Implicit Surfaces & Solid Representations

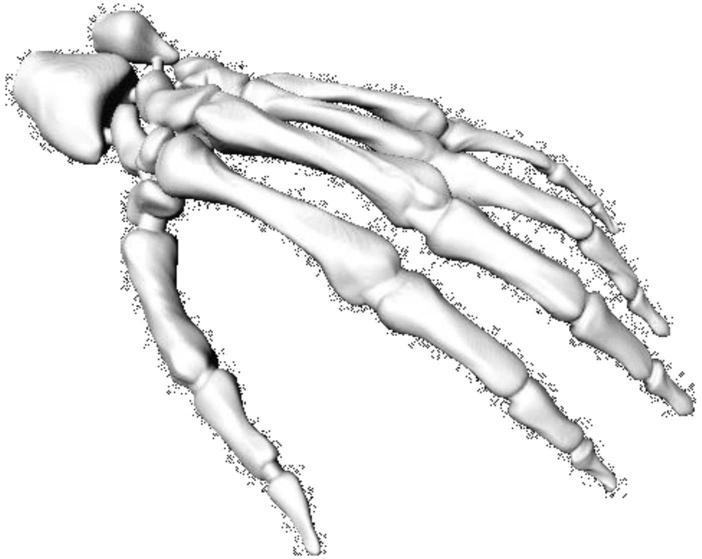
COS 426, Spring 2022
Felix Heide
Princeton University

3D Object Representations

- Raw data
  - Range image
  - Point cloud
- Surfaces
  - Polygonal mesh
  - Subdivision
  - Parametric
  - ➢ Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Application specific

3D Object Representations

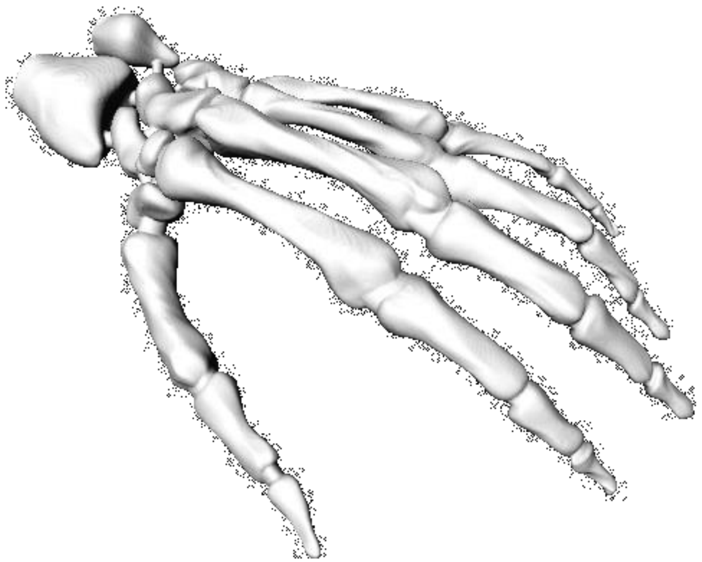
- Desirable properties of an object representation
  - Easy to acquire
  - Accurate
  - Concise
  - Intuitive editing
  - Efficient editing
  - Efficient display
  - Efficient intersections
  - Guaranteed validity
  - Guaranteed smoothness
  - etc.



Large Geometric Model Repository
Georgia Tech

3D Object Representations

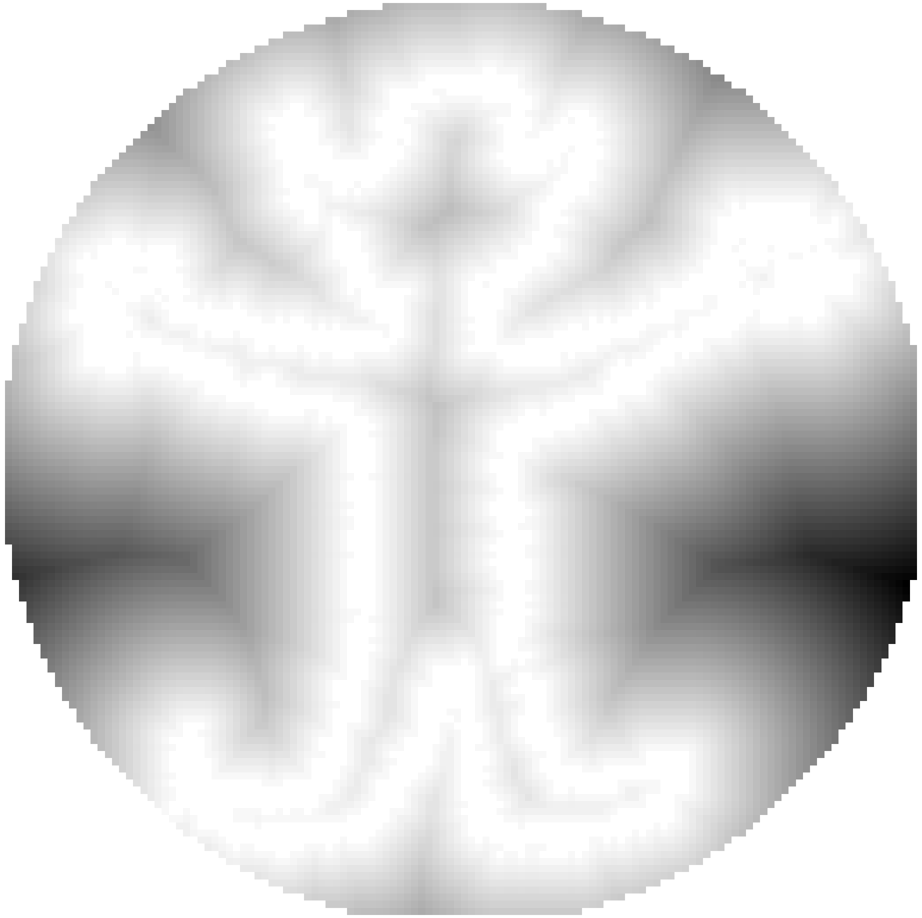
- Desirable properties of an object representation
  - Easy to acquire
  - Accurate
  - Concise
  - Intuitive editing
  - Efficient editing
  - Efficient display
  - **Efficient *intersections***
  - **Guaranteed validity**
  - Guaranteed smoothness
  - etc.



Large Geometric Model Repository
Georgia Tech

Implicit Surfaces

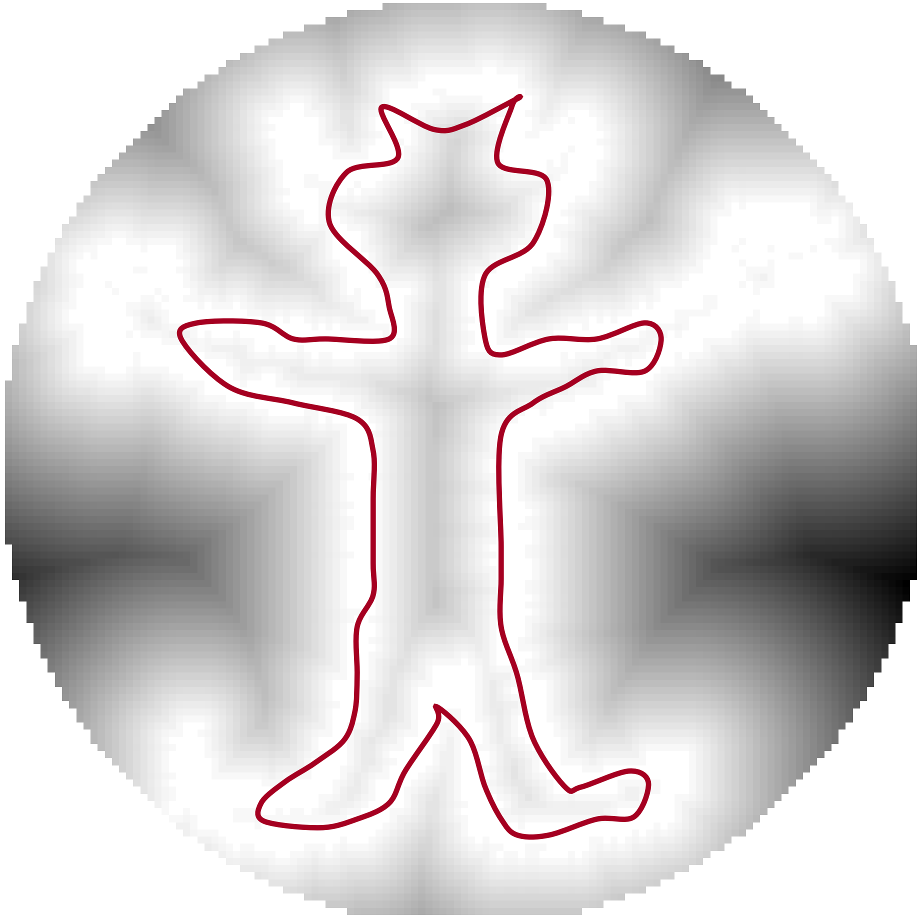
- Represent surface with function
  over space



Kazhdan

Implicit Surfaces

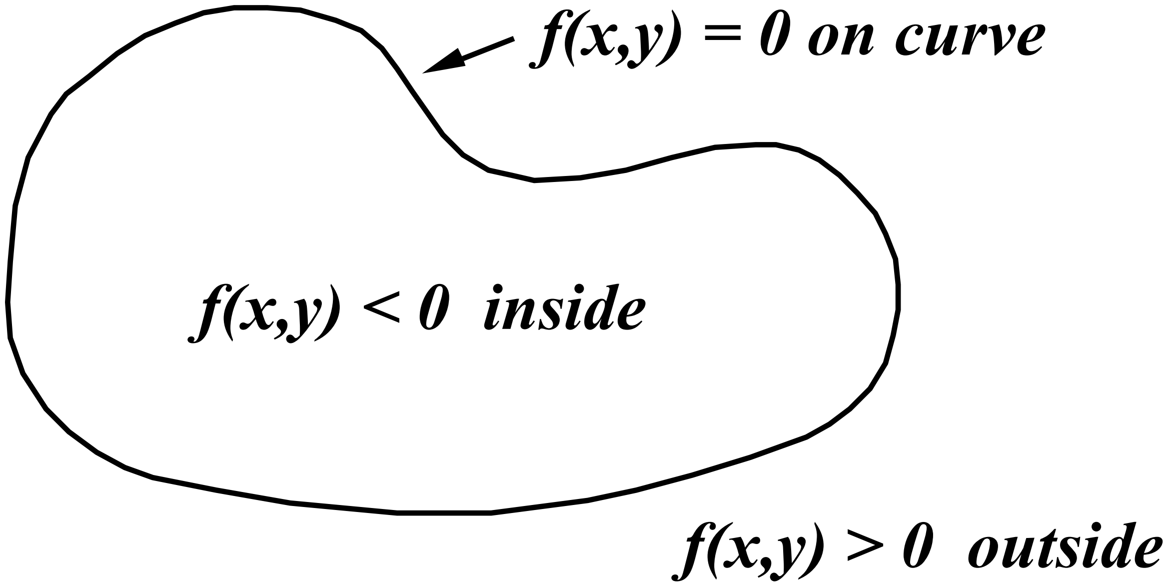
- Surface defined implicitly by function



Kazhdan

Implicit Surfaces

- Surface defined implicitly by function:
  - f (x, y, z) = 0 (on surface)
  - f (x, y, z) < 0 (inside)
  - f (x, y, z) > 0 (outside)

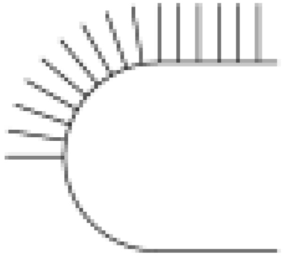


Turk

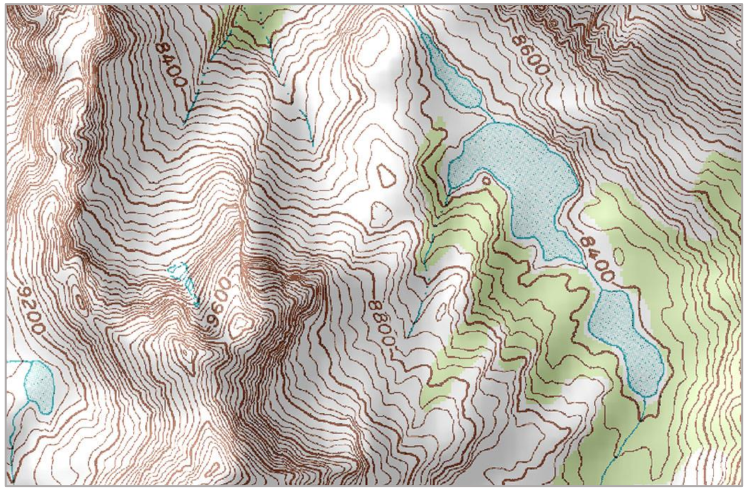
Implicit Surfaces

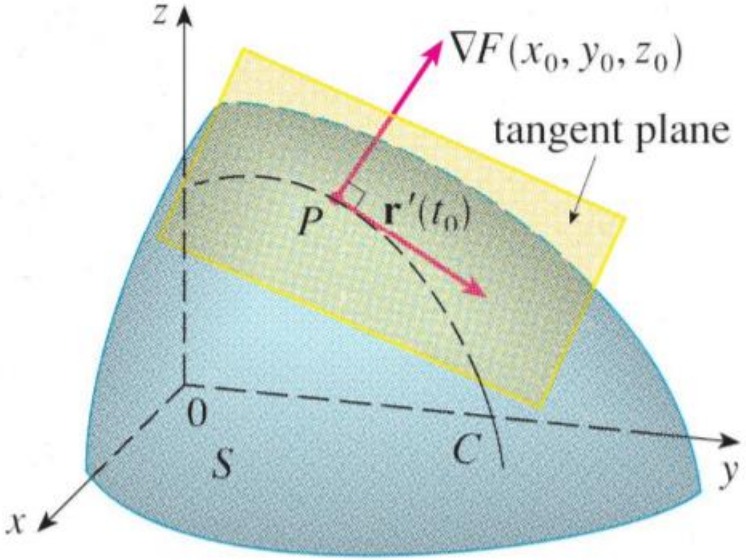
- Normals defined by partial derivatives
  - Normal $N(x, y, z) = normalize\left(\frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}, \frac{\partial F}{\partial z}\right) = normalize(\vec{\nabla} F)$
    - Example: circle $x^2 + y^2 - 3^2 = 0$
    - Proof: straight forward with an arbitrary curve $\Gamma(t)$ and the chain rule
    - Max change rate direction of F perpendicular

    - Intuition in 2D: skiing downhill on a topo-map



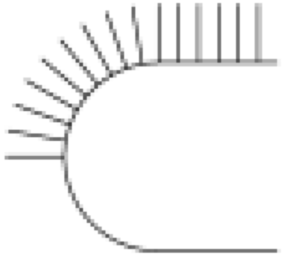
Normals





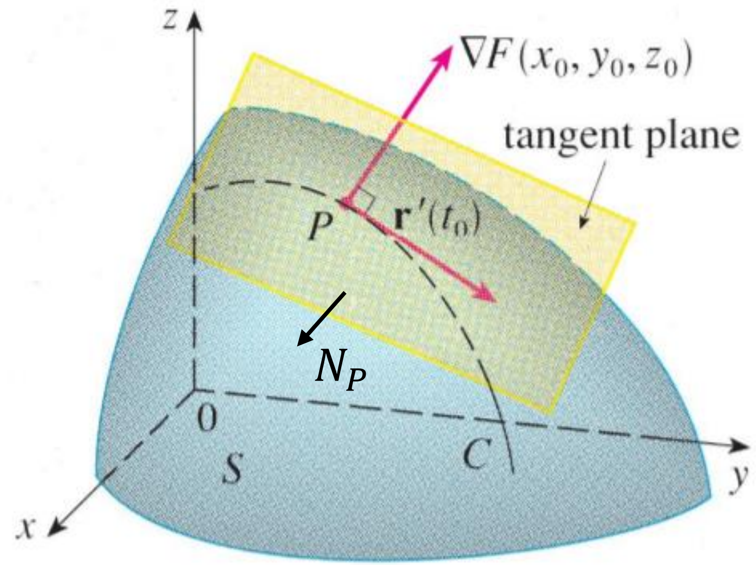
Implicit Surfaces

- Normals defined by partial derivatives
  - Normal $N(x, y, z) = normalize\left(\frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}, \frac{\partial F}{\partial z}\right) = normalize(\vec{\nabla} F)$
  - Tangent $r = N_P \times N$
    - on specific plane P, with normal N_P
    - Otherwise infinite directions



Normals

Tangents



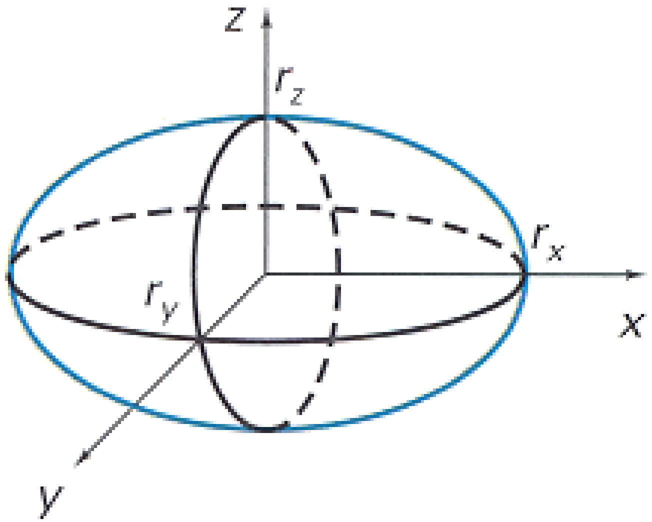
Bloomenthal

Intersection Computation

(1) Efficient check for whether point is inside

- Evaluate f(x,y,z) to see if point is inside/outside/on
- Example: ellipsoid

$$f(x,y,z) = \left(\frac{x}{r_x}\right)^2 + \left(\frac{y}{r_y}\right)^2 + \left(\frac{z}{r_z}\right)^2 - 1$$



H&B Figure 10.10

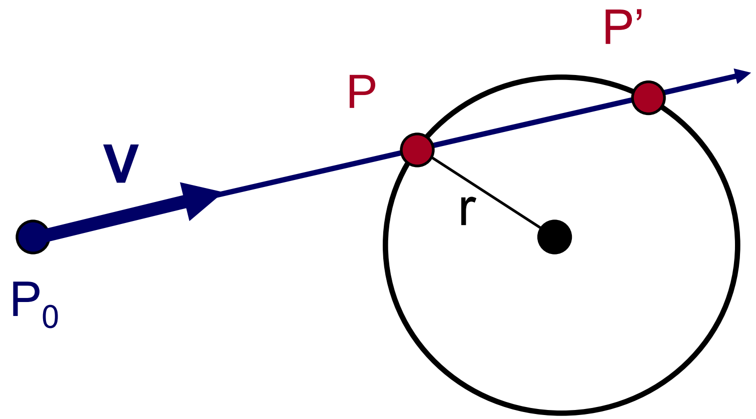
Intersection Computation

(2) Efficient surface intersections

- Substitute to find intersections

  Ray: $P = P_0 + tV$

  Sphere: $|P - O|^2 - r^2 = 0$



Intersection Computation

(2) Efficient surface intersections

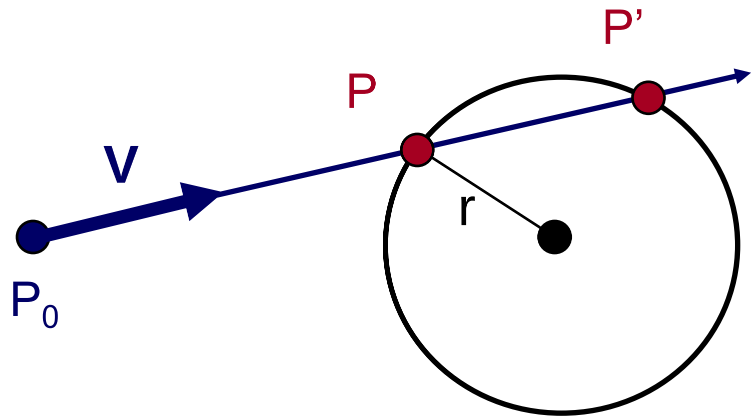
- Substitute to find intersections

  Ray: $P = P_0 + tV$

  Sphere: $|P - O|^2 - r^2 = 0$

  Substituting for P, we get:

  $$|P_0 + tV - O|^2 - r^2 = 0$$



Intersection Computation

(2) Efficient surface intersections

- Substitute to find intersections

Ray: $P = P_0 + tV$

Sphere: $|P - O|^2 - r^2 = 0$

Substituting for P, we get:

$$|P_0 + tV - O|^2 - r^2 = 0$$

Solve quadratic equation:

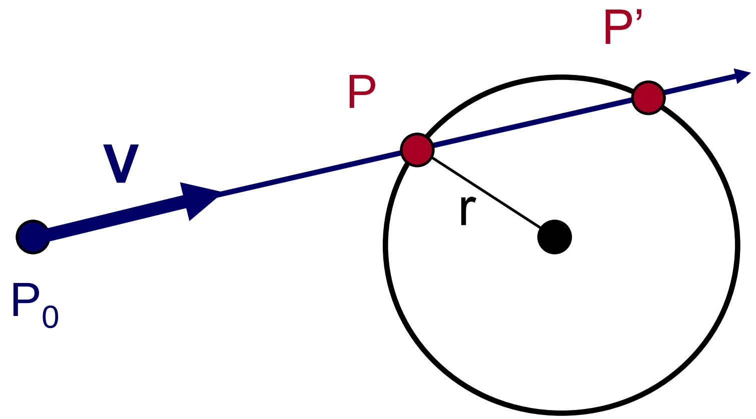
$$at^2 + bt + c = 0$$

where:

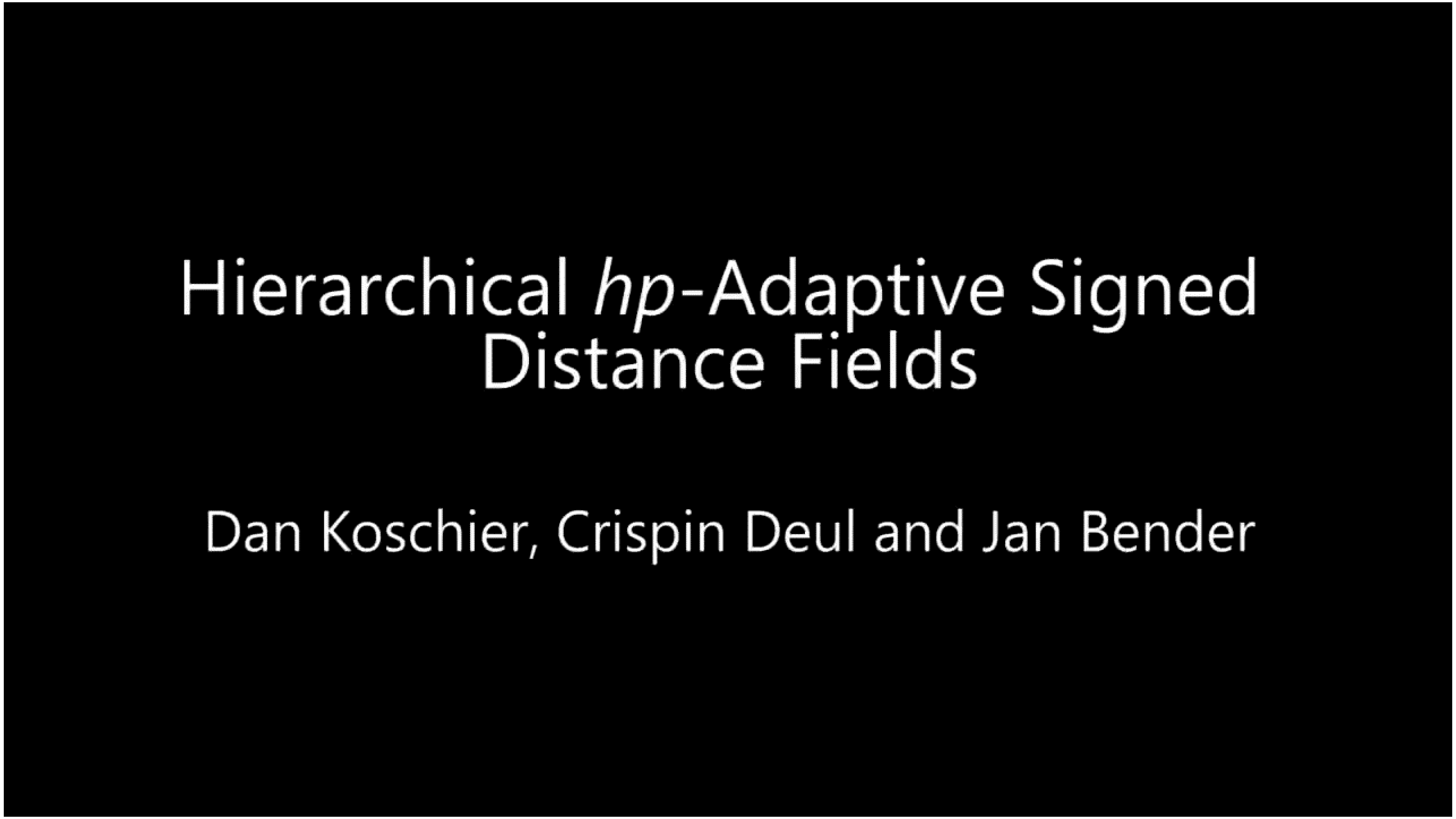
$$a = 1$$

$$b = 2\, V \bullet (P_0 - O)$$

$$c = |P_0 - C|^2 - r^2 = 0$$



Example: Simulation → Intersection Computation

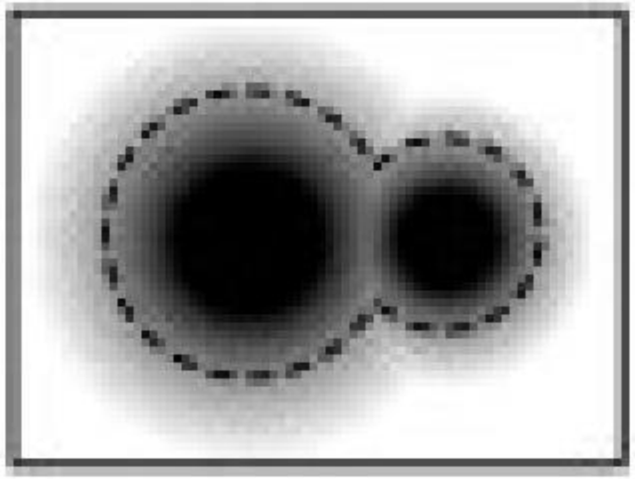


Hierarchical hp-Adaptive Signed Distance Fields
[Koschier 2016]

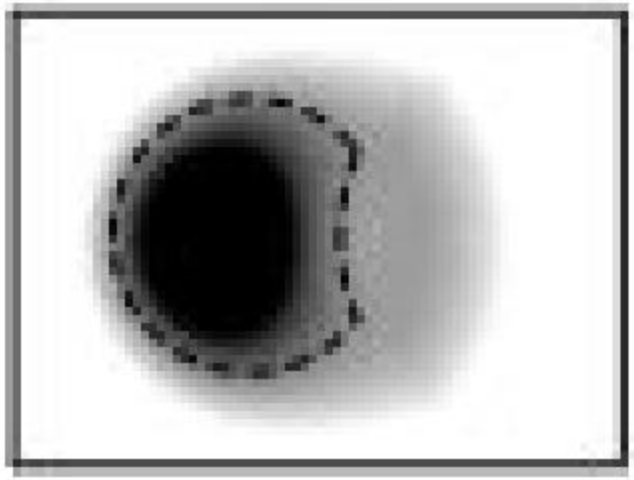
Implicit Surface Properties

(3) Efficient boolean operations (CSG – later in this lecture)

- How would you implement:
  Union? Intersection? Difference?



Union



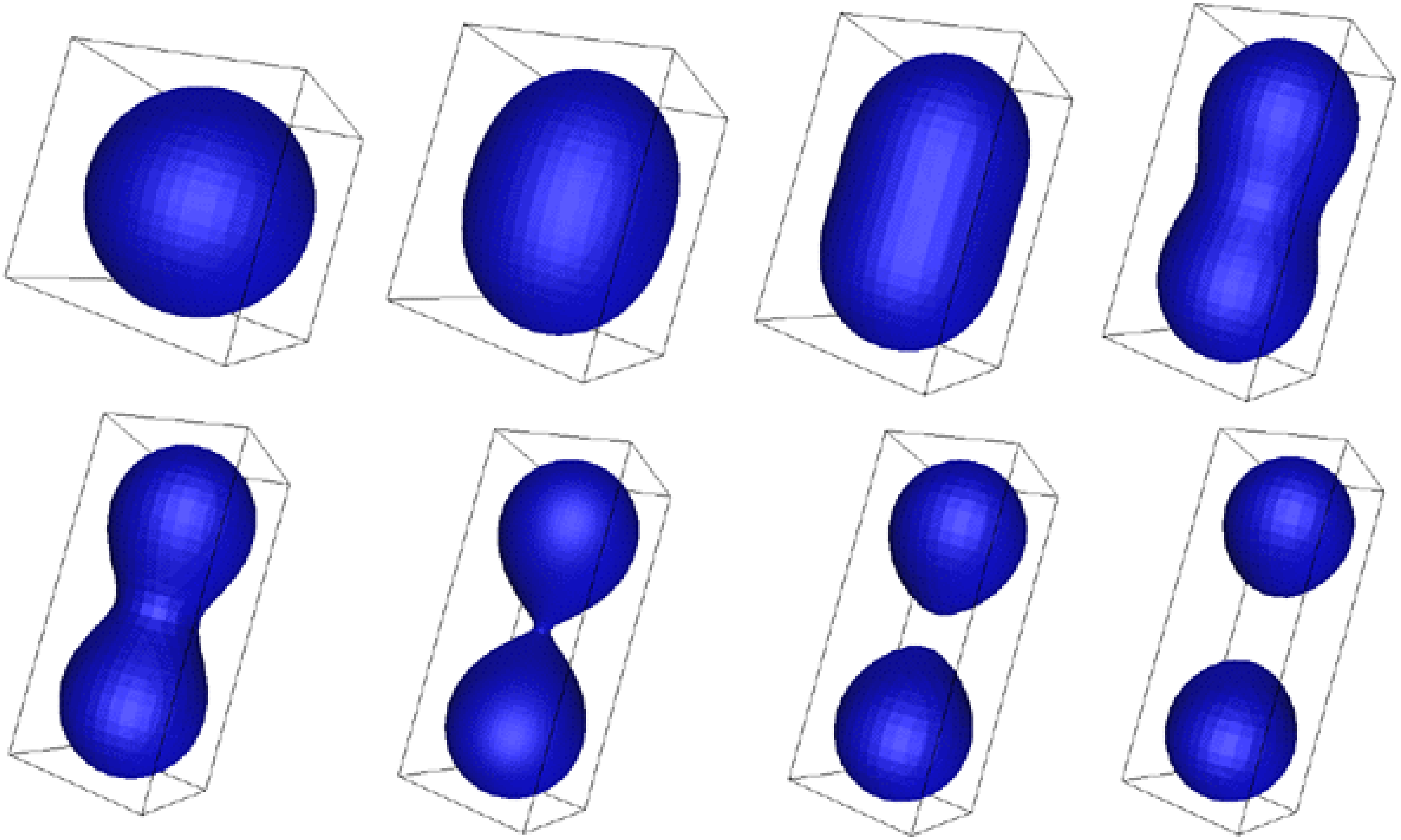
Difference

Bloomenthal

Implicit Surface Properties

(4) Efficient **topology changes!**

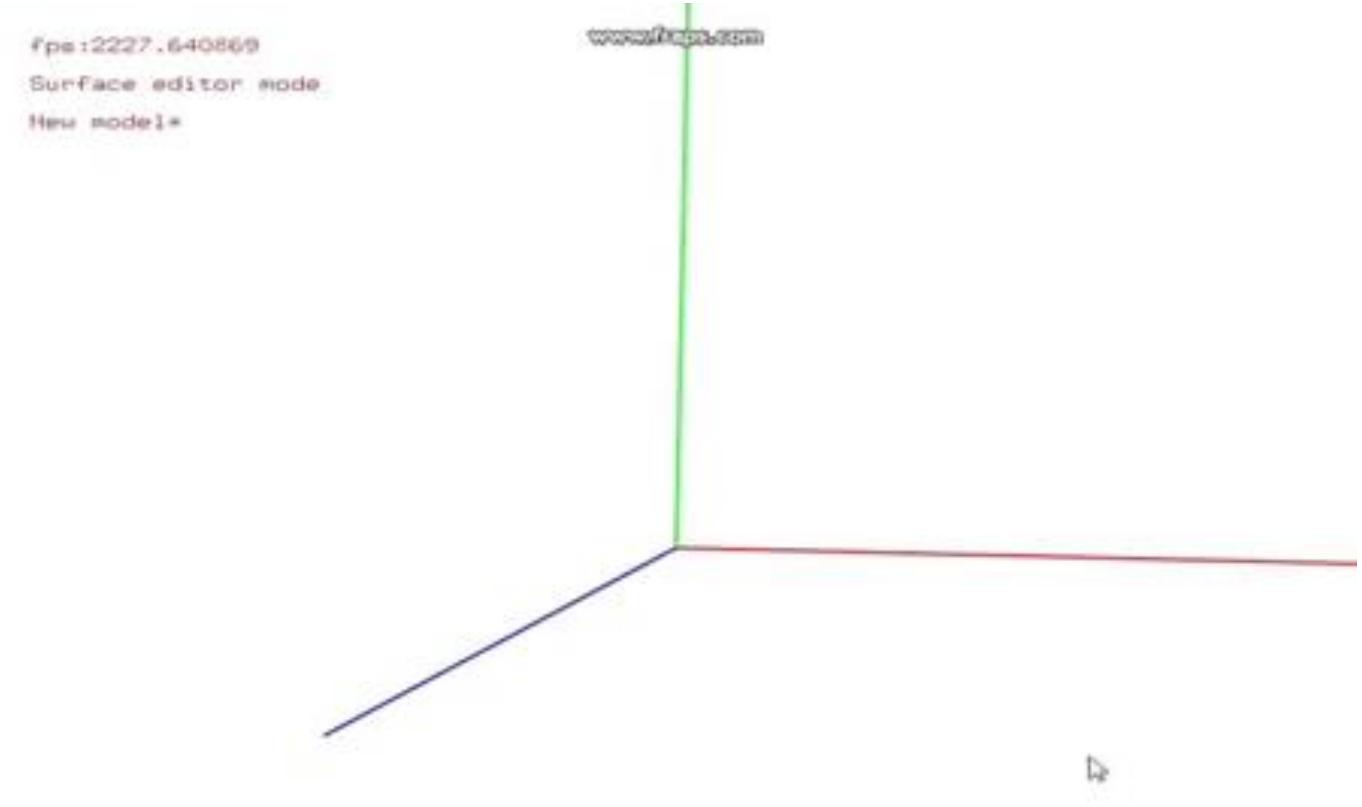
- Surface is not represented explicitly!



Bourke

Example: Modeling

[olivelarouille on Youtube]

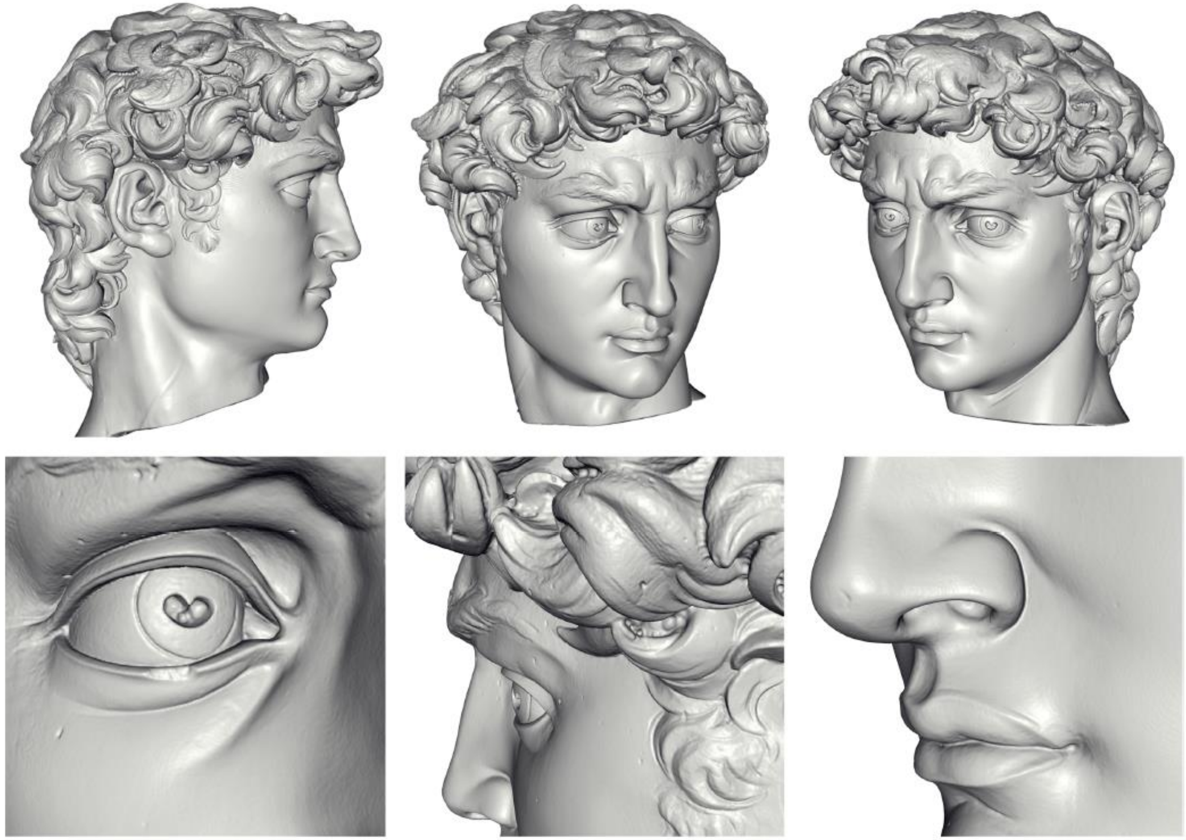


Implicit Surface Properties

(5) Computations in the volume

- Allows for continuity and smoothness
- Suitable for tasks such as reconstruction

1G sample points → 8M triangles



Poisson Surface Reconstruction [Kazhdan 06]

Comparison to Parametric Surfaces

- Implicit
  - Efficient intersections & topology changes
- Parametric
  - Efficient “***marching***” along surface & rendering

Implicit Surface Representations

- How do we define implicit function?
    - $f(x,y,z) = ?$

Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Voxels
  - Basis functions
  - Neural Networks

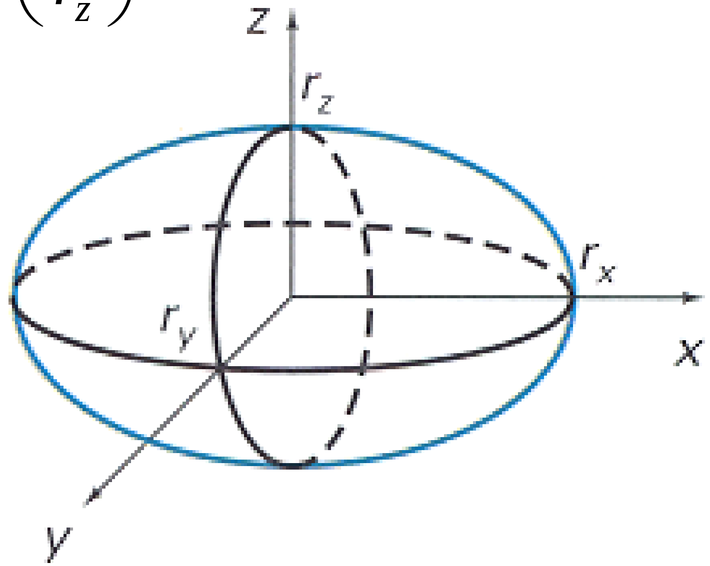
Implicit Surface Representations

- How do we define implicit function?
  - ➢ Algebraics
  - Voxels
  - Basis functions
  - Neural Networks

Algebraic Surfaces

- Implicit function is polynomial
  - $f(x,y,z)=ax^d+by^d+cz^d+dx^{d-1}y+ex^{d-1}z+fy^{d-1}x+\ldots$

$$f(x,y,z)=\left(\frac{x}{r_x}\right)^2+\left(\frac{y}{r_y}\right)^2+\left(\frac{z}{r_z}\right)^2-1$$



H&B Figure 10.10

Algebraic Surfaces

- Most common form: quadrics
  - $f(x,y,z)=ax^2+by^2+cz^2+2dxy+2eyz+2fxz+2gx+2hy+2jz+k$

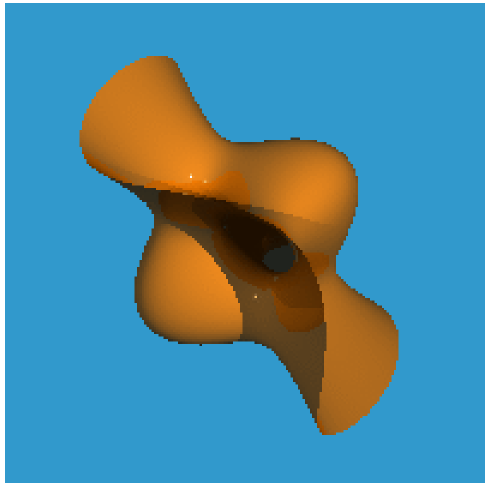
- Examples
  - Sphere
  - Ellipsoid
  - Paraboloid
  - Hyperboloid

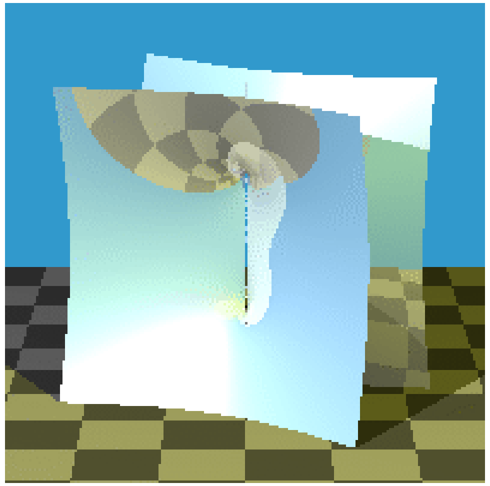
http://tutorial.math.lamar.edu/Classes/CalcIII/QuadricSurfaces.aspx

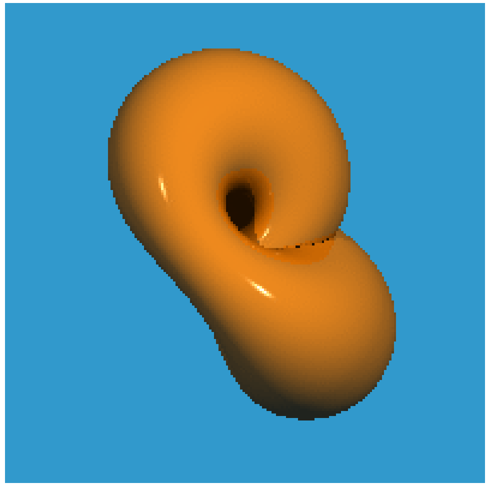
Menon

Algebraic Surfaces

- Higher degree algebraics

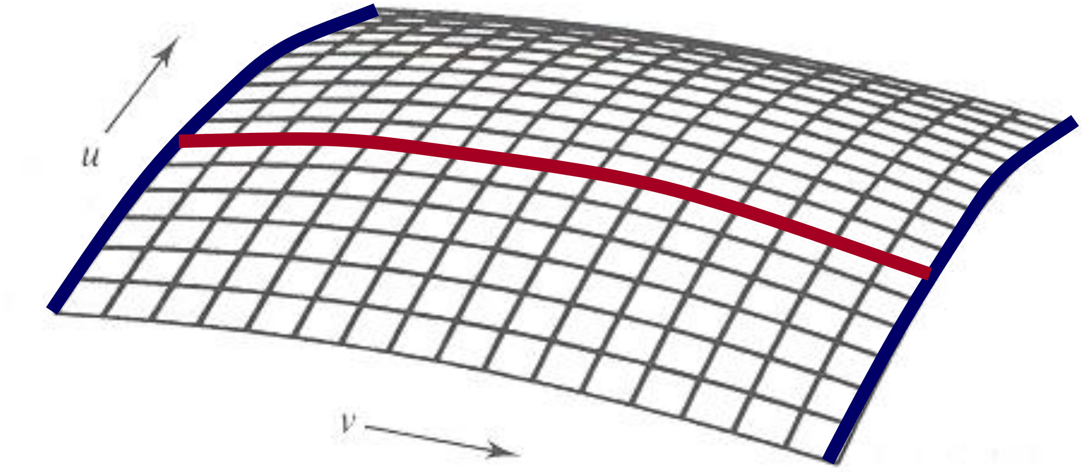

Cubic


Quartic


Degree six

Algebraic Surfaces

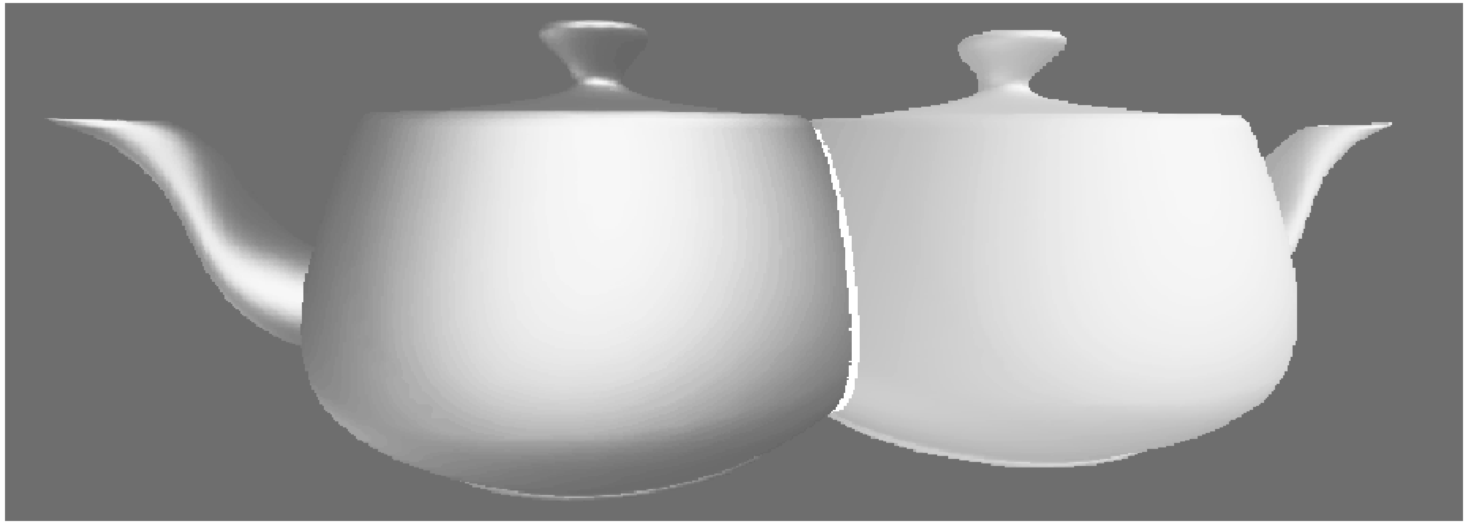
- Equivalent parametric surface
  - Tensor product patch of degree m and n curves yields algebraic function with degree 2mn



Bicubic patch has degree 2*3*3 = 18!

Algebraic Surfaces

- Intersection
  - Intersection of degree m and n algebraic surfaces yields curve with degree mn
  - Computationally hefty!



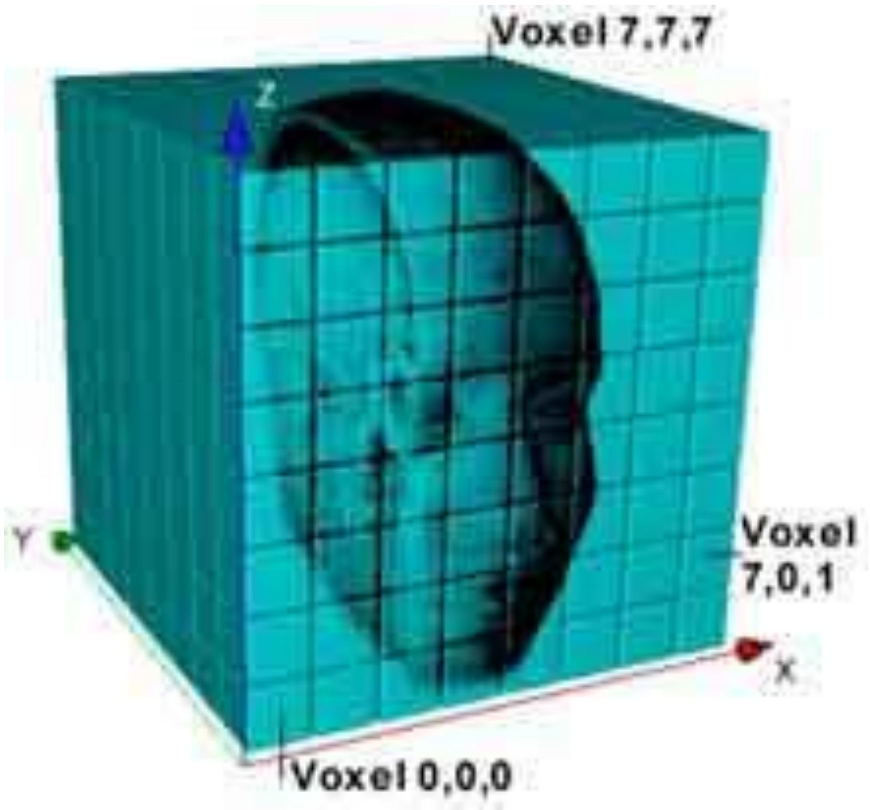
Intersection of bicubic patches has degree 18*18 = 324!

Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - ➢Voxels
  - Basis Functions
  - Neural Networks

Voxels

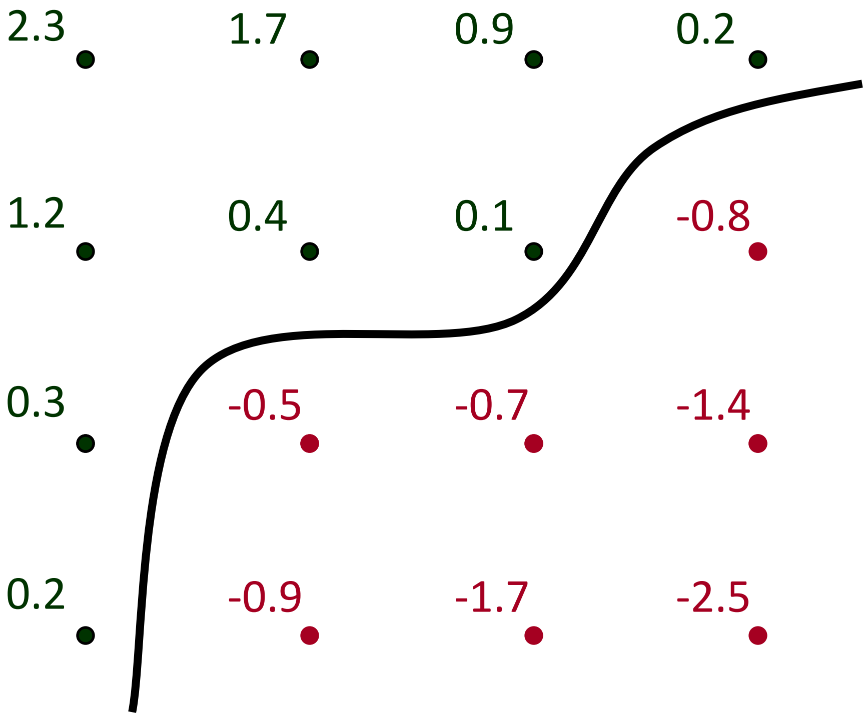
- Regular array of 3D samples (like image)
  - Samples are called *voxels* (“**vo**lume pi**xels**”)



www.volumegraphics.com

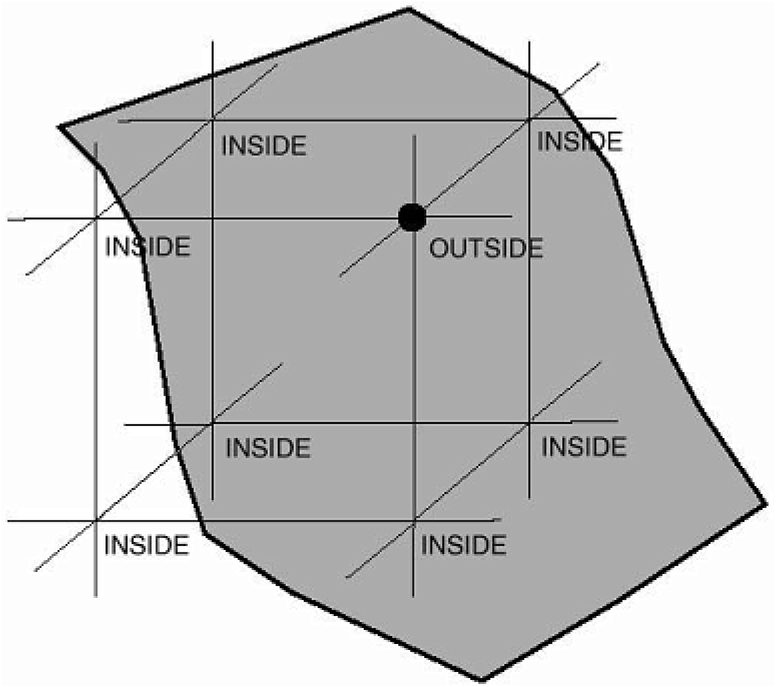
Voxels

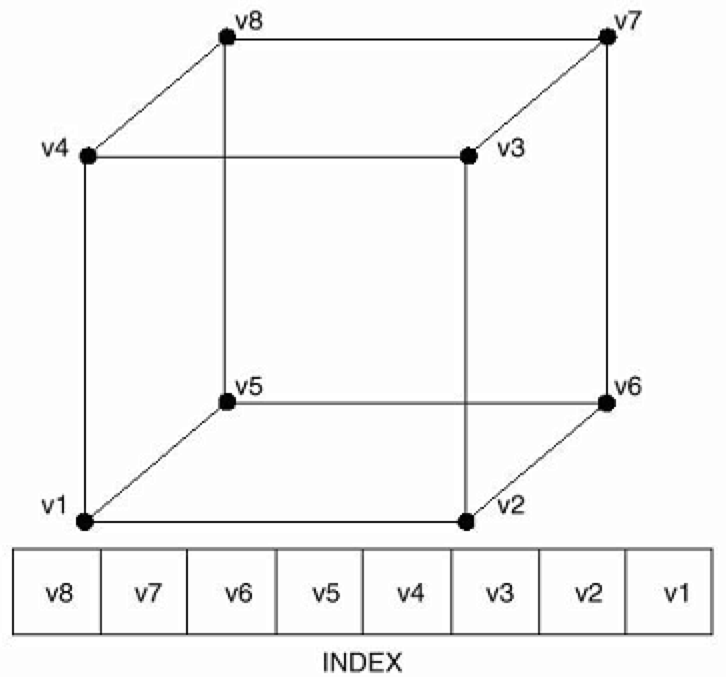
- Regular array of 3D samples (like image)
  - Applying reconstruction filter (e.g. trilinear) yields f(x,y,z)
  - Isosurface at f(x,y,z) = 0 defines surface



Voxels

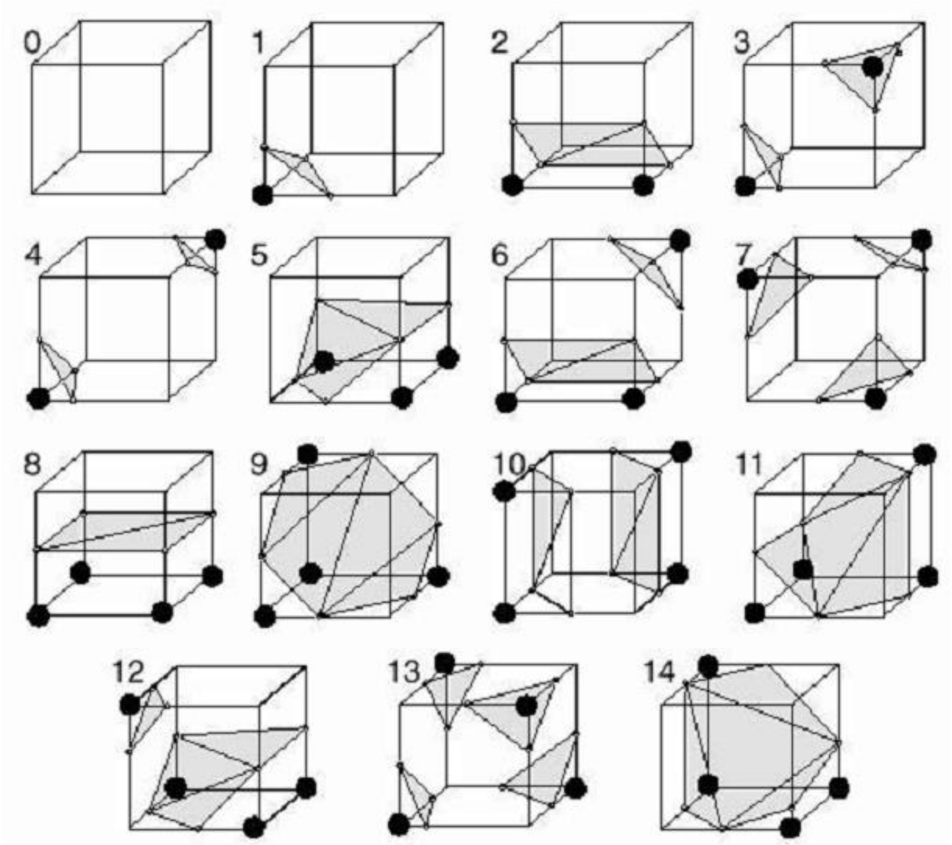
- Iso-surface extraction algorithm
  - e.g., Marching cubes

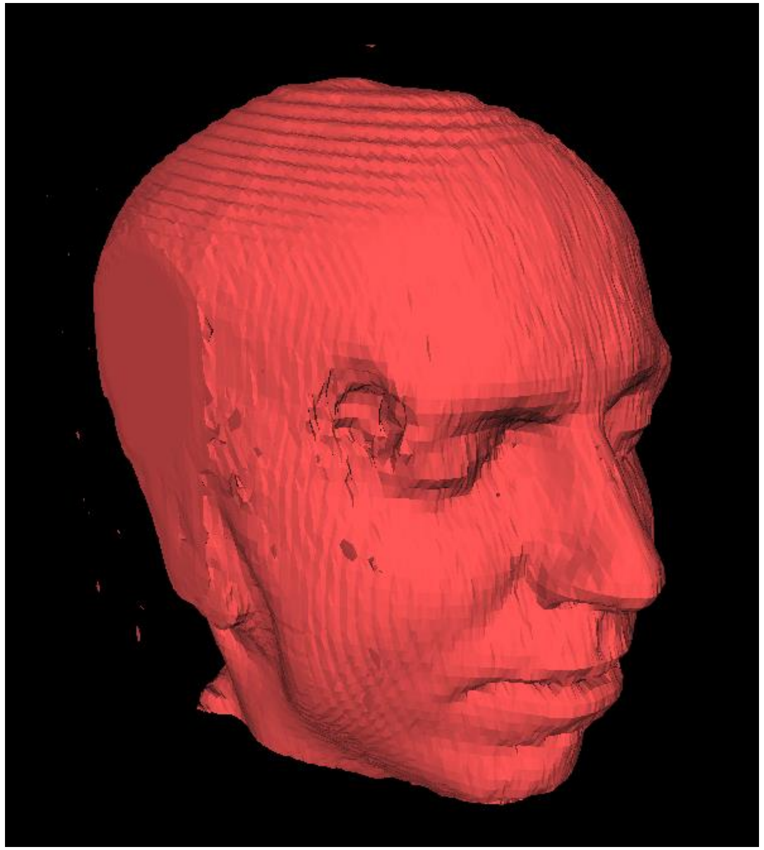




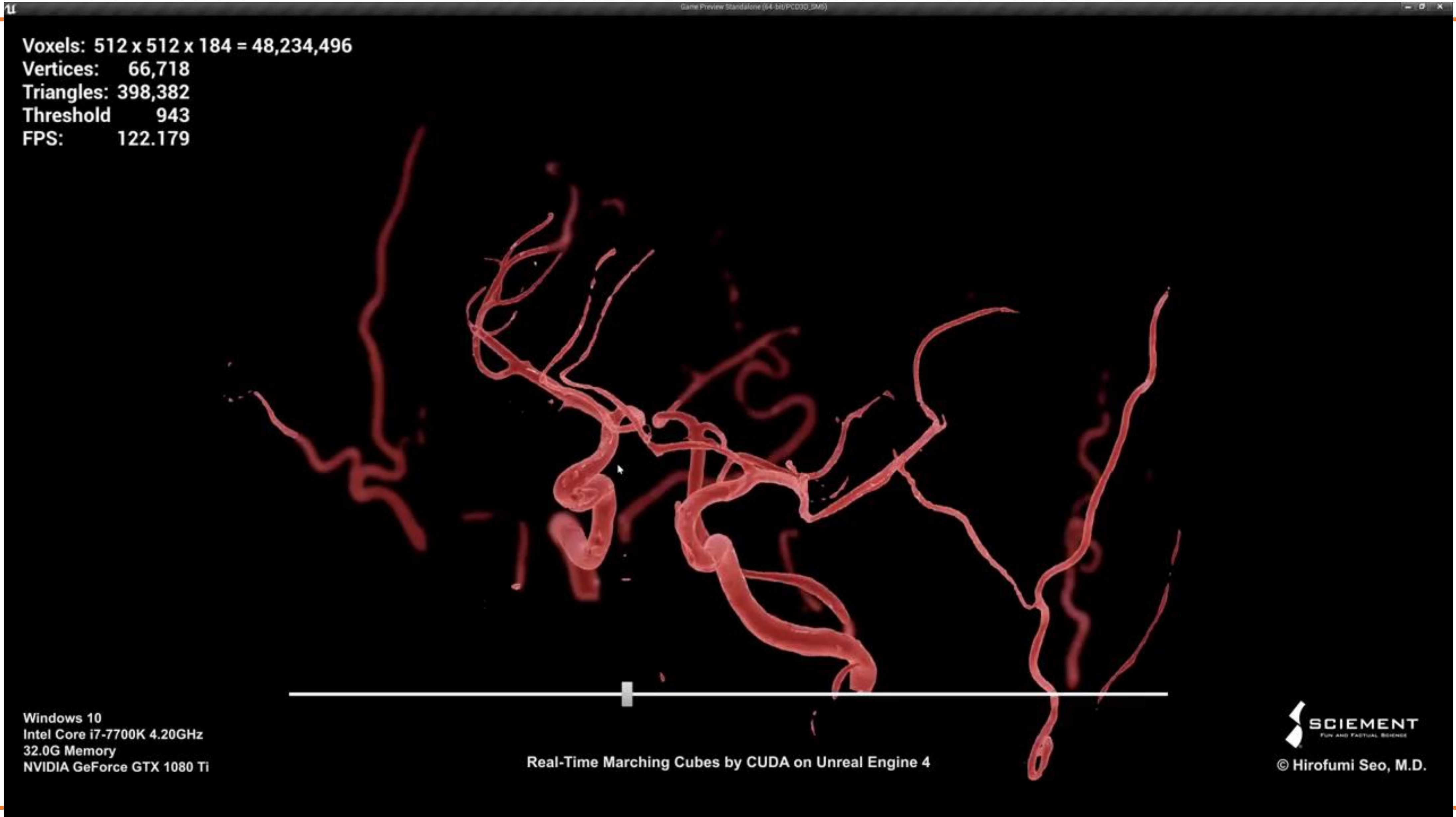
Voxels

- Iso-surface extraction algorithm
  - e.g., Marching cubes (15 cases)



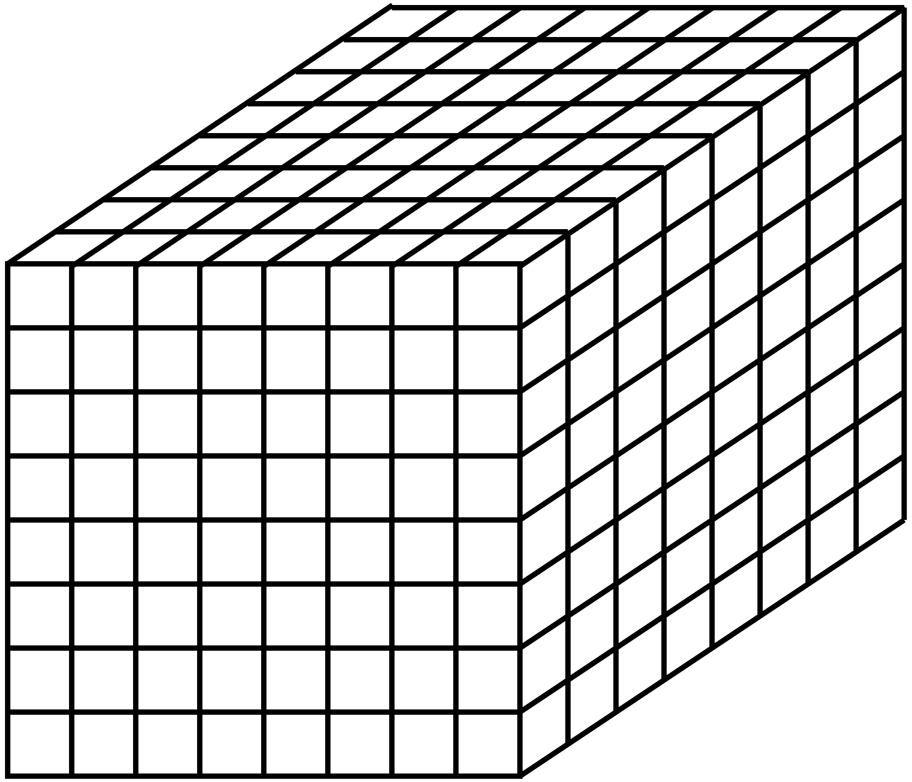


Example: Marching Cubes



Voxel Storage

- $O(n^3)$ storage for n x n x n grid
  - 1 billion voxels for 1000 x 1000 x 1000



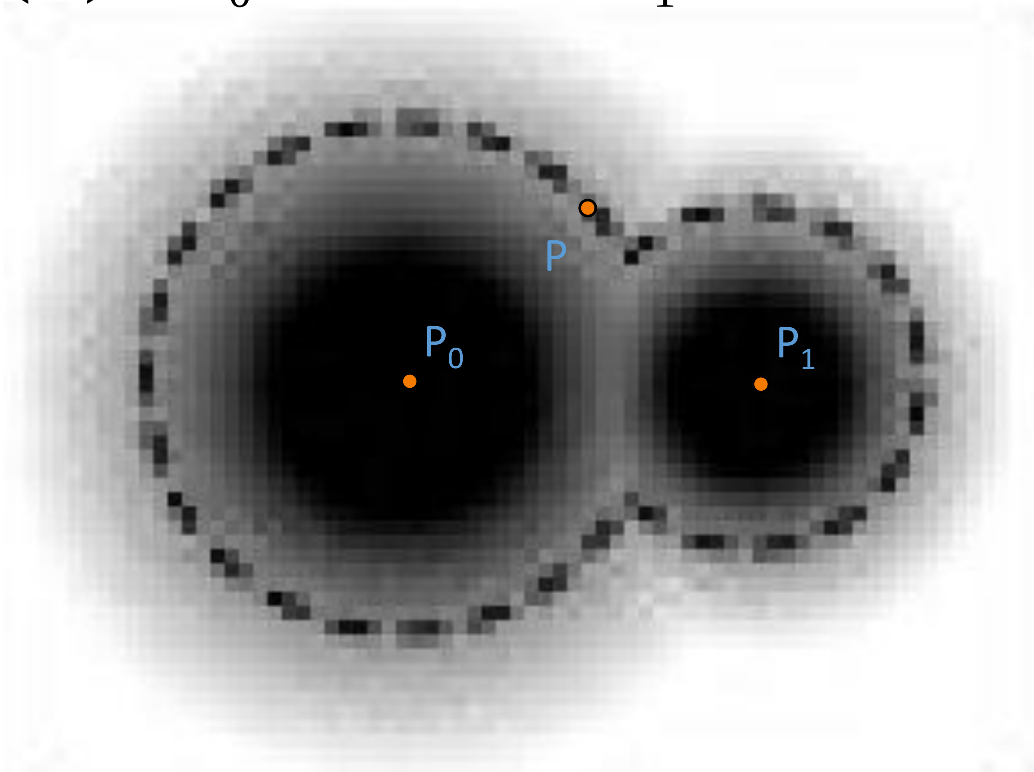
Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Voxels
  - ➢ Basis functions
  - Neural Networks

Blobby Models

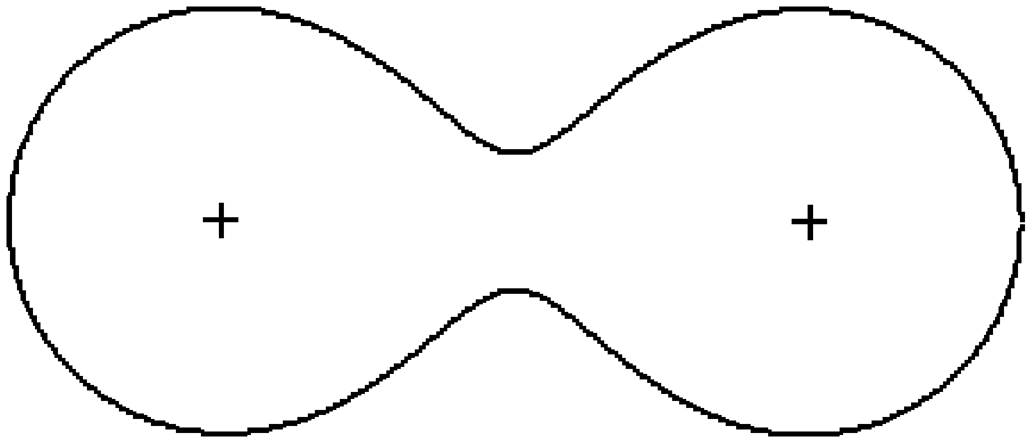
- Implicit function is sum of Gaussians

$$f(P) = a_0 e^{-b_0 d(P,P_0)^2} + a_1 e^{-b_1 d(P,P_1)^2} + \cdots - \tau$$



Blobby Models

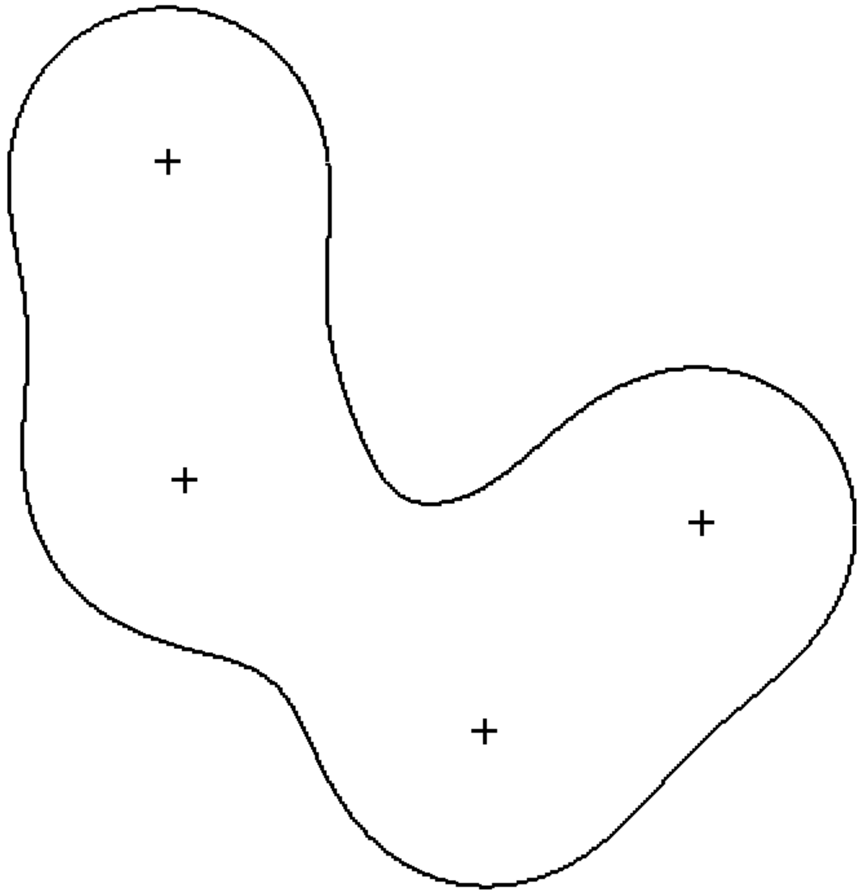
- Sum of two blobs



Turk

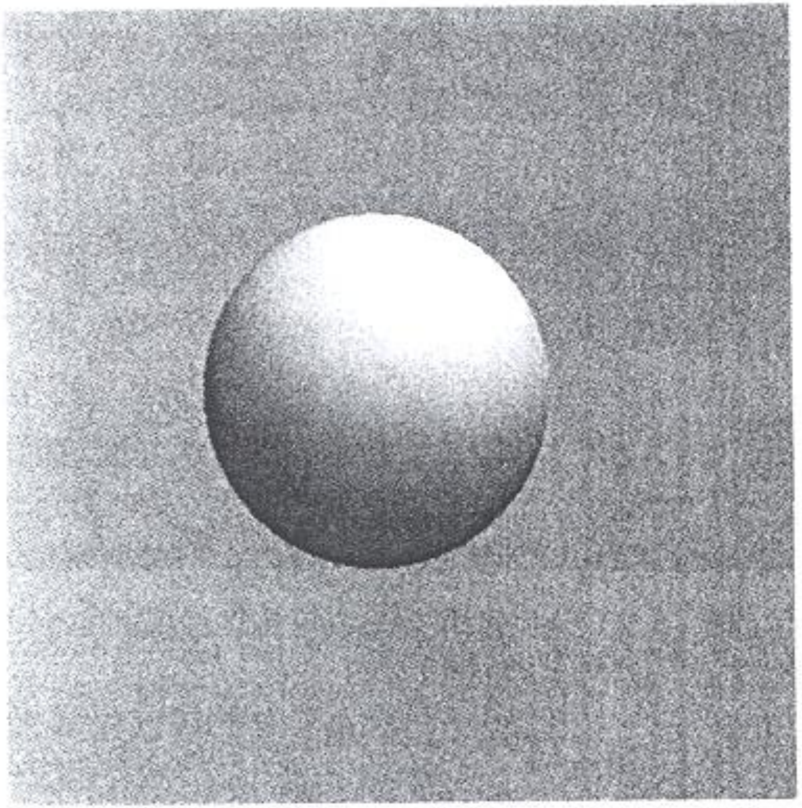
Blobby Models

- Sum of four blobs

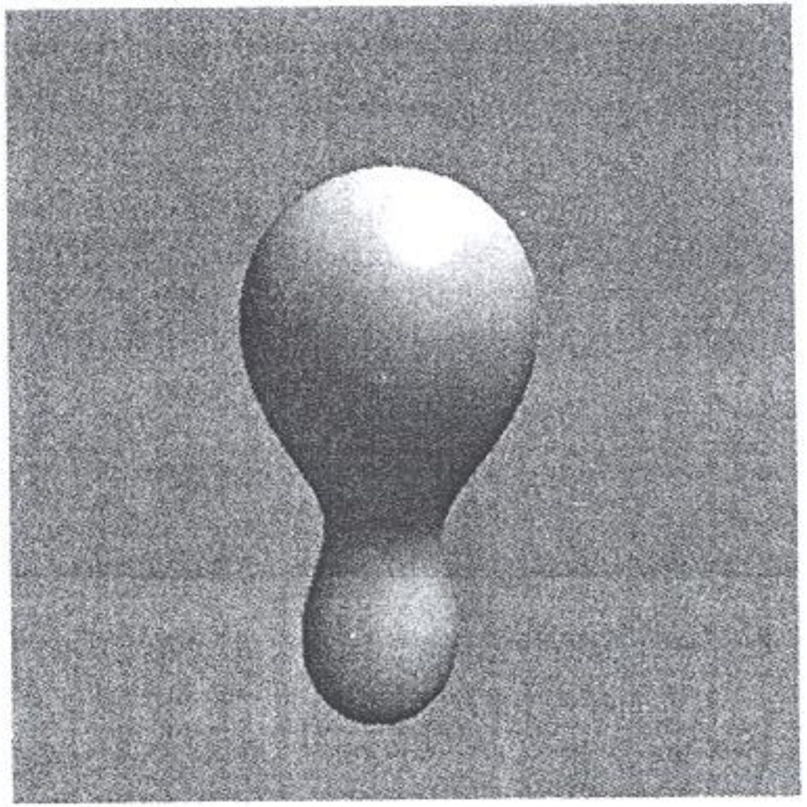


Turk

3D Blobby Model of Face

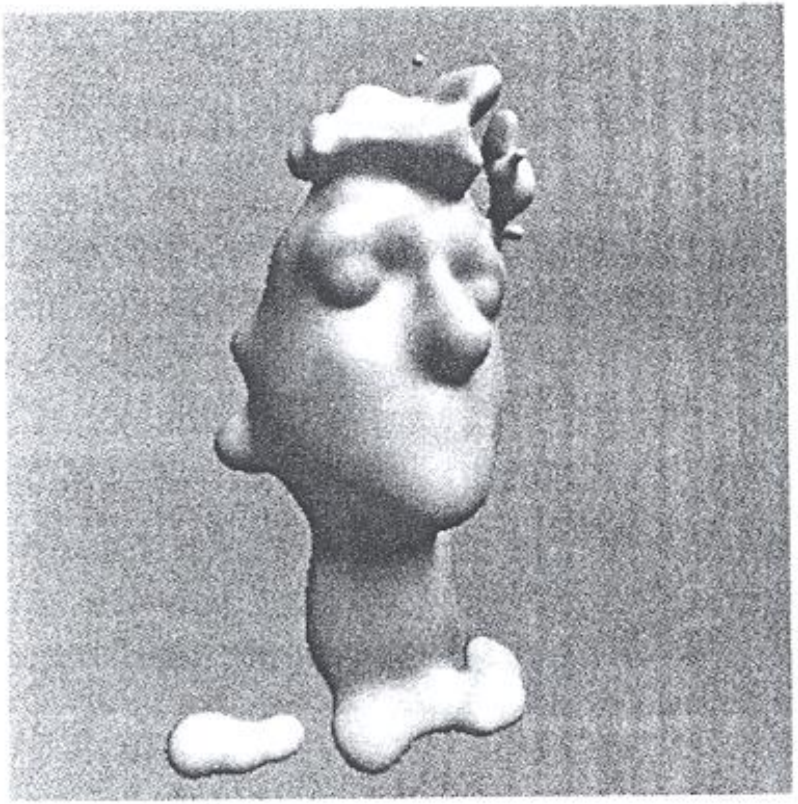


(a) $N = 1$

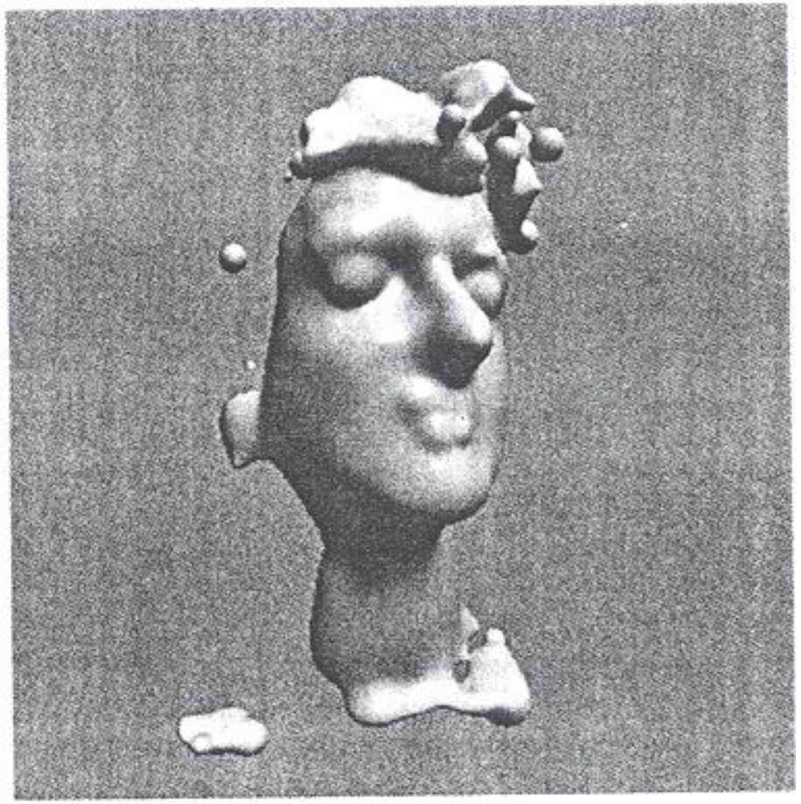


(b) $N = 2$

3D Blobby Model of Face

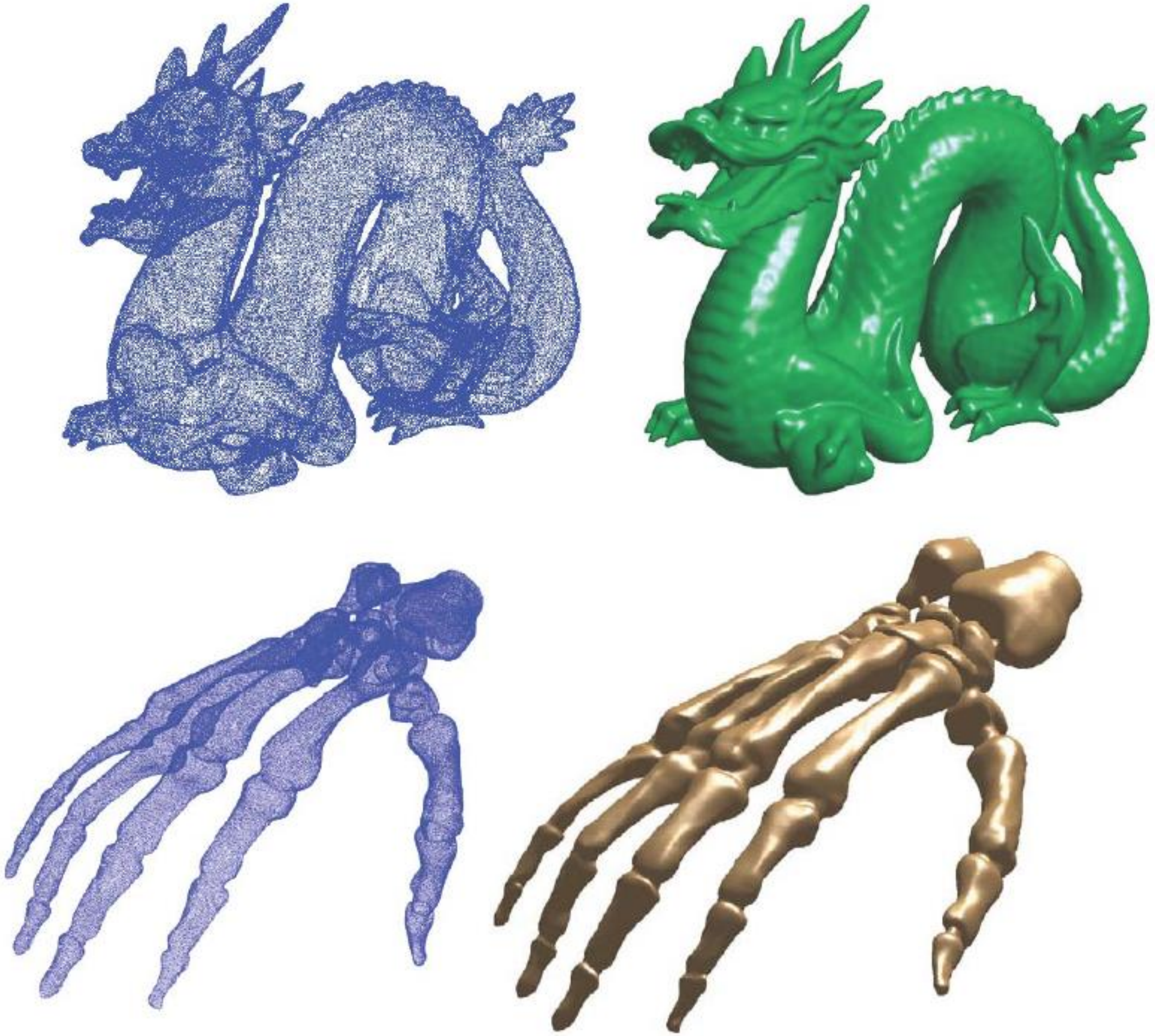


(e) $N = 70$



(f) $N = 243$

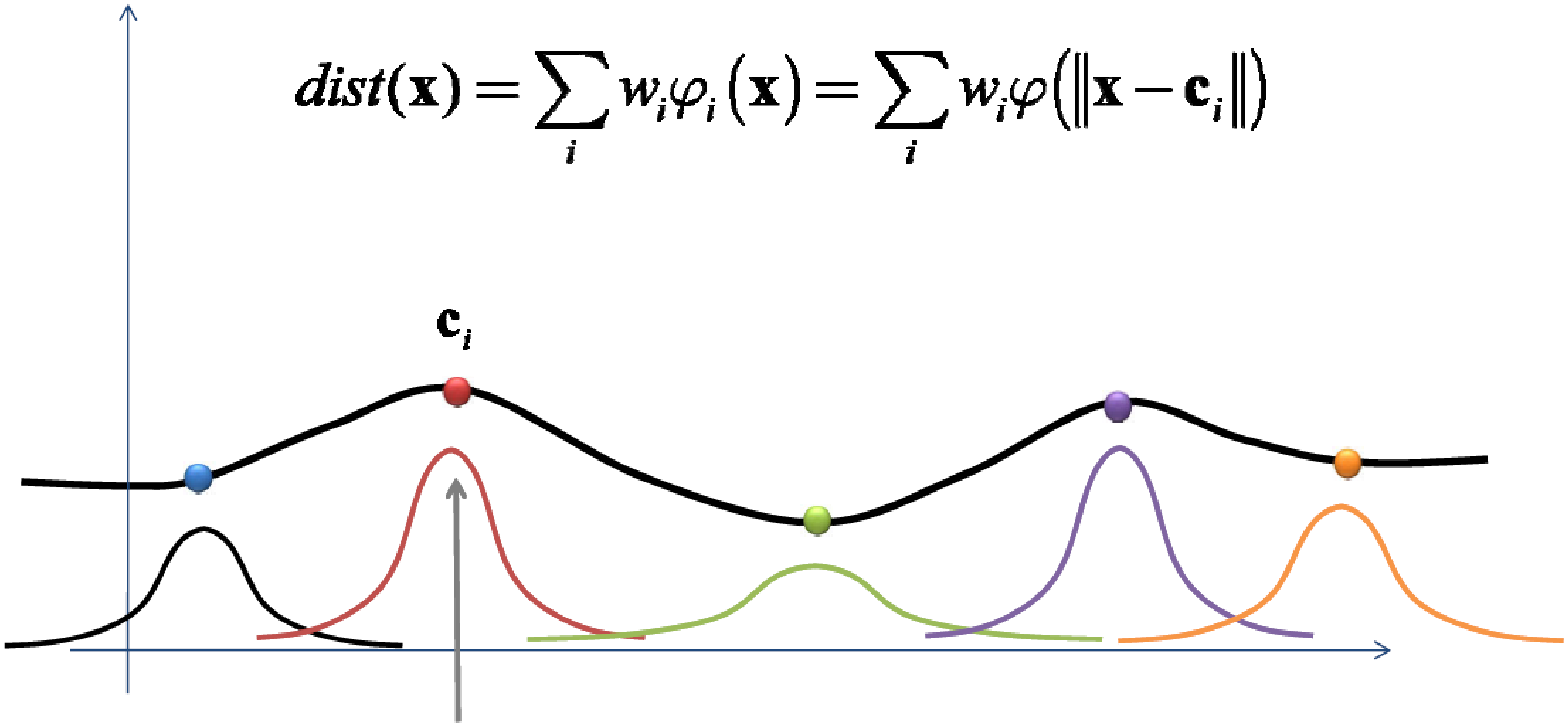
Reconstruction from Point Sets



Reconstruction from Point Sets

- Implicit function is sum of basis functions

$$dist(\mathbf{x}) = \sum_i w_i \varphi_i(\mathbf{x}) = \sum_i w_i \varphi(\|\mathbf{x} - \mathbf{c}_i\|)$$



Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Voxels
  - Basis functions
  - ➢Neural Networks

The problem of novel view interpolation

Inputs: sparsely sampled images of scene

Outputs: new views of same scene

NeRF (neural radiance fields):

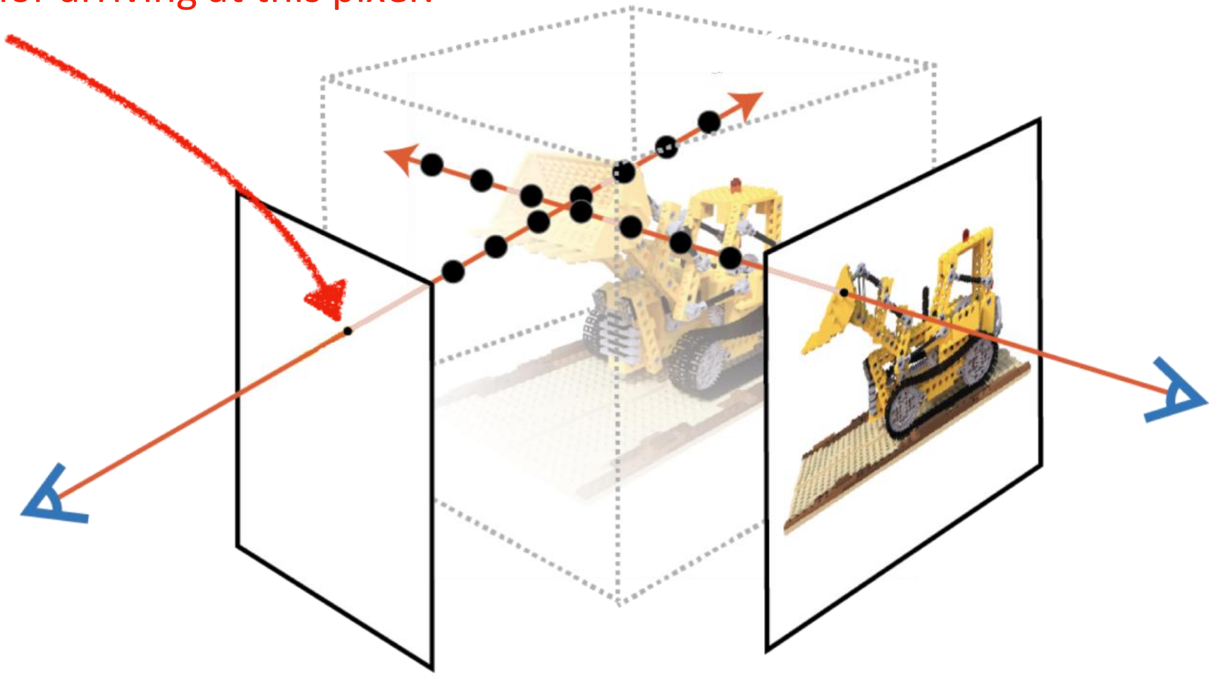
Neural networks as a volume representation, using volume rendering to do view synthesis.

$$(x, y, z, \theta, \phi) \rightarrow color, opacity$$

Neural Volumetric Rendering

Neural Volumetric Rendering

What's the radiance/color arriving at this pixel?

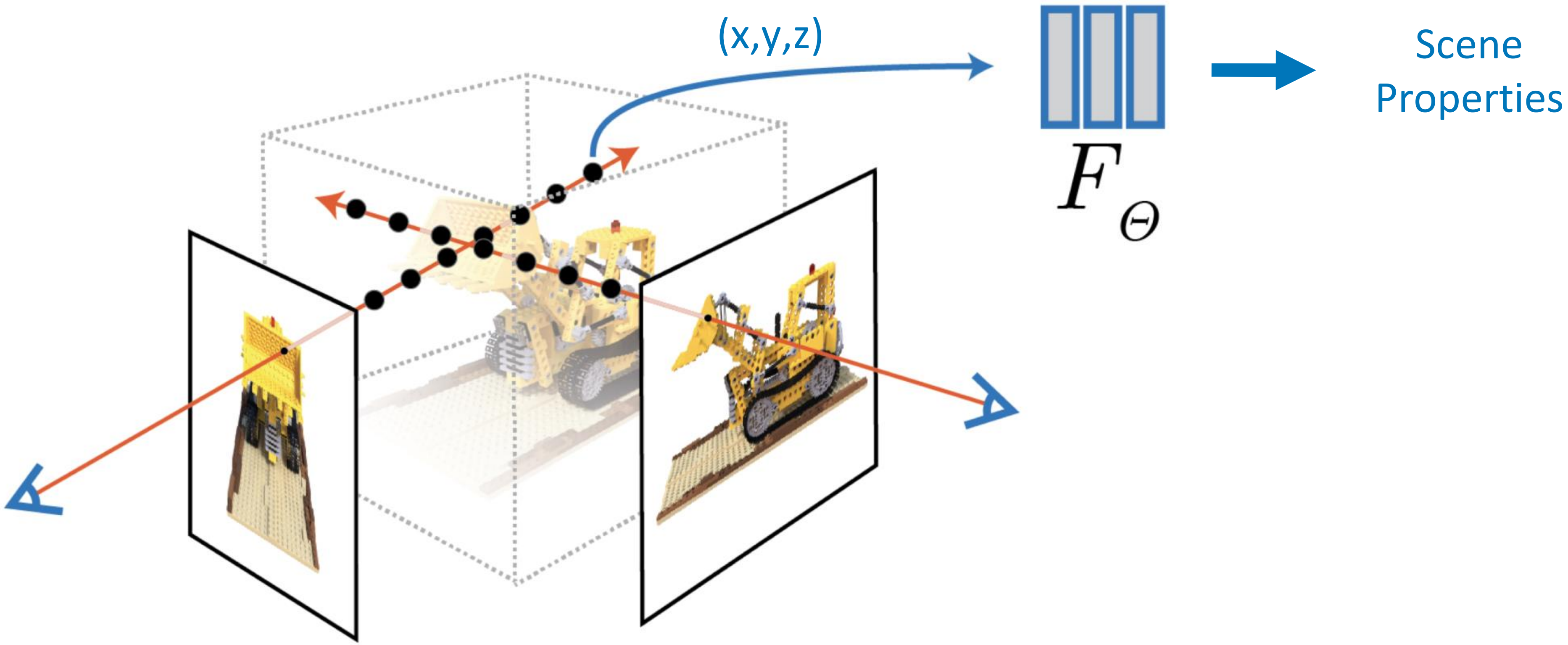


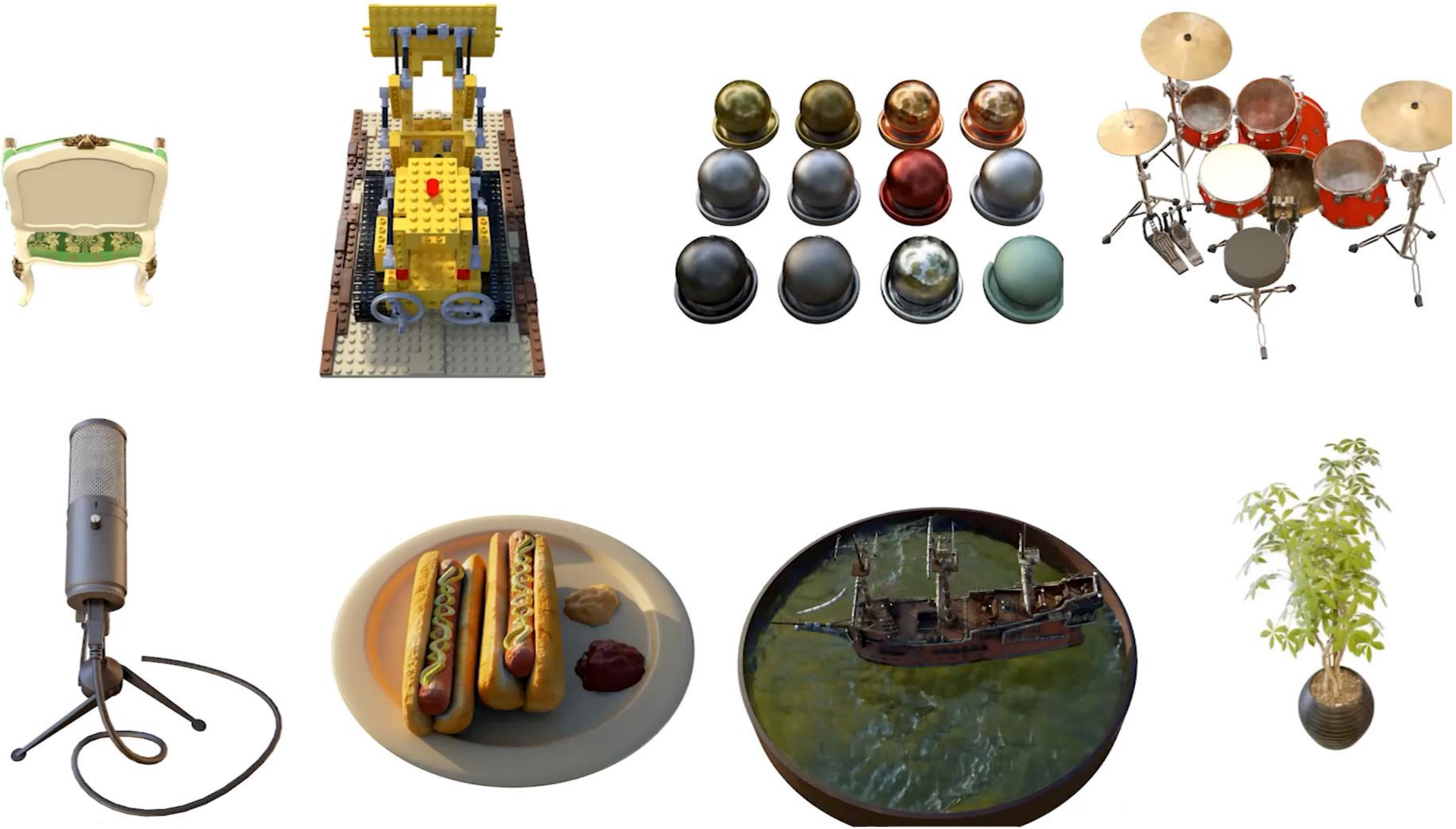
Neural Volumetric Rendering

- “Soft” volumetric functions better suited for gradient-based optimization

Neural Volumetric Rendering

- (Coordinate-based) neural network represents scene as continuous function



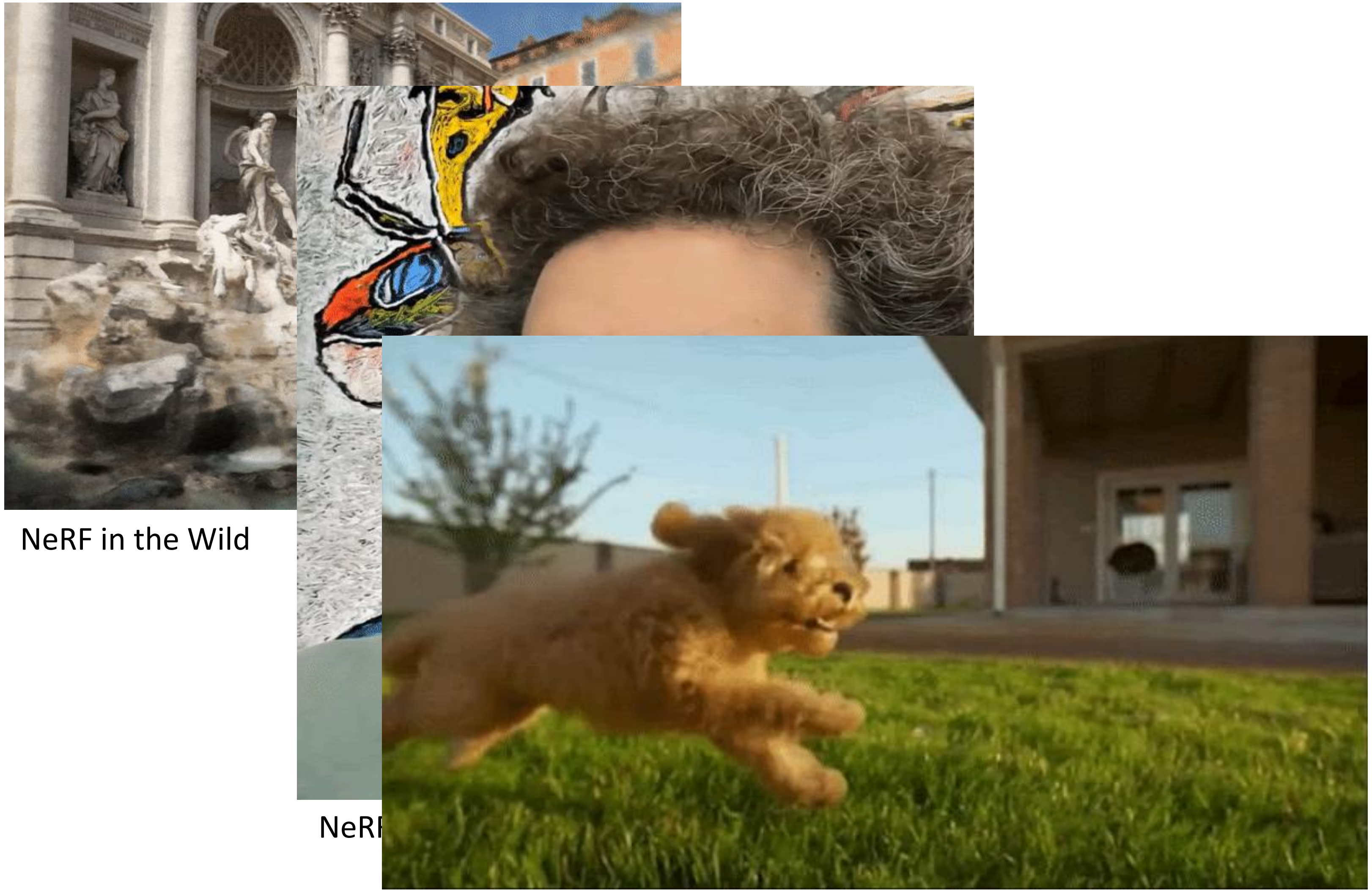


NeRF encodes detailed scene geometry with occlusion effects


NeRF in the Wild, Martin-Brualla et al.

NeRF in the Wild, M

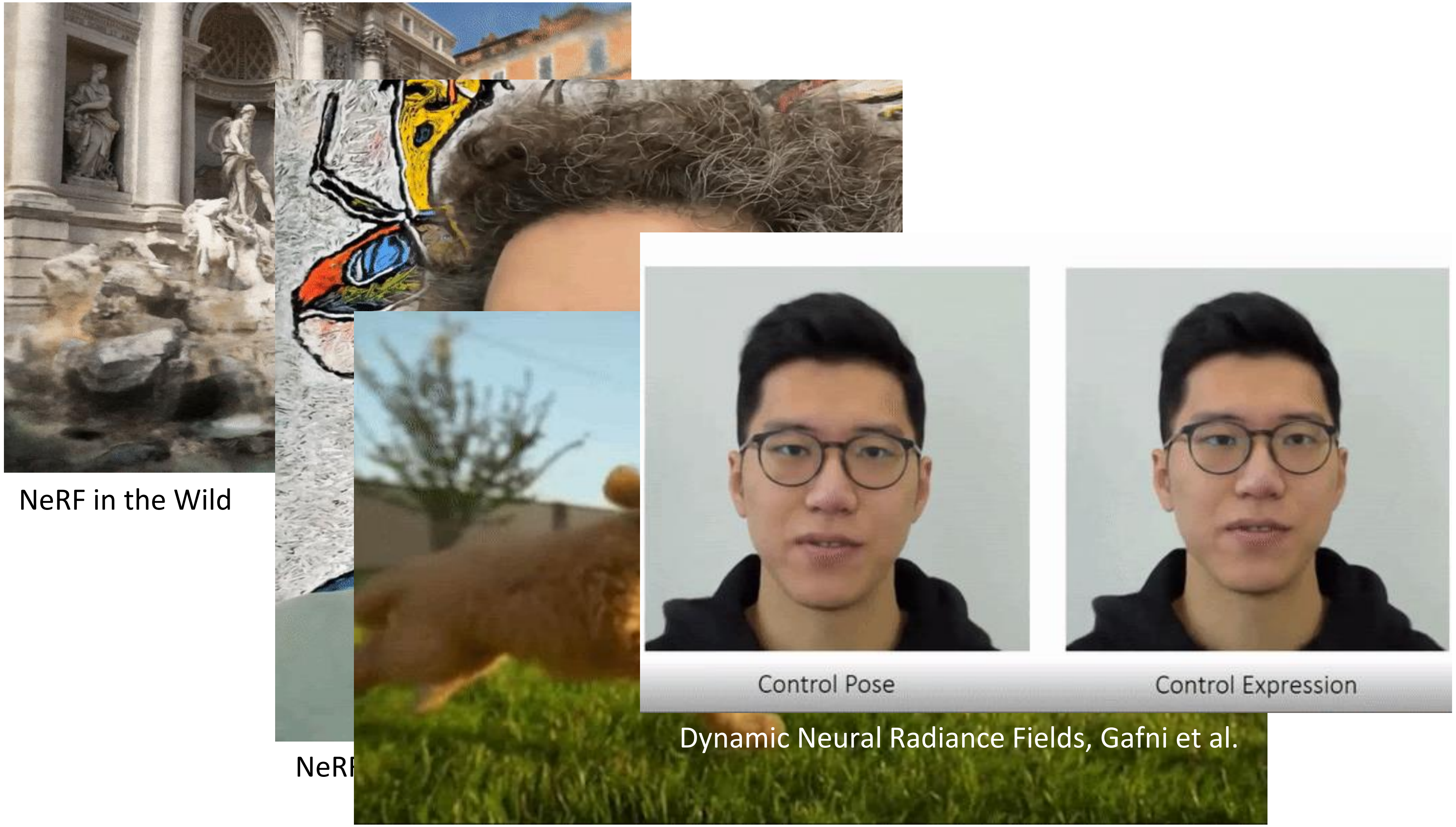
NeRFies, Park et al.



NeRF in the Wild

NeRF

Neural Scene Flow Fields, Li et al.



NeRF in the Wild

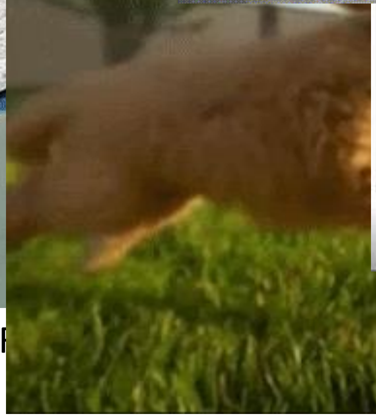
NeRF

Dynamic Neural Radiance Fields, Gafni et al.

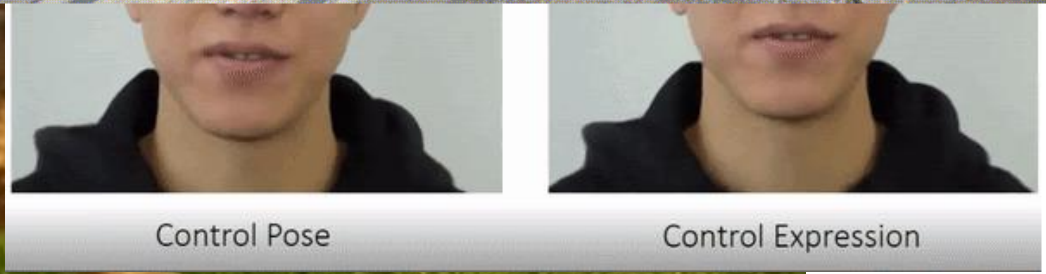
Neural Scene Flow Fields, Li et al.

NeRF in the Wild

NeR



Neural Scene Flow Fields, Li et al.



Dynamic Neural Radiance Fields, Gafni et al.

Neural Scene Graphs, Ost et al.

Implicit Surface Summary

- Advantages:
  - Easy to test if point is on surface
  - Easy to compute intersections/unions/differences
  - Easy to handle topological changes
- Disadvantages:
  - Indirect specification of surface
  - Hard to describe sharp features
  - Hard to enumerate points on surface
    - Slow rendering

Summary

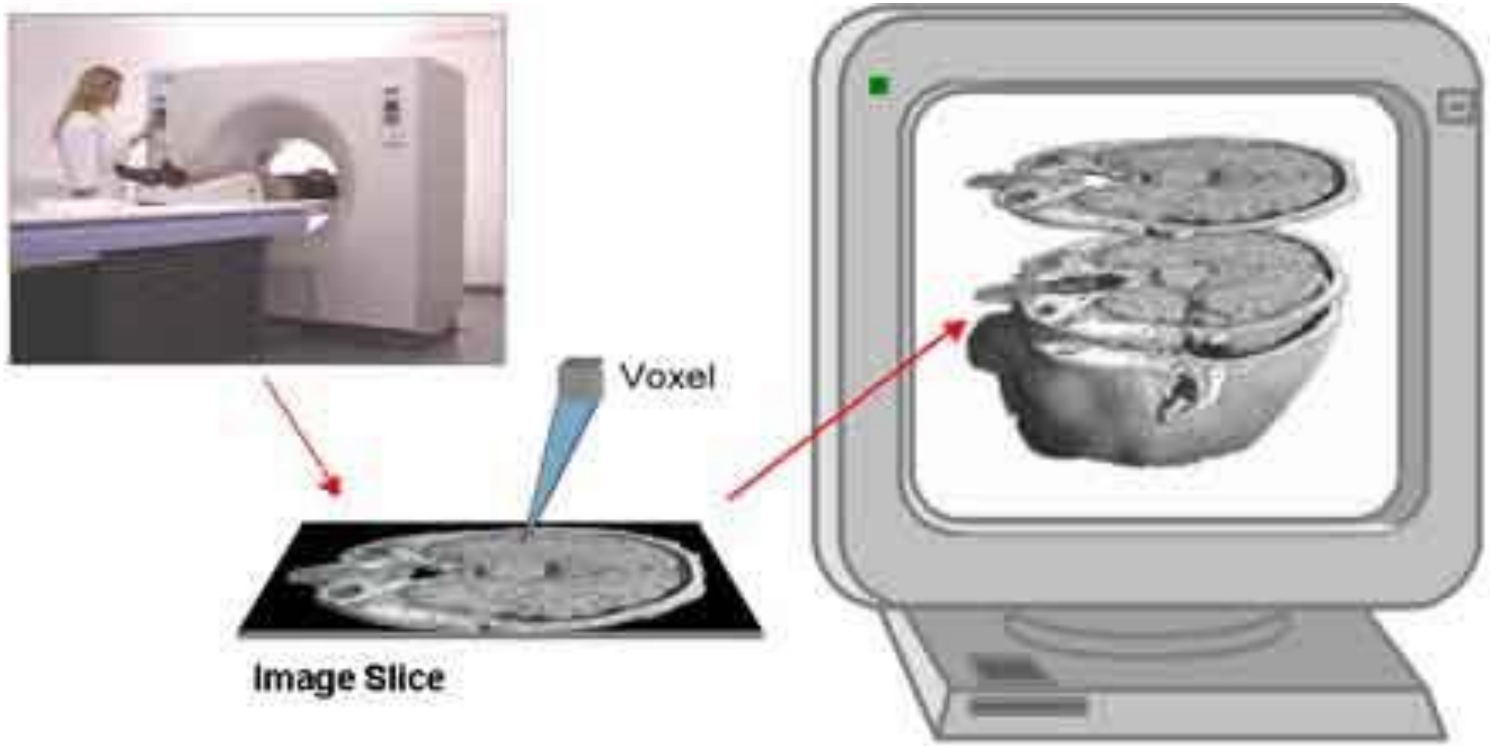
Feature	Polygonal Mesh	Implicit Surface	Parametric Surface	Subdivision Surface
Accurate	No	Yes	Yes	Yes
Concise	No	Yes	Yes	Yes
Intuitive specification	No	No	Yes	No
Local support	Yes	No	Yes	Yes
Affine invariant	Yes	Yes	Yes	Yes
Arbitrary topology	Yes	No	No	Yes
Guaranteed continuity	No	Yes	Yes	Yes
Natural parameterization	No	No	Yes	No
Efficient display	Yes	No	Yes	Yes
Efficient intersections	No	Yes	No	No

3D Object Representations

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

Solid Modeling

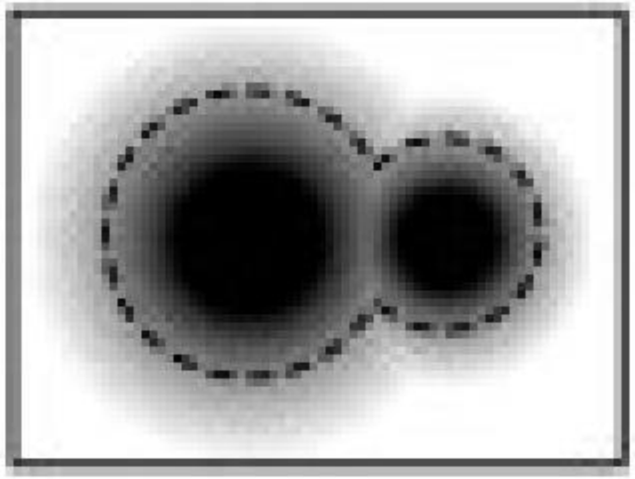
- Represent solid interiors of objects



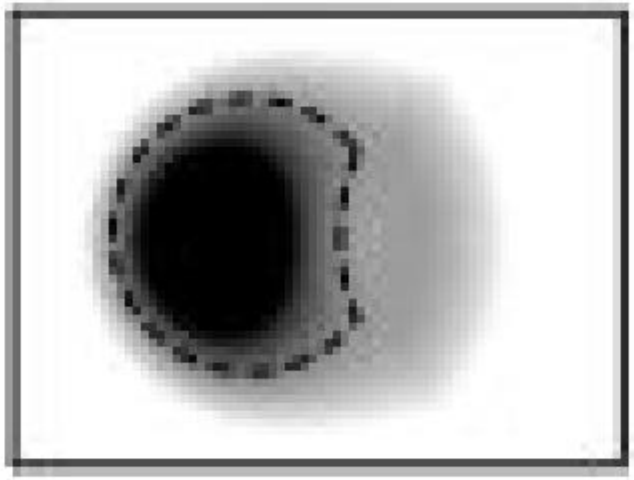
www.volumegraphics.com

Motivation

- Some operations are easier with solids
  - Example: union, difference, intersection



Union



Difference

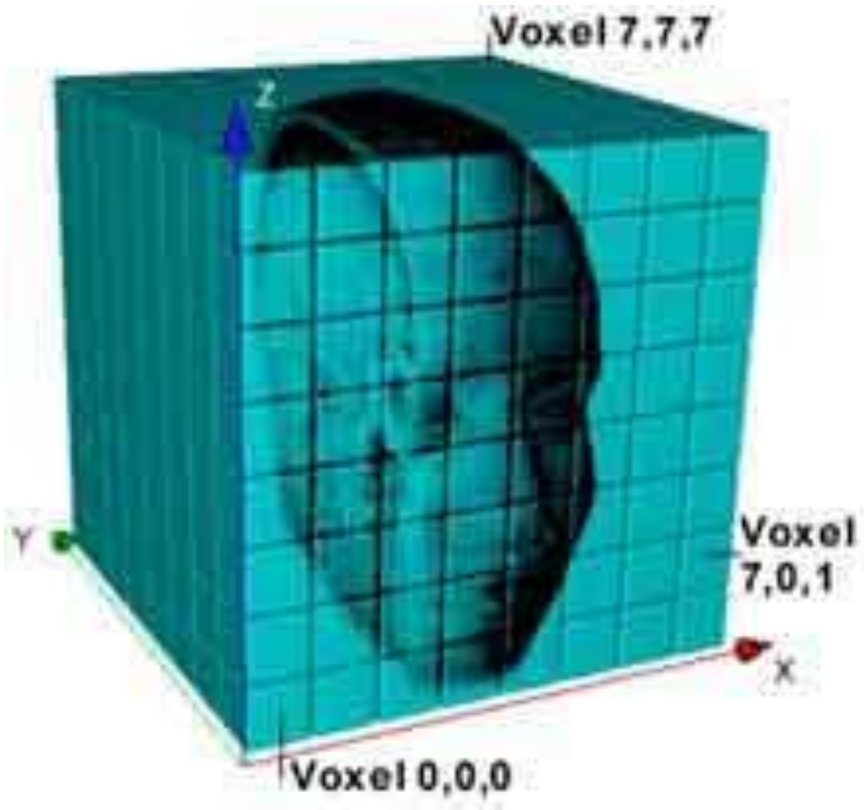
Bloomenthal

3D Object Representations

- Raw data
  - Range image
  - Point cloud
- Surfaces
  - Polygonal mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - ➢ Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Application specific

Return to Voxels

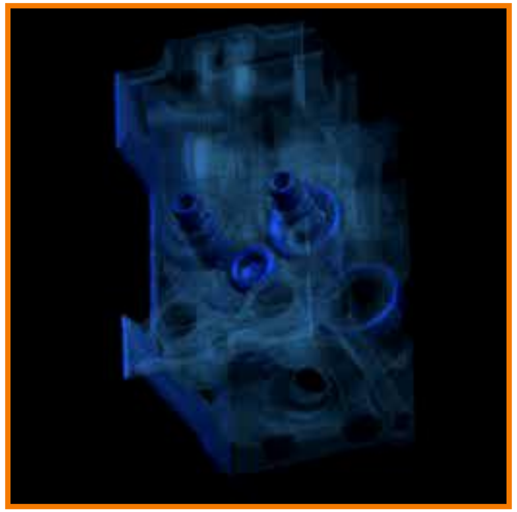
- Regular array of 3D samples (like image)



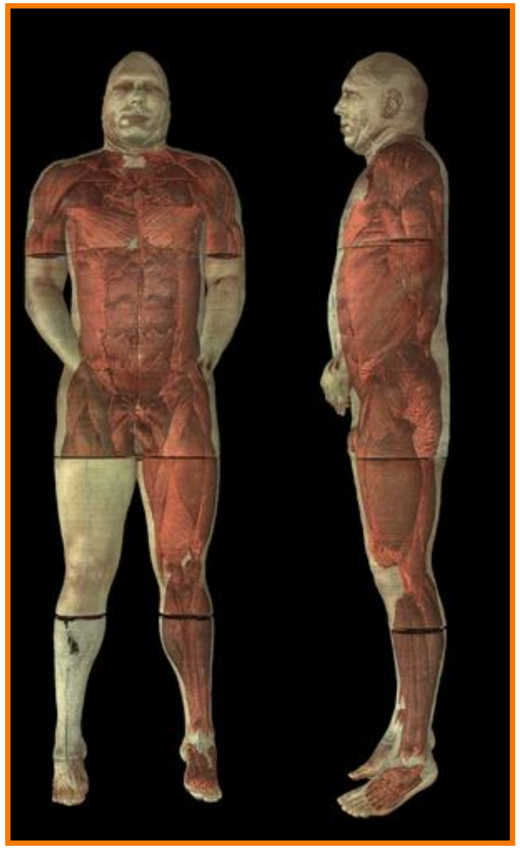
www.volumegraphics.com

Voxels

- Store properties of solid object with each voxel
  - Occupancy
  - Color
  - Density
  - Temperature
  - etc.



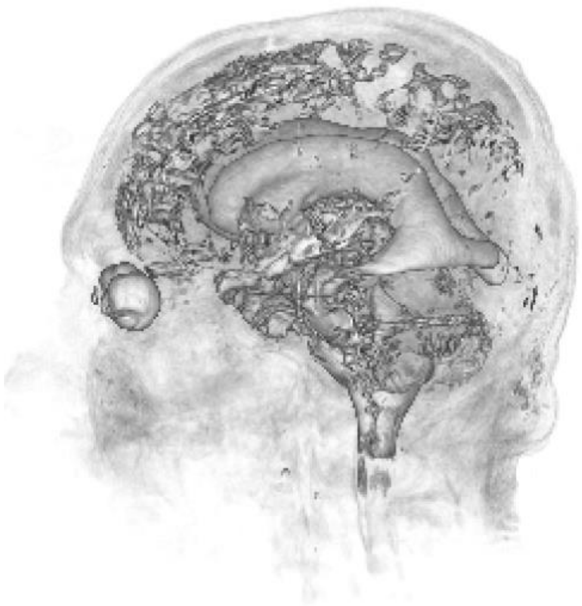
Engine Block
Stanford University



Visible Human
(National Library of Medicine)

Voxel Processing

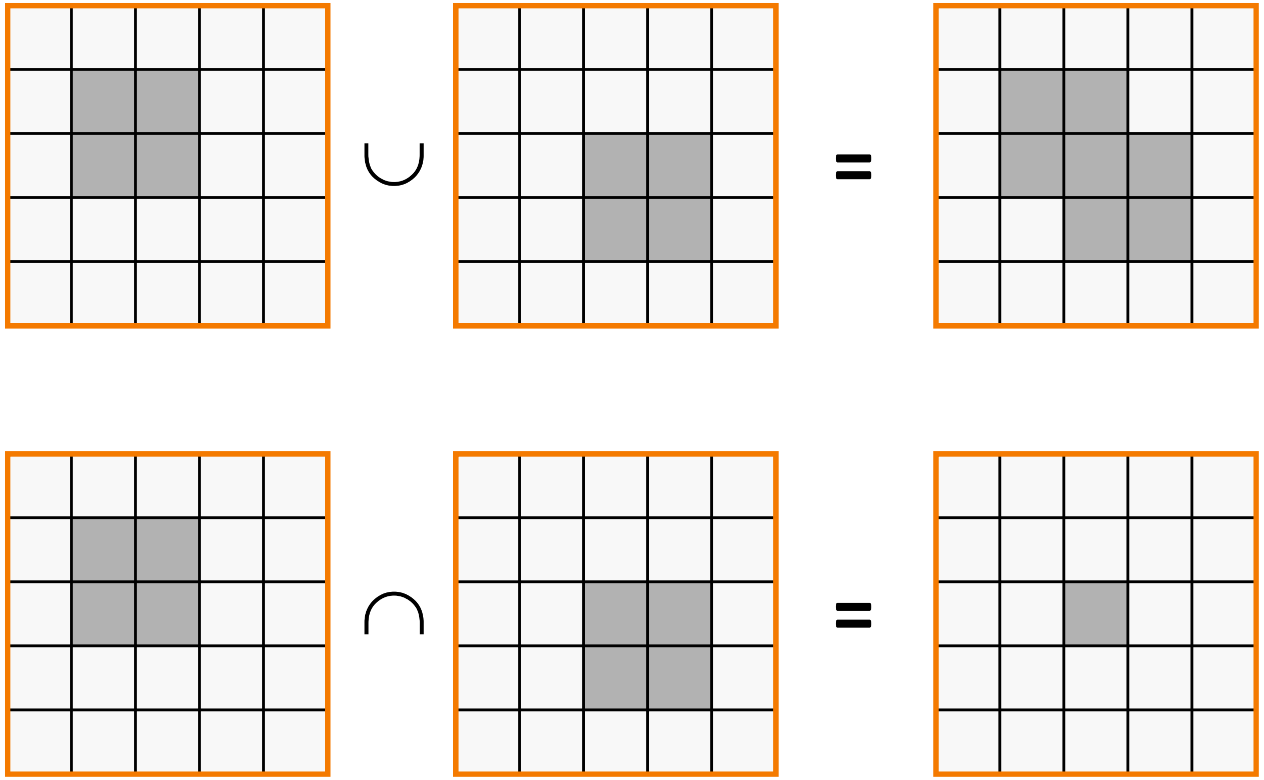
- Signal processing (just like images)
  - Reconstruction
  - Resampling
- Typical operations
  - Blur
  - Edge detect
  - Warp
  - etc.
- Often fully analogous to image processing



www.volumegraphics.com

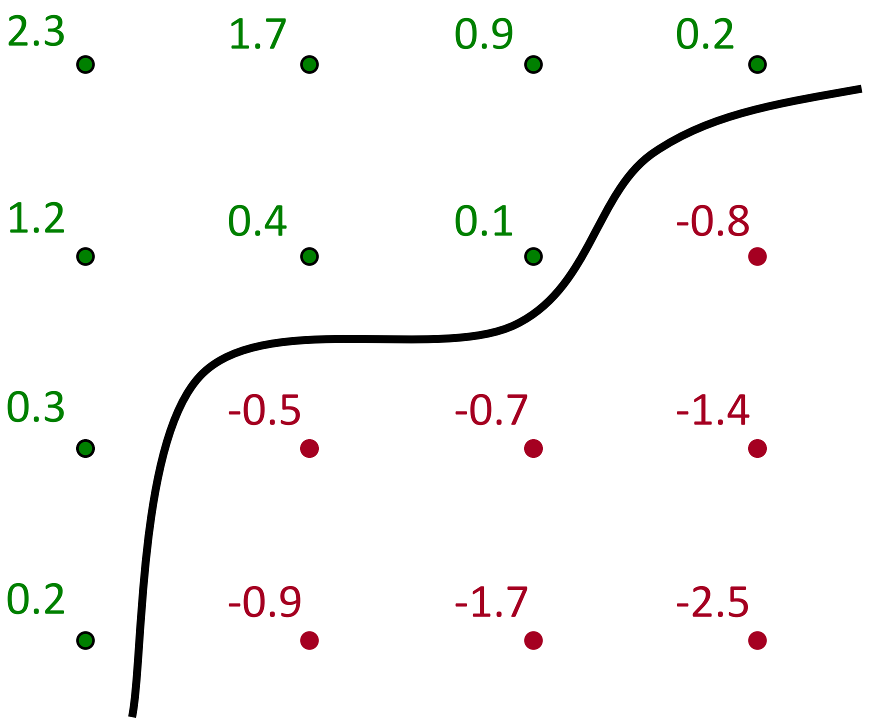
Voxel Boolean Operations

- Compare objects voxel by voxel
  - Trivial



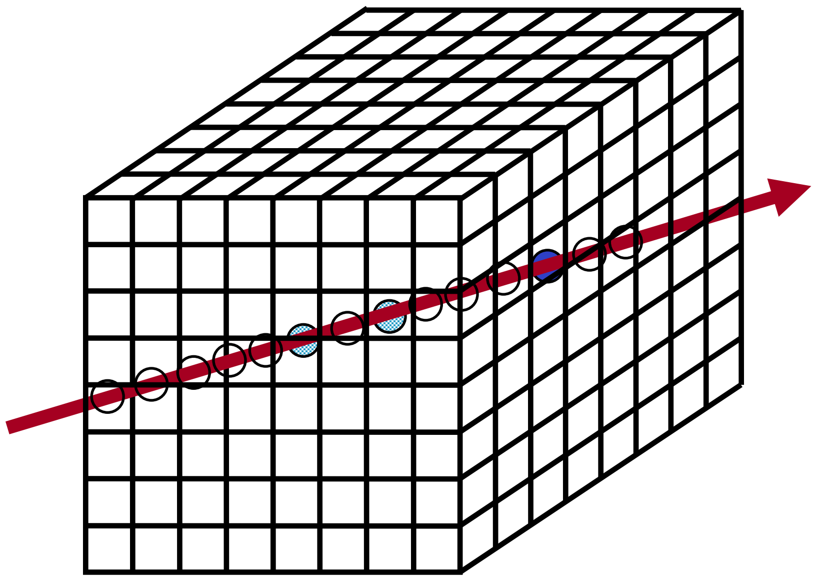
Voxel Display

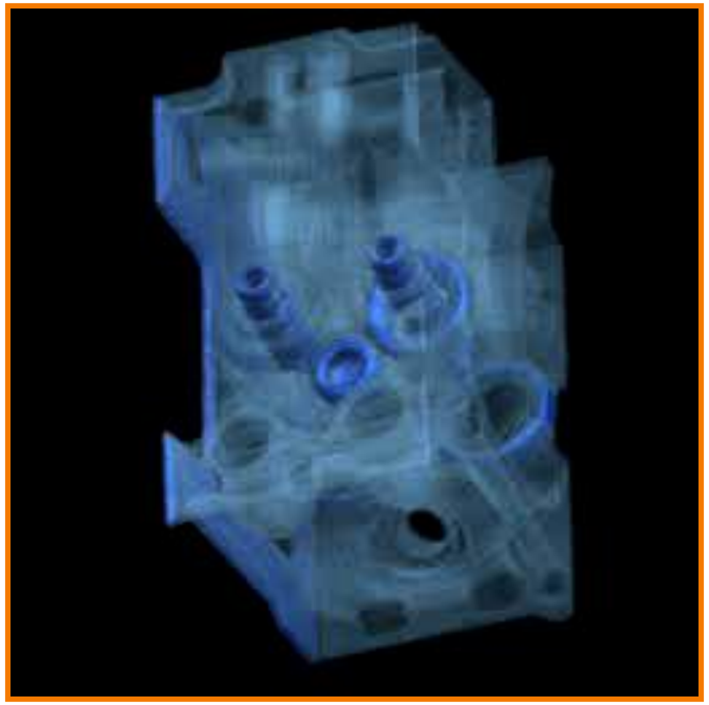
- Isosurface rendering
  - Interpolate samples stored on regular grid
  - Isosurface at f(x,y,z) = 0 defines surface



Voxel Display

- Ray casting
  - Integrate density along rays: compositing!

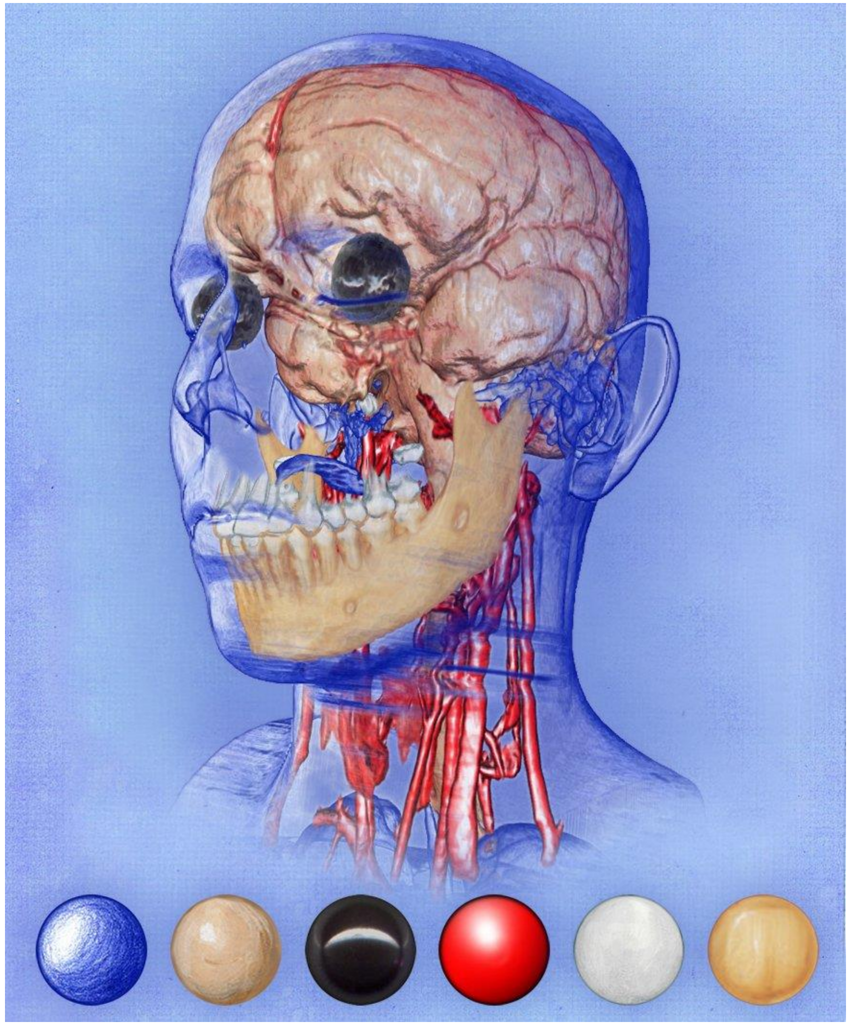




Engine Block
Stanford University

Voxel Display

- Extended ray-casting
  - Transfer functions: Map voxel values to opacity and material
  - Normals (for lighting) from density gradient



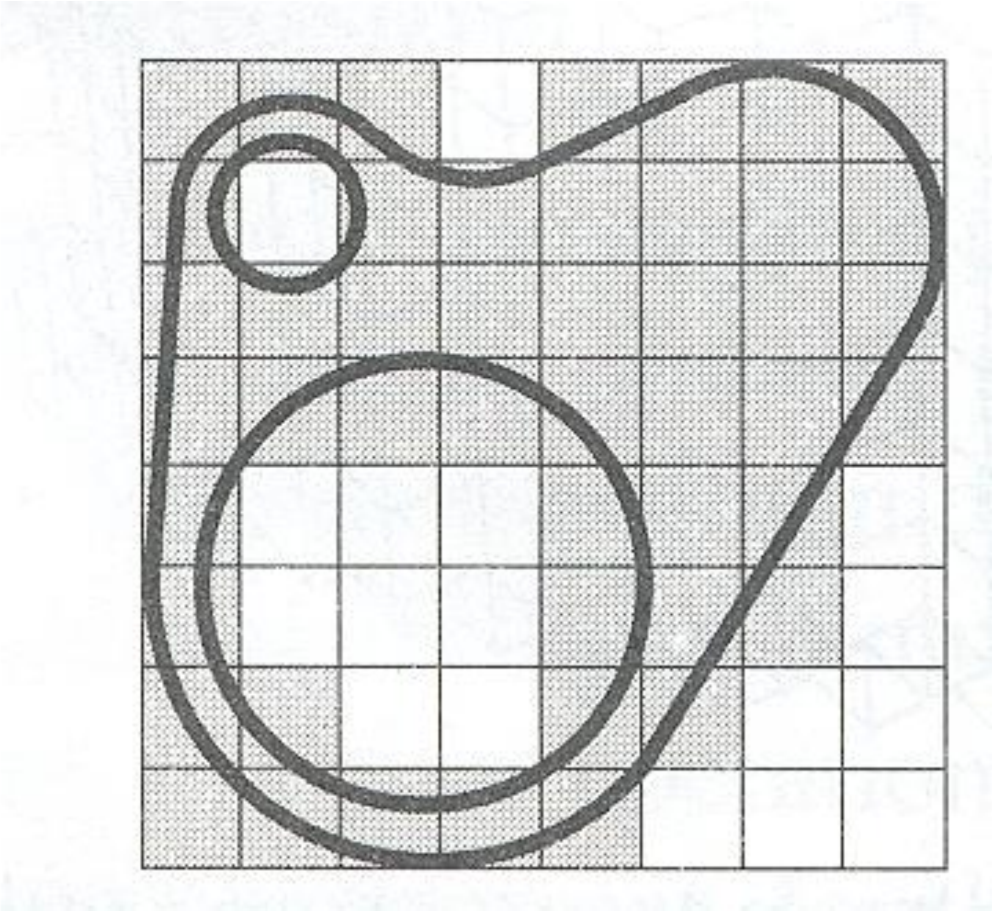
Bruckner et al. 2007

Voxels

- Advantages
  - Simple, intuitive, unambiguous
  - Same complexity for all objects
  - Natural acquisition for some applications
  - Trivial boolean operations
- Disadvantages
  - Approximate
  - Expensive display
  - Large storage requirements

Voxels

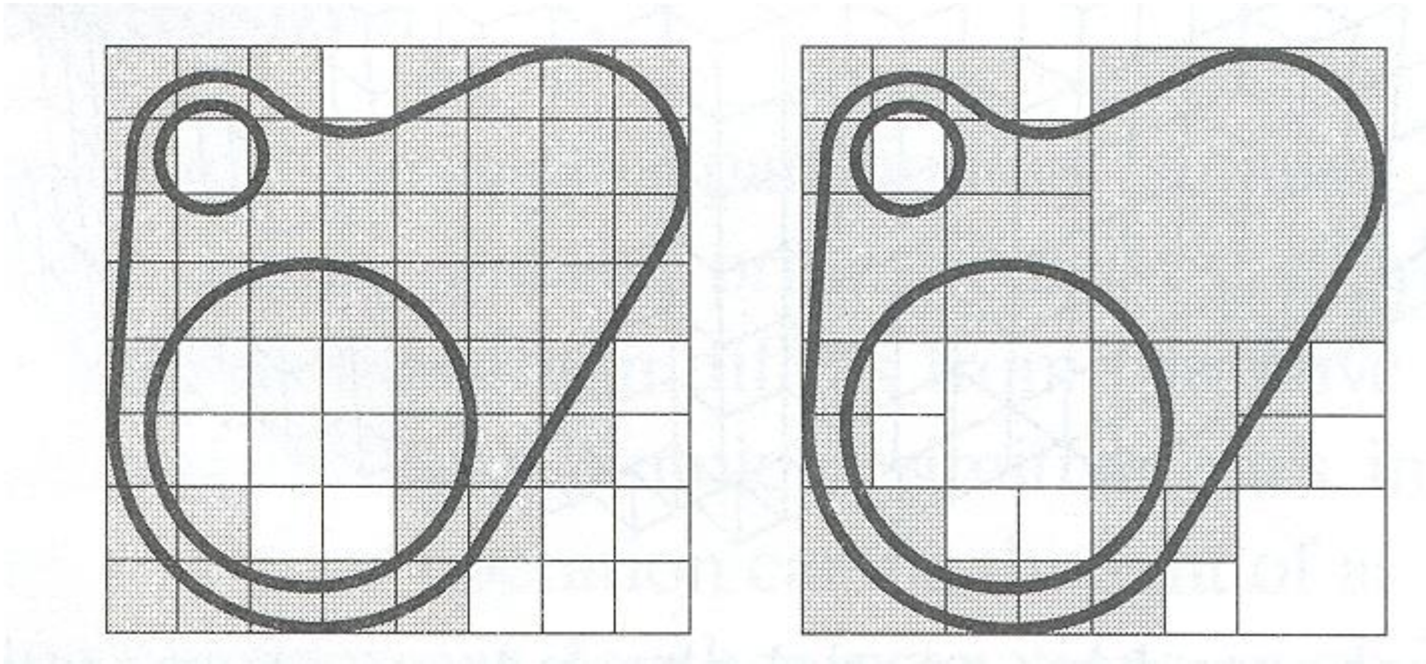
- What resolution should be used?



FvDFH Figure 12.21

Quadtrees & Octrees

- Refine resolution of voxels hierarchically
  - More concise and efficient for non-uniform objects

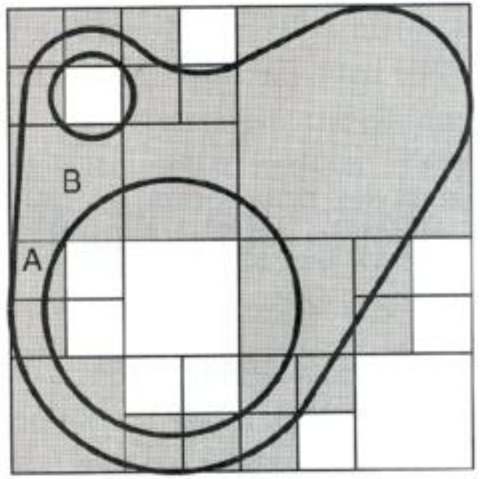


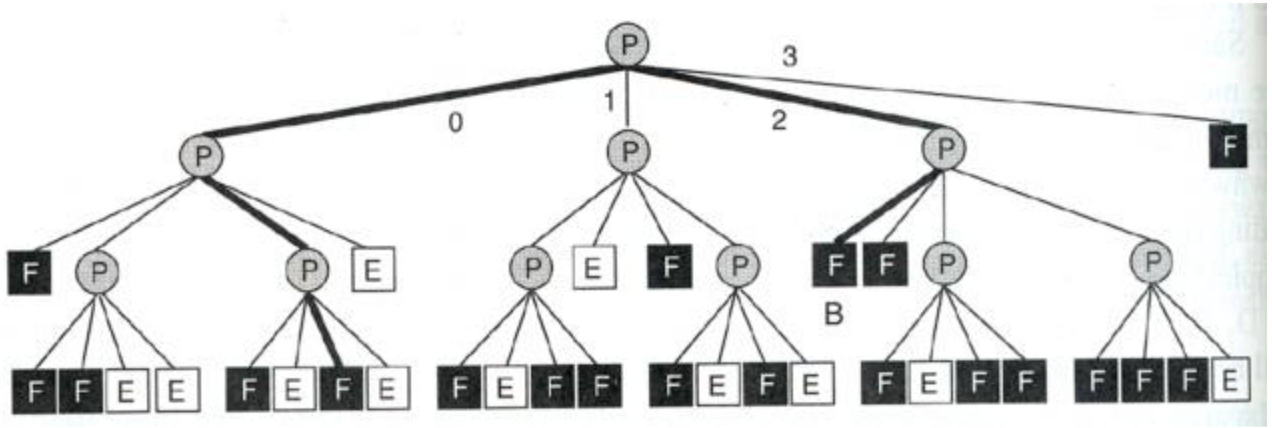
Uniform Voxels   Quadtree (Octree in 3D)

FvDFH Figure 12.21

Quadtree Processing

- Hierarchical versions of voxel methods
  - Finding neighbor cell requires traversal of hierarchy: expected/amortized O(1)



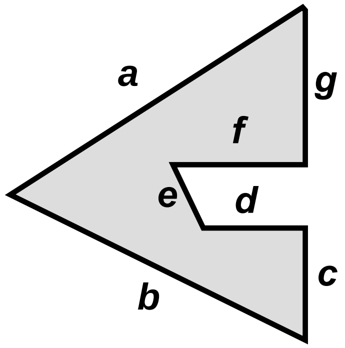


FvDFH Figure 12.25

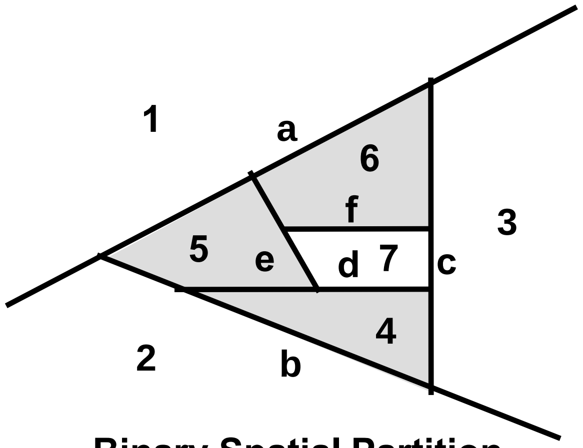
3D Object Representations

- Raw data
  - Range image
  - Point cloud
- Surfaces
  - Polygonal mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - ➢ BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Application specific

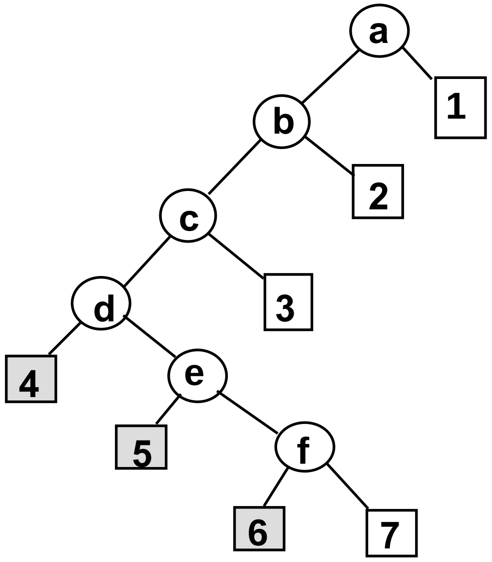
BSP Trees



Object



Binary Spatial Partition



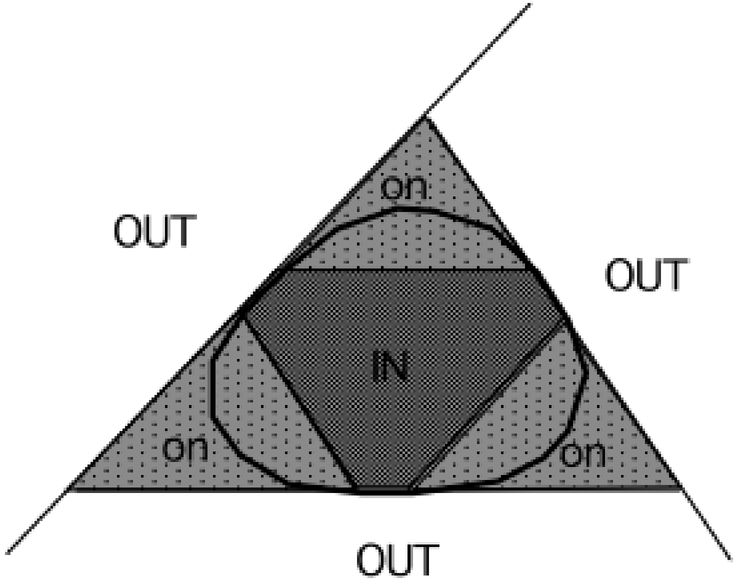
Binary Tree

Naylor

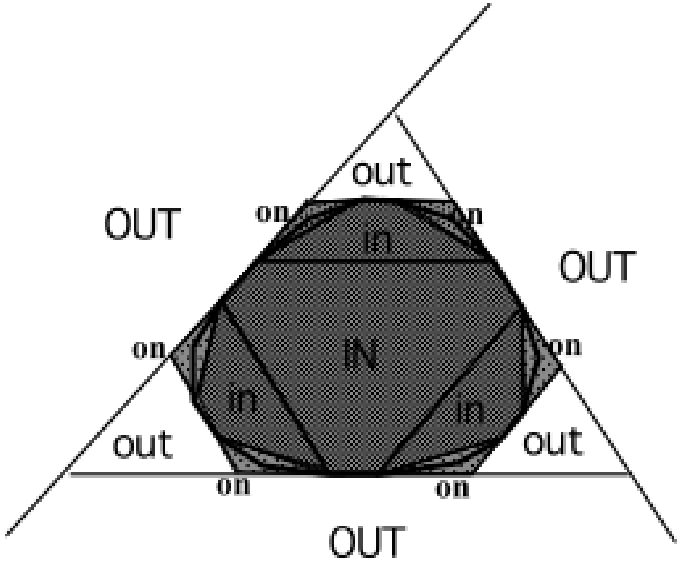
75

BSP Trees

- Key properties
  - visibility ordering (later)
  - hierarchy of convex regions (useful for collision)



1st level Approximation



2nd level Approximation

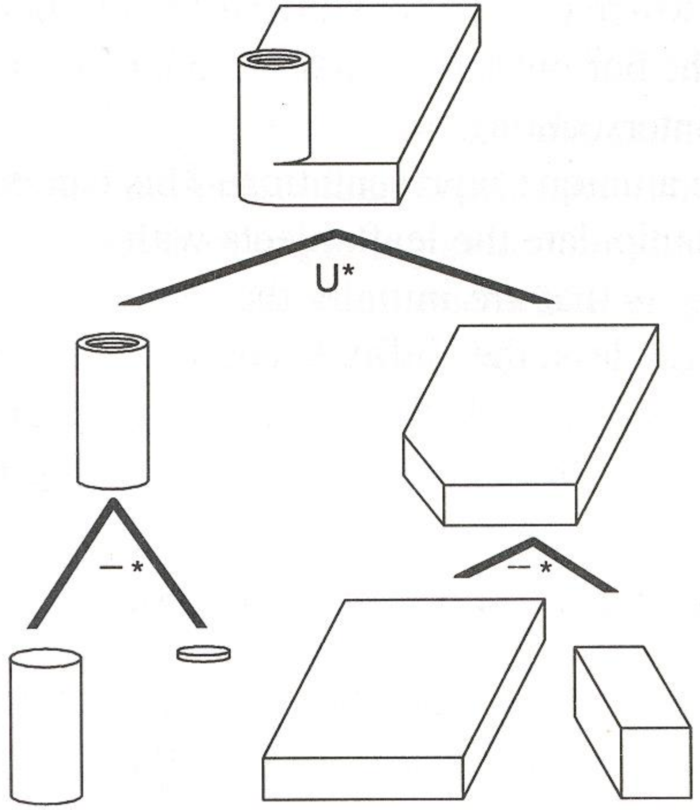
Naylor

3D Object Representations

- Raw data
  - Range image
  - Point cloud
- Surfaces
  - Polygonal mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - ➢ CSG
  - Sweep
- High-level structures
  - Scene graph
  - Application specific

Constructive Solid Geometry (CSG)

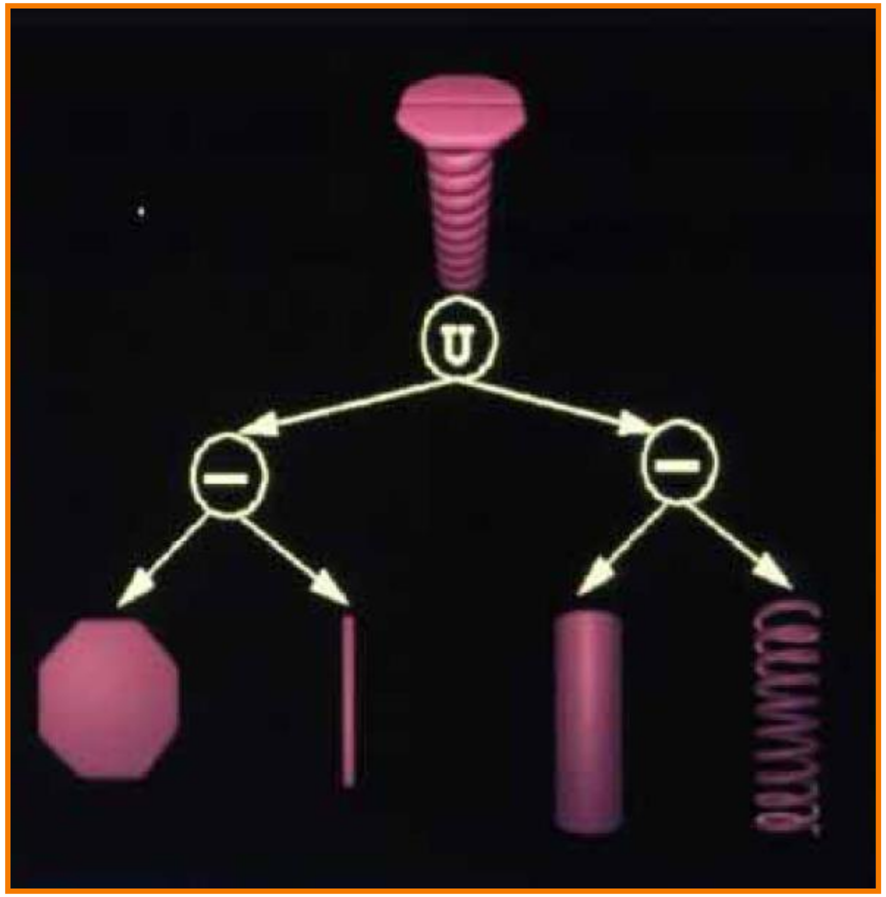
- Represent solid object as hierarchy of boolean operations
  - Union
  - Intersection
  - Difference



FvDFH Figure 12.27

CSG Acquisition

- Interactive modeling programs
  - Intuitive way to design objects

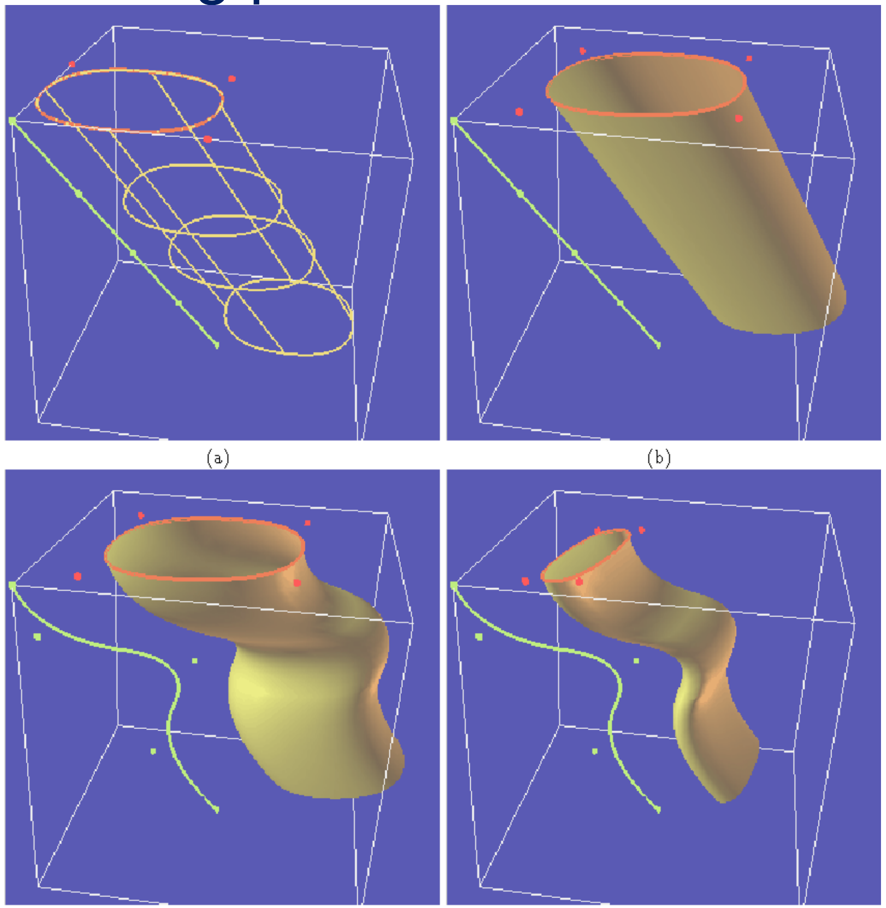


3D Object Representations

- Raw data
  - Range image
  - Point cloud
- Surfaces
  - Polygonal mesh
  - Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - ➢ Sweep
- High-level structures
  - Scene graph
  - Application specific

Sweeps

- Swept volume
  - Sweep one curve along path of another curve

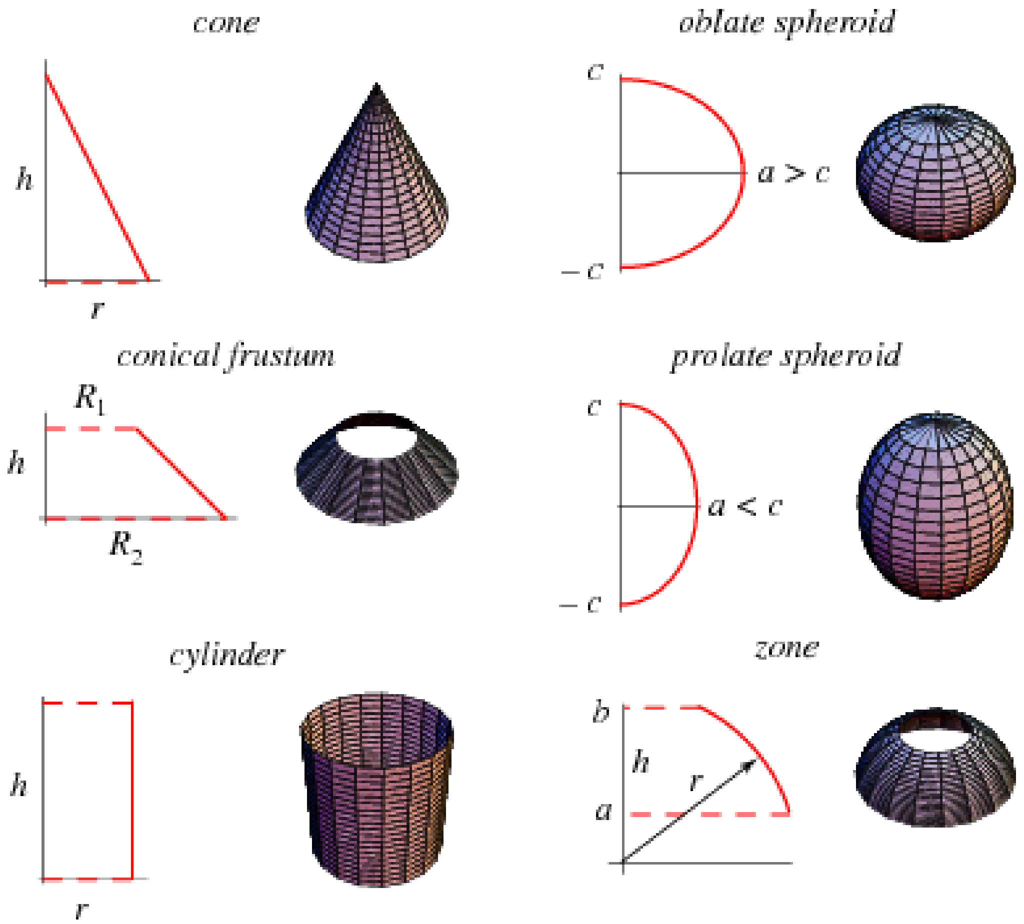


(a) (b) (c) (d)

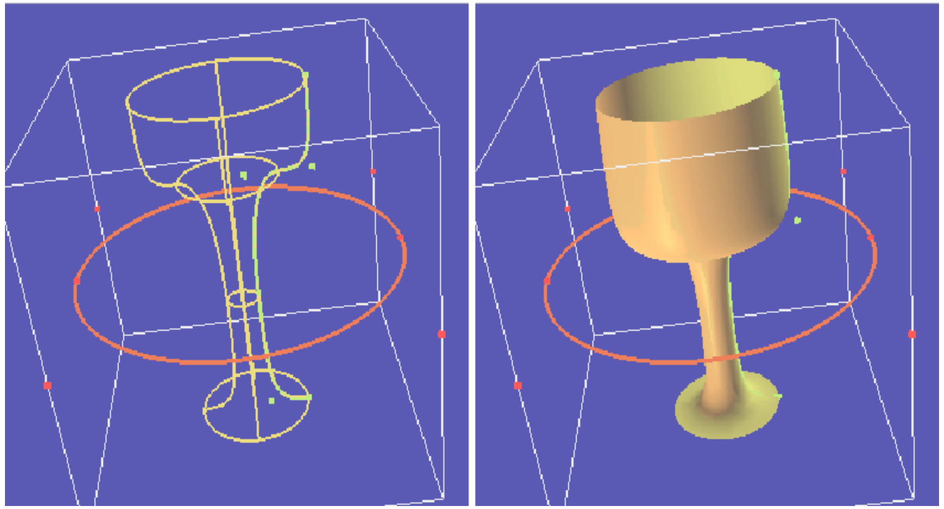
Demetri Terzopoulos

Sweeps

- Surface of revolution
  - Take a curve and rotate it about an axis



Wolfram



Demetri Terzopoulos

Summary

Feature	Voxels	Octree	BSP	CSG
Accurate	No	No	Some	Some
Concise	No	No	No	Yes
Affine invariant	No	No	Yes	Yes
Easy acquisition	Some	Some	No	Some
Guaranteed validity	Yes	Yes	Yes	No
Efficient boolean ops	Yes	Yes	Yes	Yes
Efficient display	No	No	Yes	No