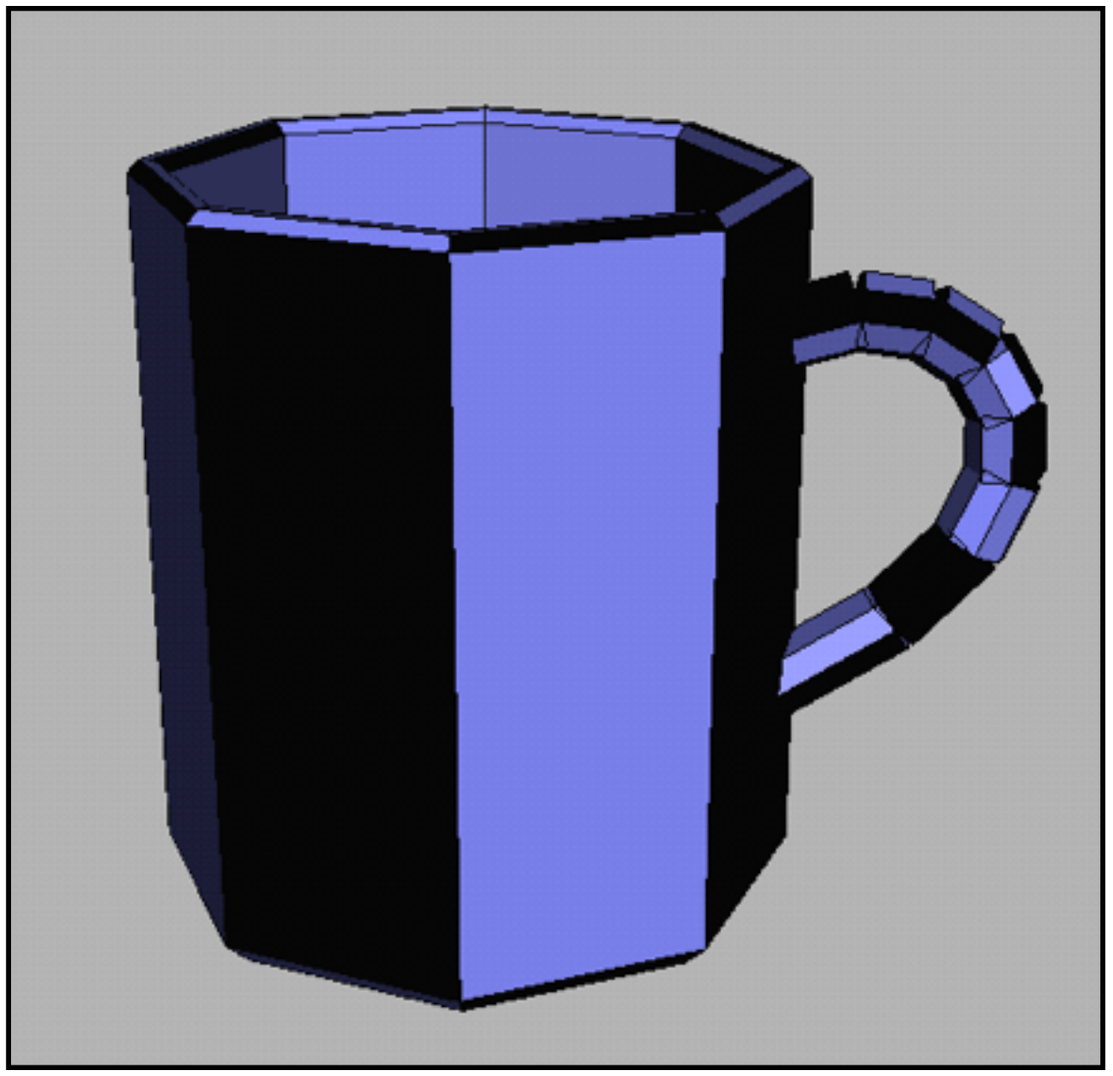


Range Image Tesselation Range Surface

Brian Curless
SIGGRAPH 99
Course #4 Notes

Polygon Soup

- Unstructured set of polygons
  - Created with interactive modeling systems, combining range images, etc.



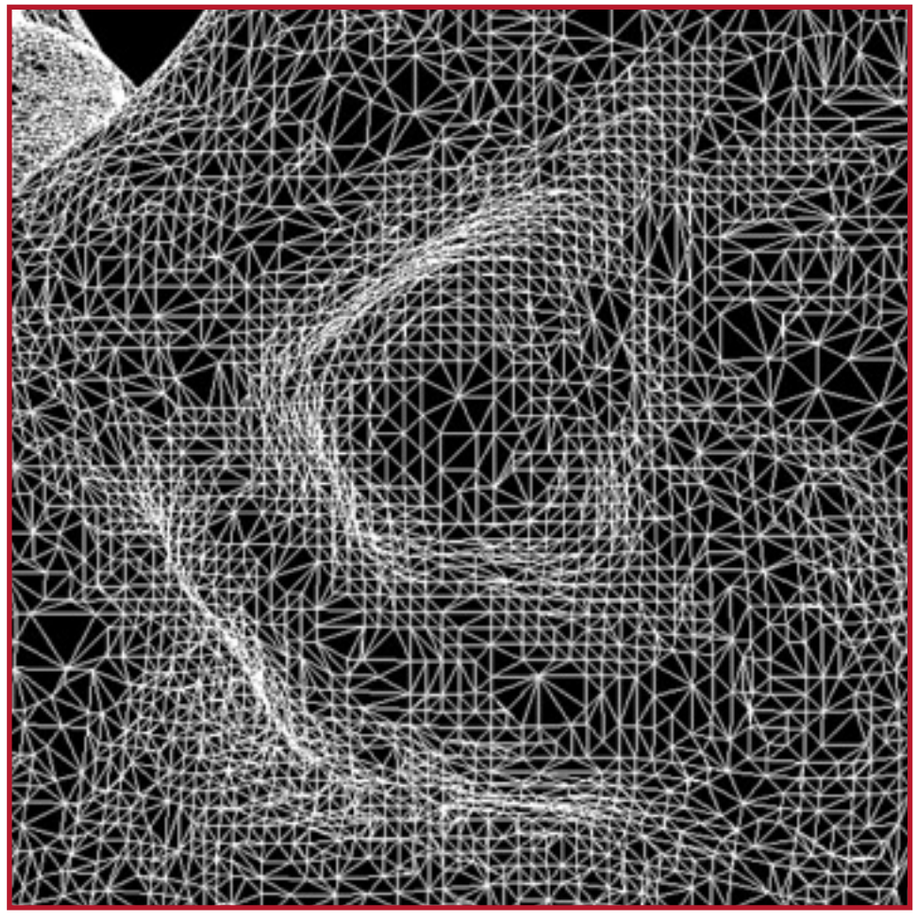
Larson

3D Object Representations

- Raw data
  - Point cloud
  - Range image
  - Polygon soup
- Surfaces
  - Mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Skeleton
  - Application specific

Mesh

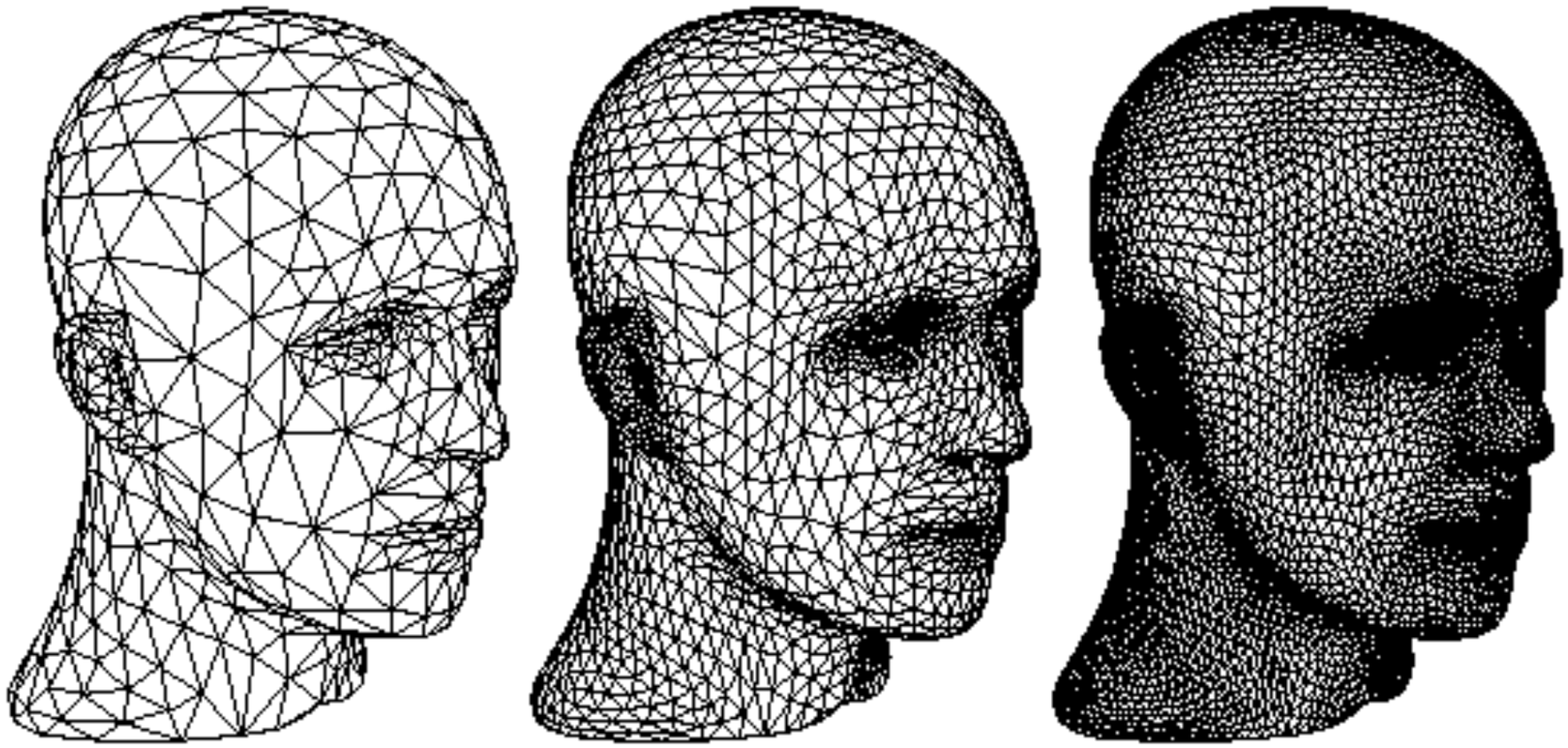
- Connected set of polygons (usually triangles)
  - May not be closed



Stanford Graphics Laboratory

Subdivision Surface

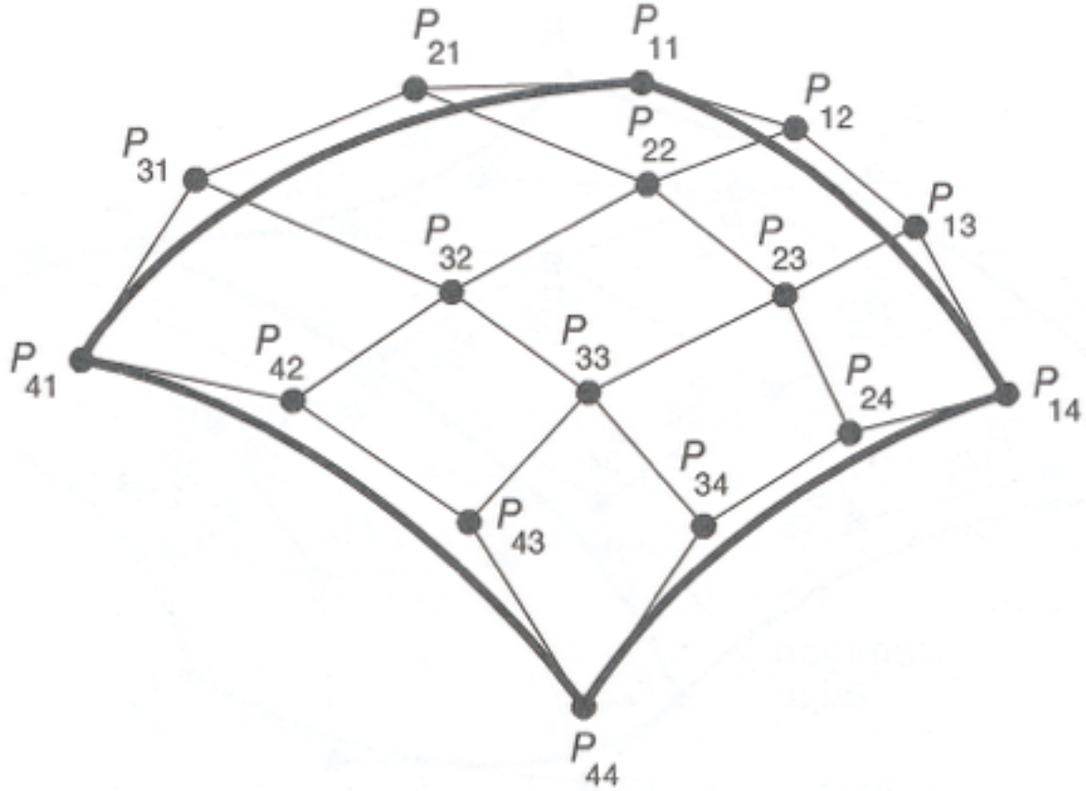
- Coarse mesh & subdivision rule
  - Define smooth surface as limit of sequence of refinements



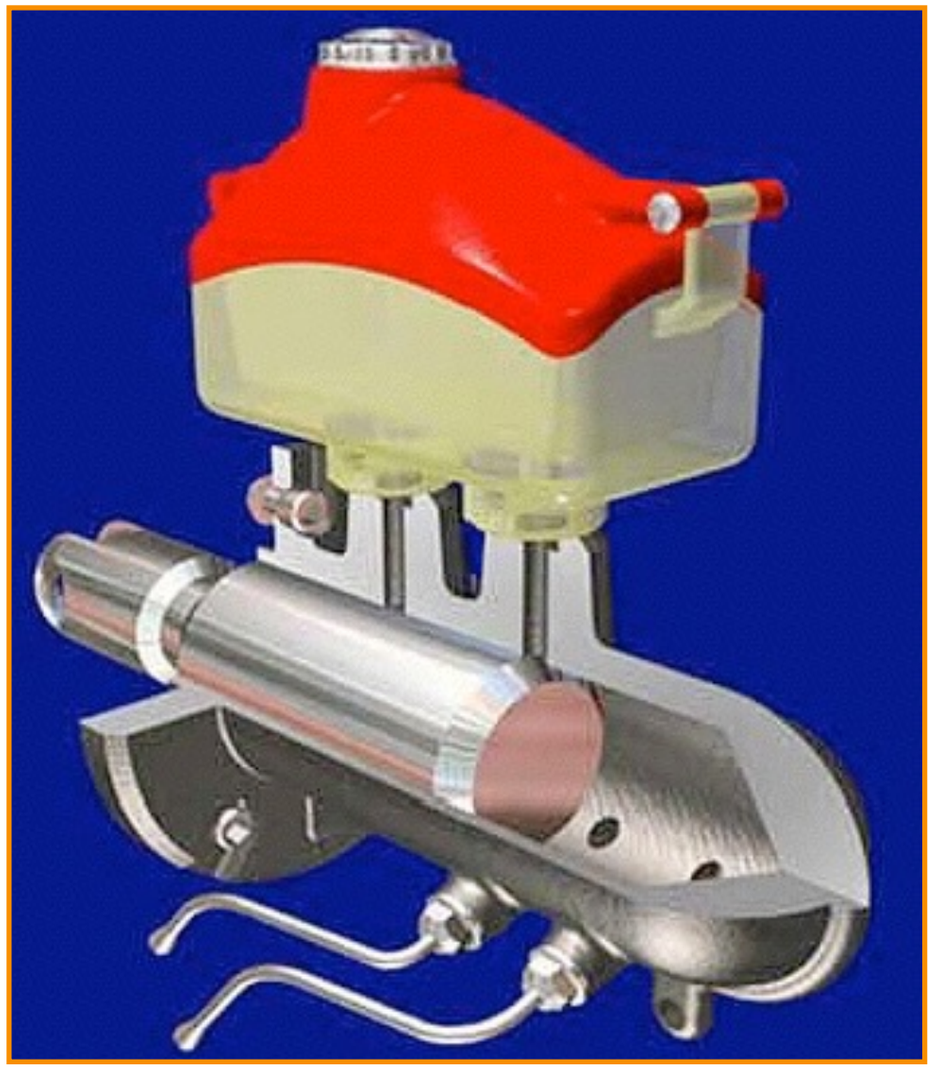
Zorin & Schroeder
SIGGRAPH 99
Course Notes

Parametric Surface

- Tensor product spline patches
  - Careful use of constraints to maintain continuity

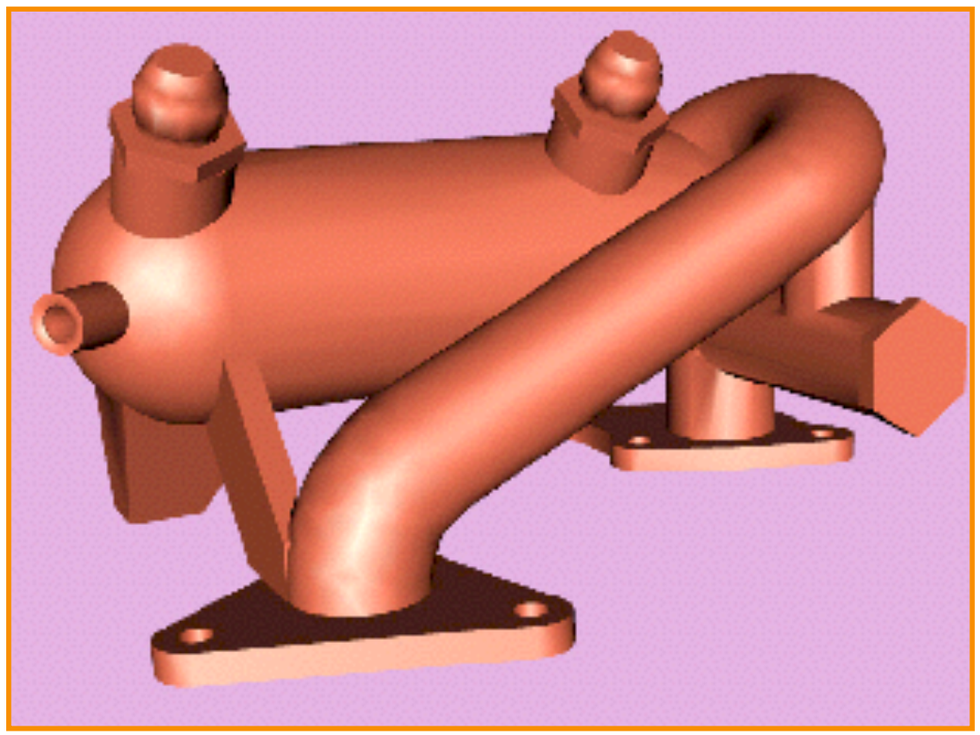


FvDFH Figure 11.44

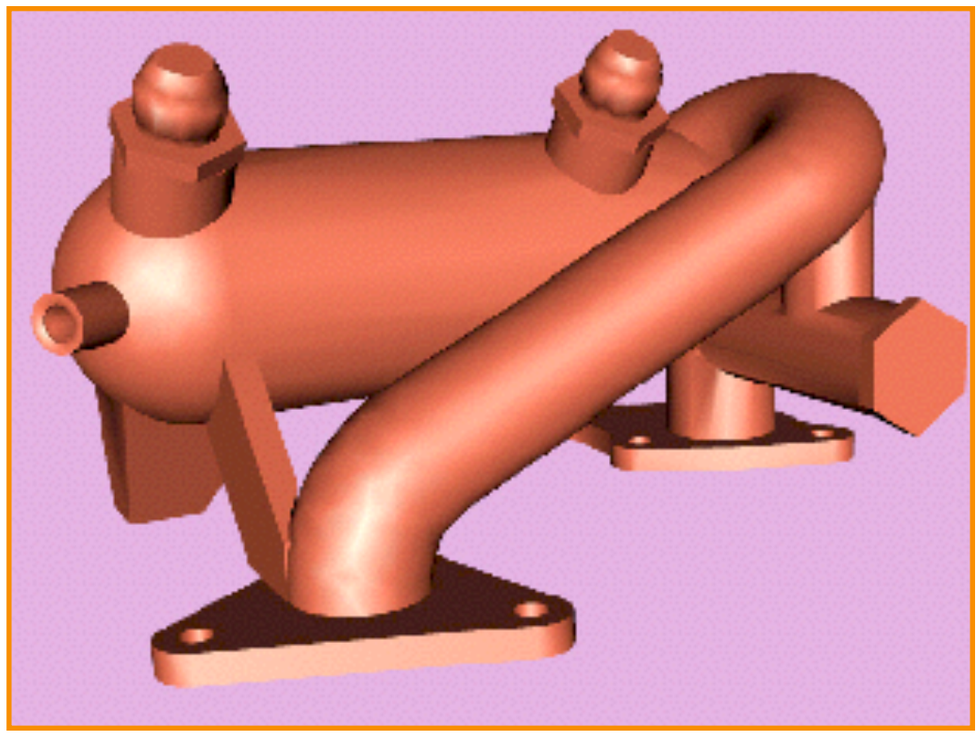


Implicit Surface

- Points satisfying: F(x,y,z) = 0



Polygonal Model



Implicit Model

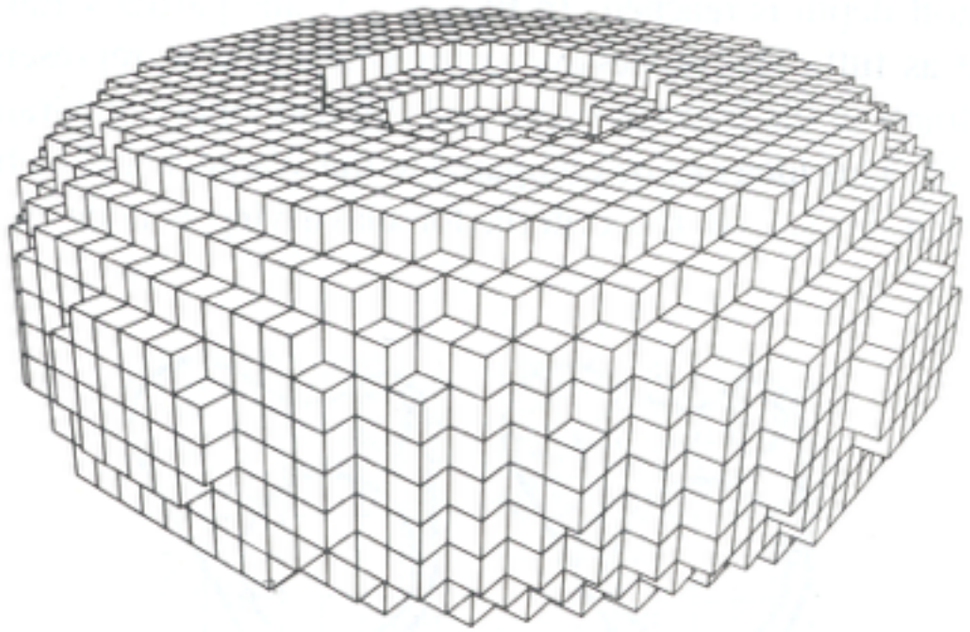
Bill Lorensen
SIGGRAPH 99
Course #4 Notes

3D Object Representations

- Raw data
  - Point cloud
  - Range image
  - Polygon soup
- Surfaces
  - Mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Skeleton
  - Application specific

Voxels

- Uniform grid of volumetric samples
  - Acquired from CT, MRI, etc.

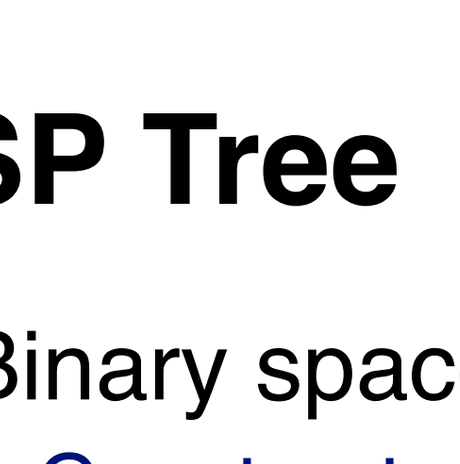


FvDFH Figure 12.20

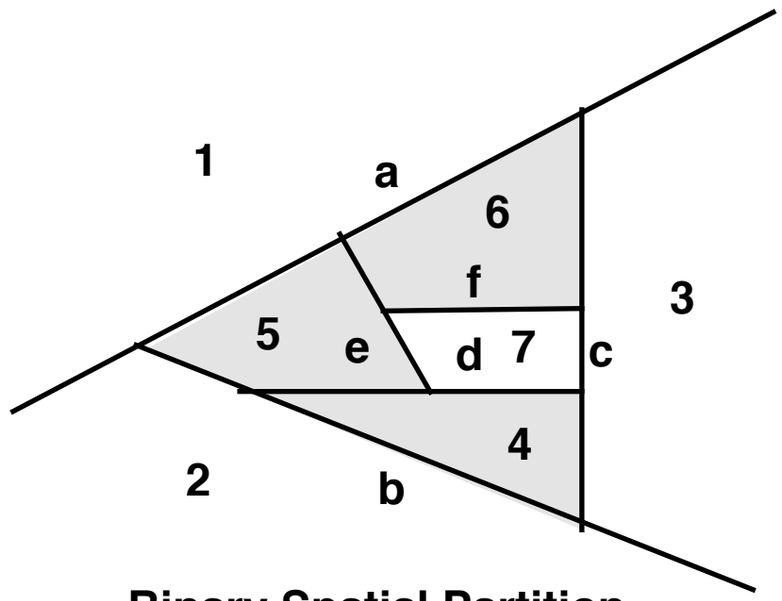
Stanford Graphics Laboratory

BSP Tree

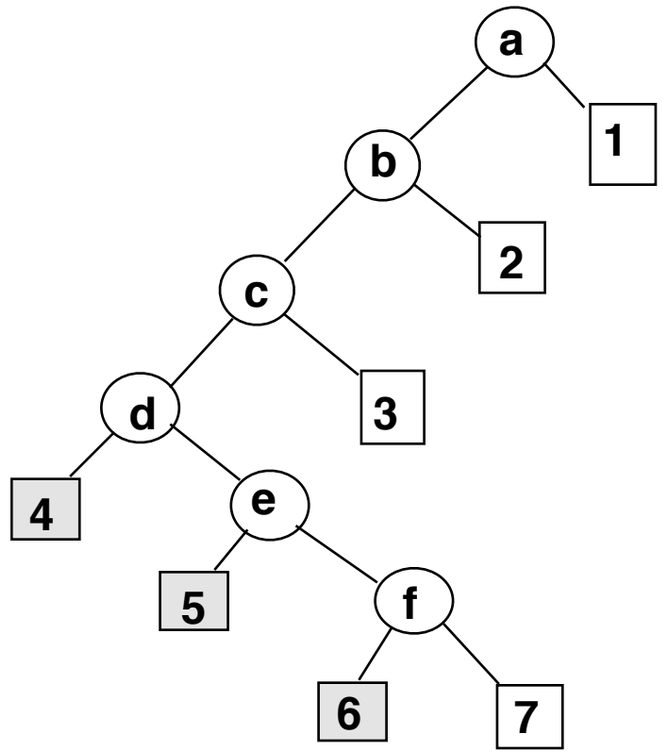
- Binary space partition with solid cells labeled
  - Constructed from polygonal representations



Object



Binary Spatial Partition

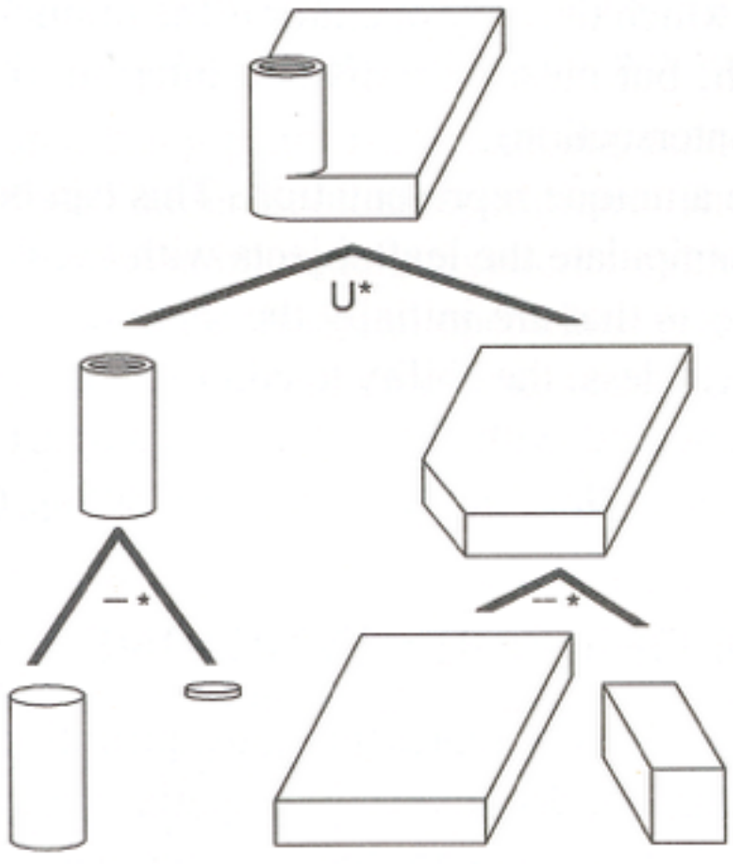


Binary Tree

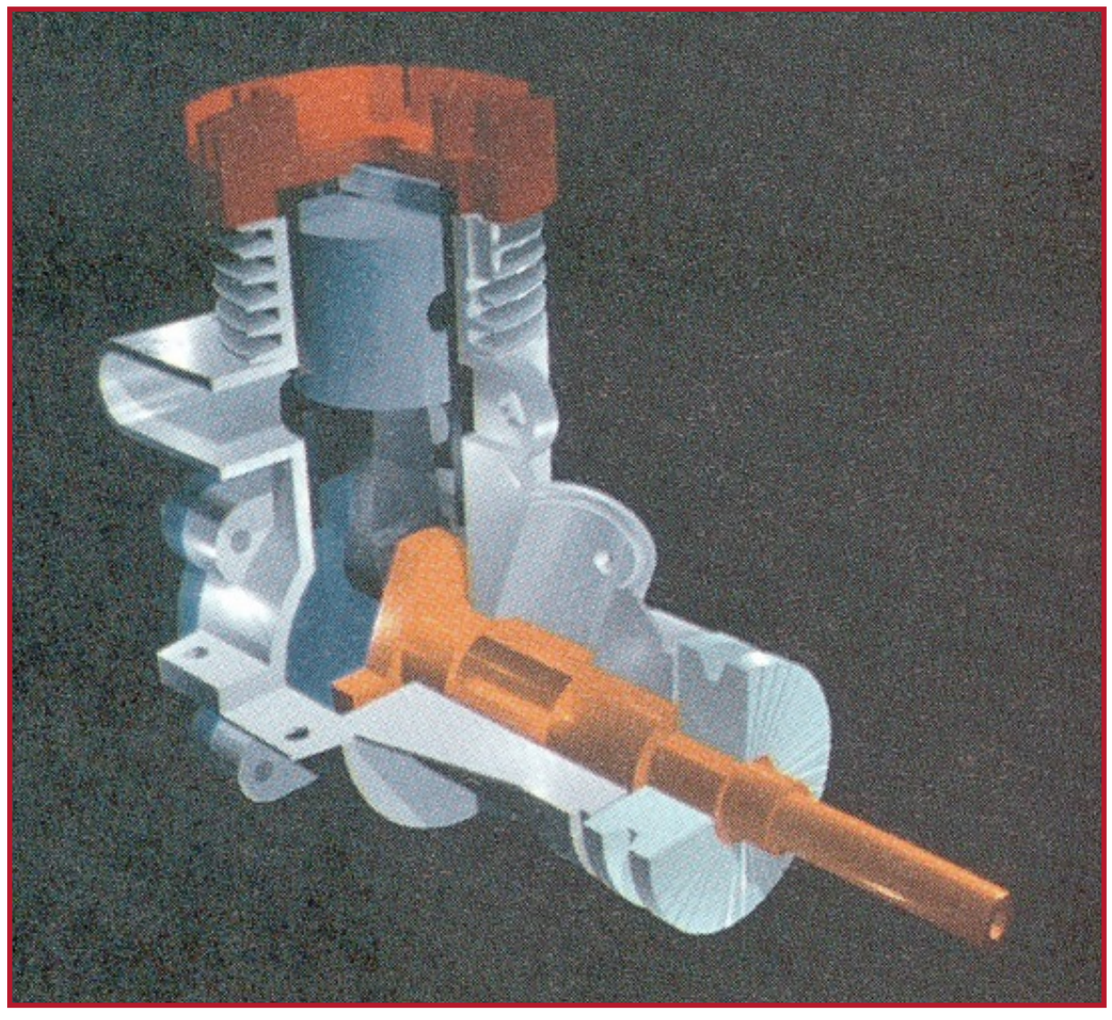
Naylor

Constructive Solid Geometry (CSG)

- Hierarchy of boolean set operations (union, difference, intersect) applied to simple shapes



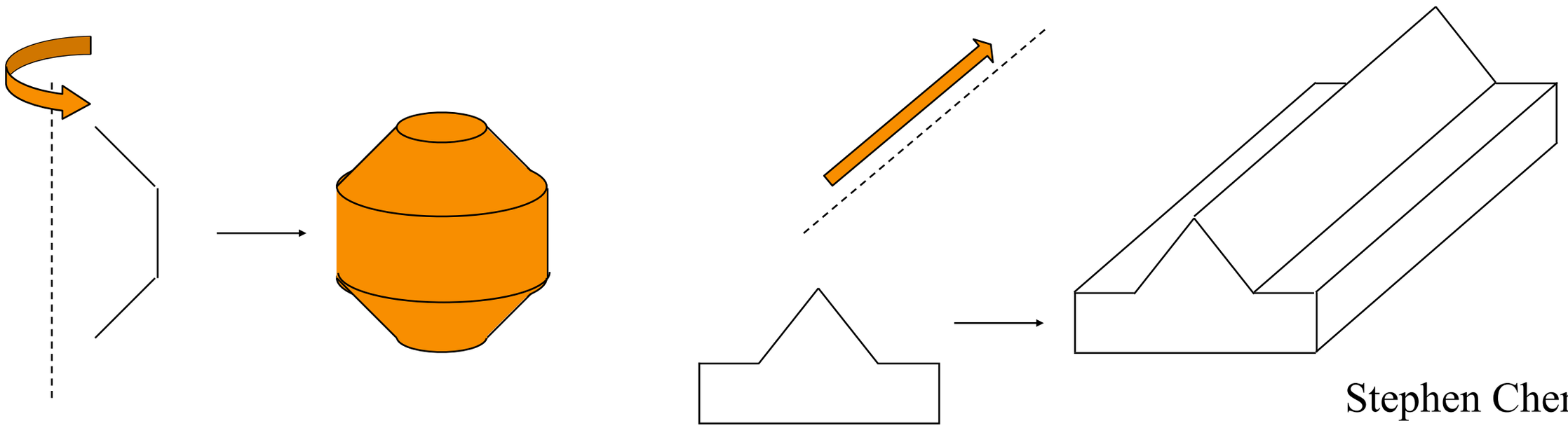
FvDFH Figure 12.27



H&B Figure 9.9

Sweep

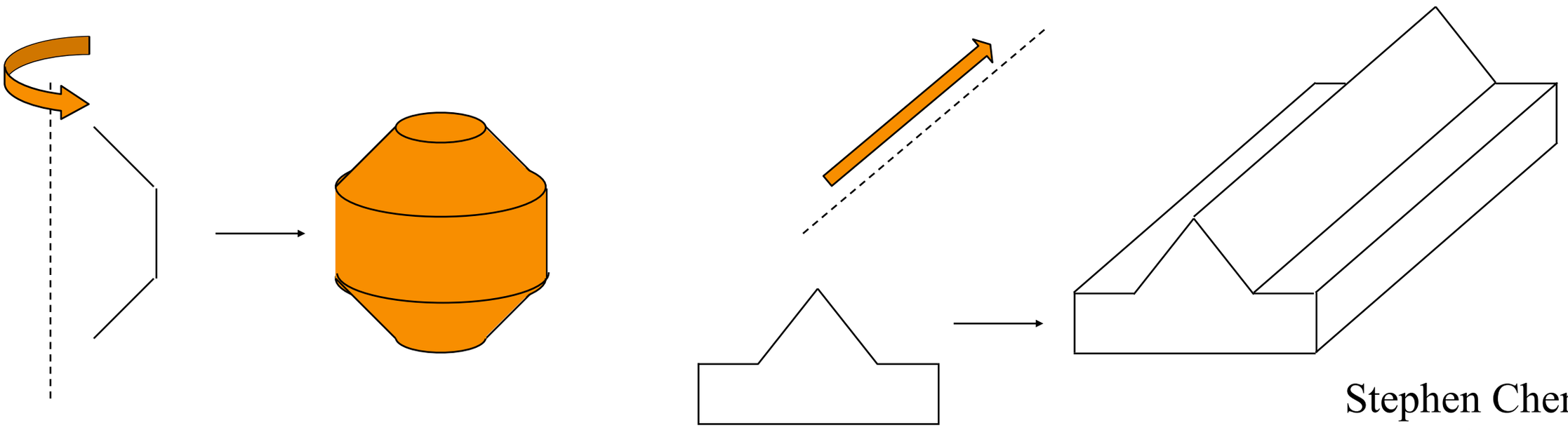
- Solid swept by curve along trajectory



Stephen Chenney
U Wisconsin

Sweep

- Solid swept by curve along trajectory



Stephen Chenney
U Wisconsin

- Curve may be arbitrary
- Sweep polygon may deform (scale, rotate) with respect to the path orientation

Example of Several Representations

- Scalable KinectFusion
- Which representations are being used?

3D Object Representations

- Raw data
  - Point cloud
  - Range image
  - Polygon soup
- Surfaces
  - Mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Skeleton
  - Application specific

Scene Graph

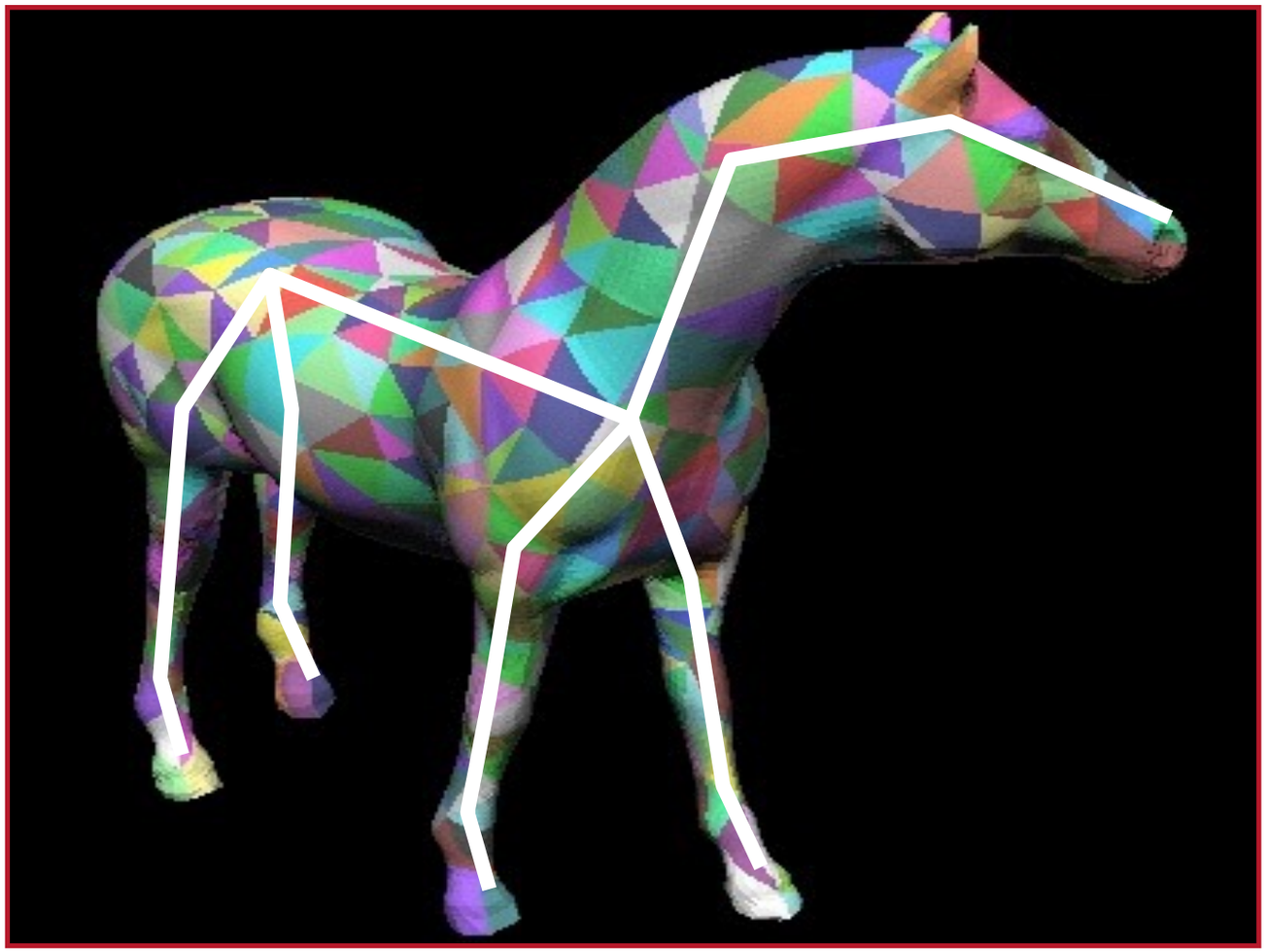
- Union of objects at leaf nodes

Bell Laboratories

avalon.viewpoint.com

Skeleton

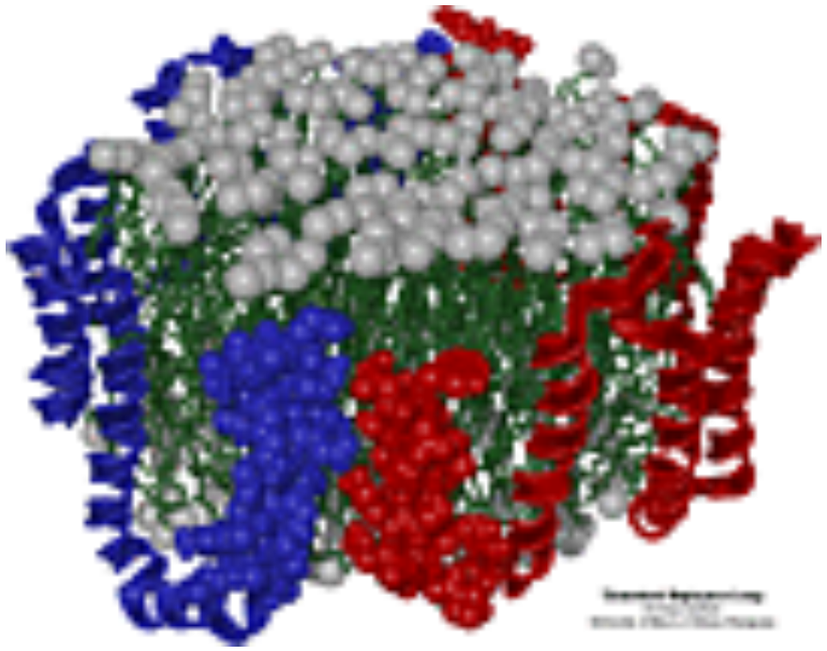
- Graph of curves with radii



Stanford Graphics Laboratory

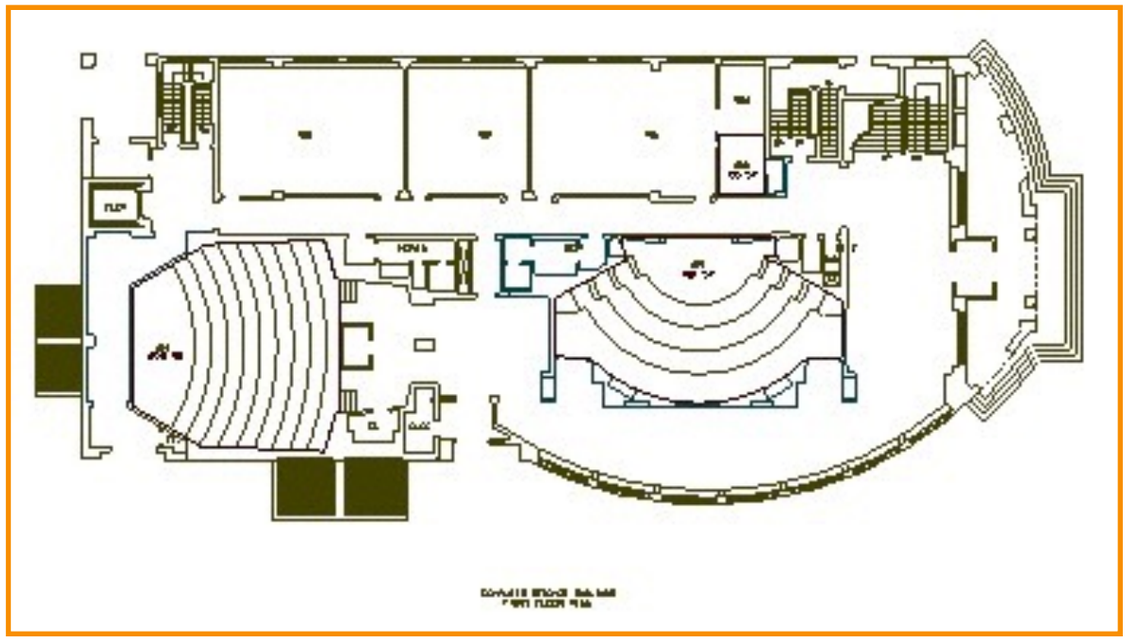
SGI

Application Specific



Apo A-1

(Theoretical Biophysics Group,
University of Illinois at Urbana-Champaign)



Architectural Floorplan

Equivalence of Representations

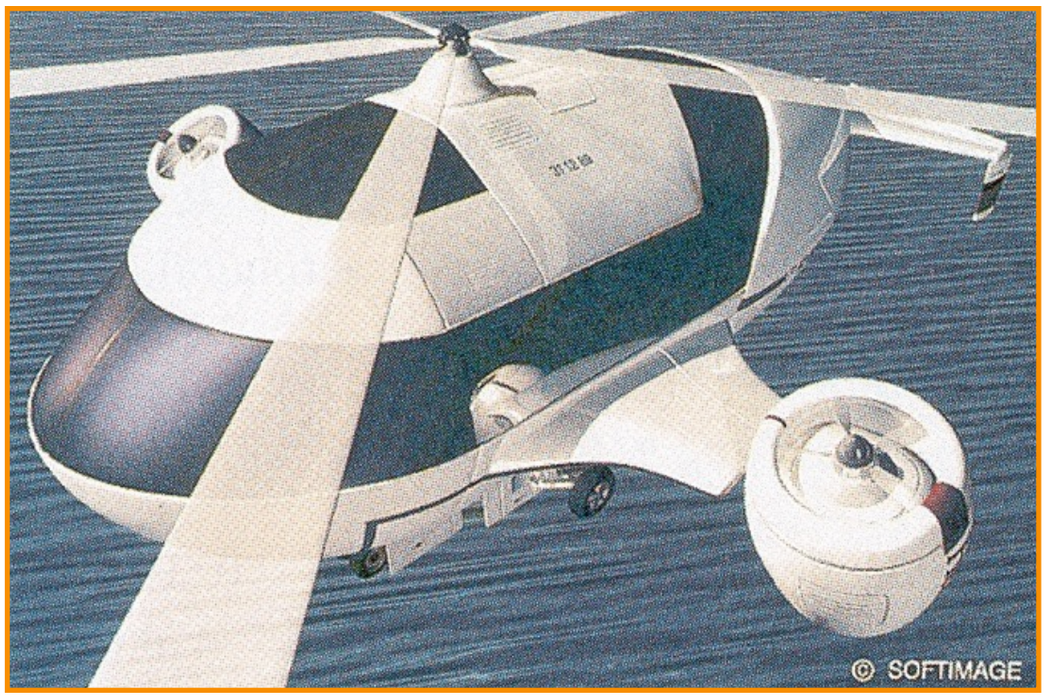
- Thesis:
  - Each fundamental representation has enough expressive power to model the shape of any geometric object
  - It is possible to perform all geometric operations with any fundamental representation!
- Analogous to Turing-Equivalence:
  - All computers today are Turing-equivalent, but we still have many different processors

Computational Differences

- Efficiency
  - Combinatorial complexity
  - Space/time trade-offs
  - Numerical accuracy/stability
- Simplicity
  - Ease of acquisition
  - Hardware acceleration
- Usability

Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections



H&B Figure 10.46

Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections

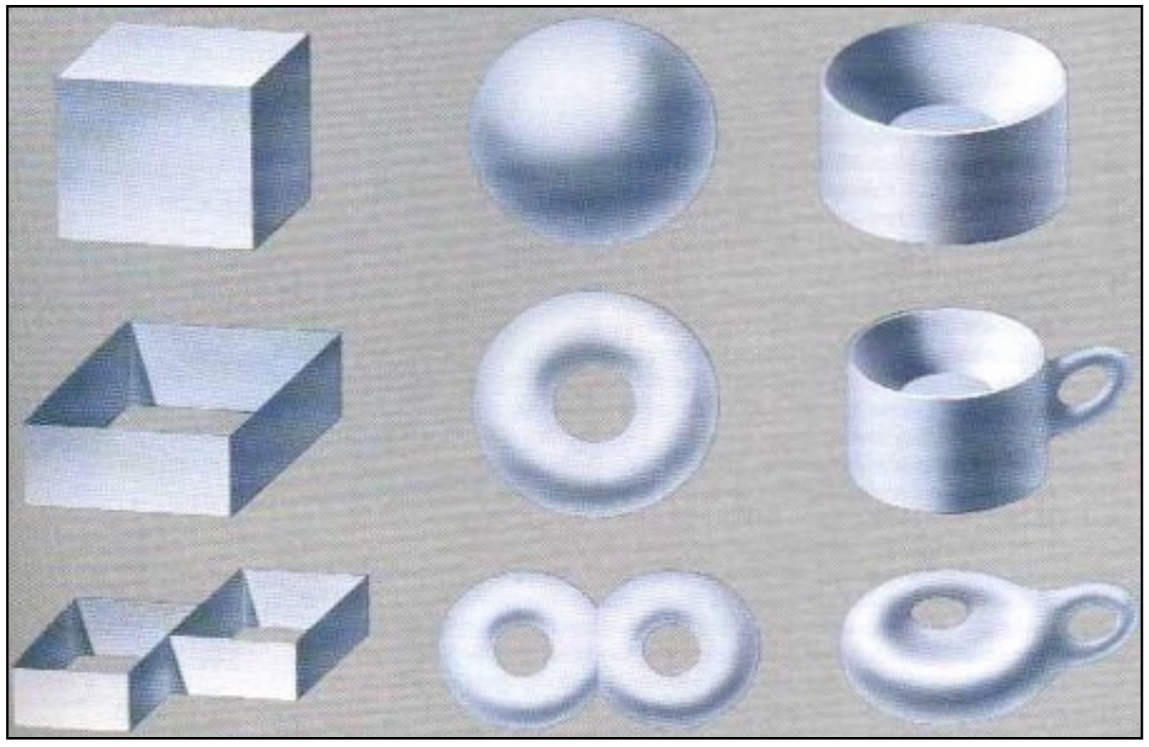
Not Local Support

Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections

Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections



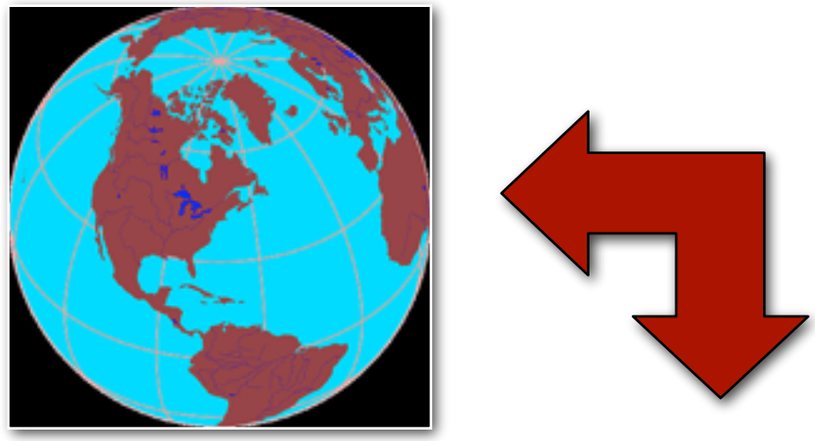
Topological Genus Equivalences

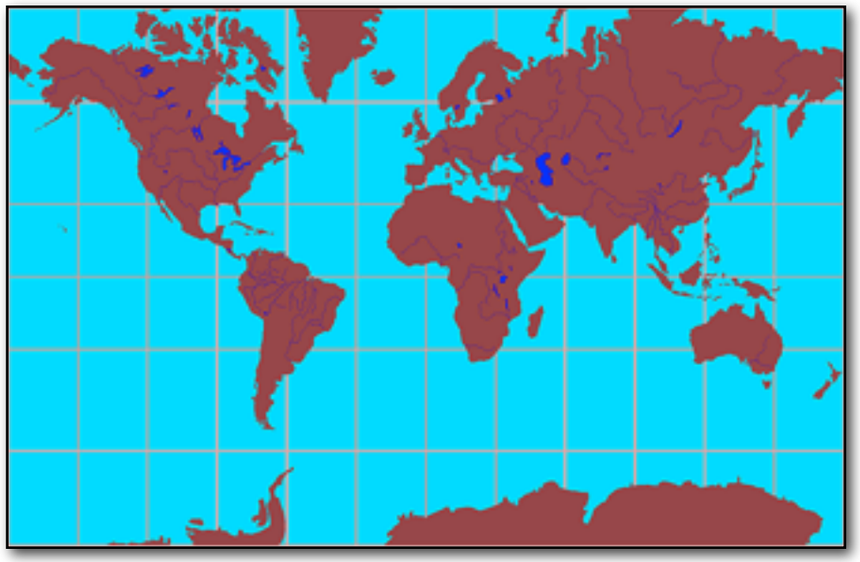
Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections

Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections





A Parameterization (not necessarily natural)

Surfaces

- What makes a good surface representation?
  - Concise
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections