



Solid Modeling

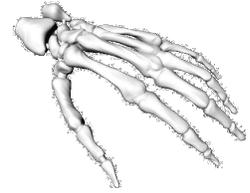
Adam Finkelstein & Tim Weyrich
Princeton University
COS 426, Spring 2008

1



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.



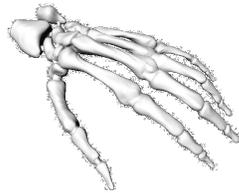
Large Geometric Model Repository
Georgia Tech

2



3D Object Representations

- Properties of surface representations
 - Easy to acquire
 - Accurate
 - Concise
 - Intuitive editing
 - Efficient editing
 - Efficient display
 - ~~Efficient intersections~~
 - ~~Guaranteed validity~~
 - Guaranteed smoothness
 - etc.



Large Geometric Model Repository
Georgia Tech

3



3D Object Representations

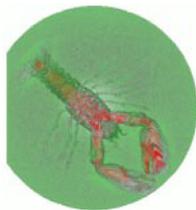
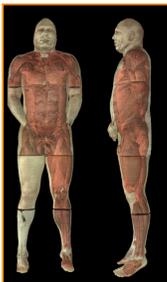
- Points
 - Range image
 - Point cloud
- Surfaces
 - Polygonal mesh
 - Subdivision
 - Parametric
 - Implicit
- Solids
 - Voxels
 - BSP tree
 - CSG
 - Sweep
- High-level structures
 - Scene graph
 - Application specific

4



Solid Modeling

- Represent solid interiors of objects
 - Surface may not be described explicitly



Visible Human
(National Library of Medicine)

5



Motivation 1

- Some acquisition methods generate solids
 - Example: CAT scan



Stanford University

6

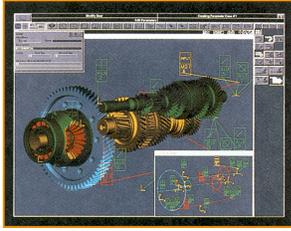
Motivation 2



- Some applications require solids
 - Examples: medicine, CAD/CAM



SUNY Stony Brook



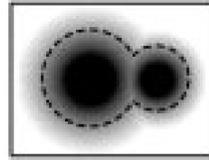
Intergraph Corporation

7

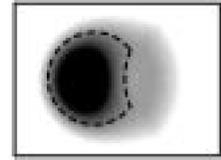
Motivation 3



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



Union



Difference

Bloomenthal

8

Solid Modeling Representations



- Voxels
- Quadrees & Octrees
- KD and BSP Trees
- Constructive solid geometry

9

Solid Modeling Representations



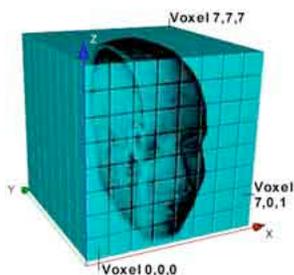
- **Voxels**
- Quadrees & Octrees
- KD and BSP Trees
- Constructive solid geometry

10

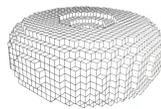
Voxels



- Regular array of 3D samples (like image)
 - Samples are called *voxels* ("volume pixels")



www.volumegraphics.com



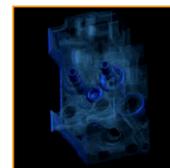
FvDFH Figure 12.20

11

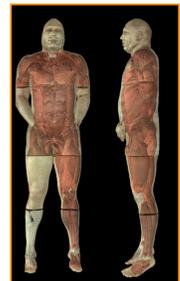
Voxels



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



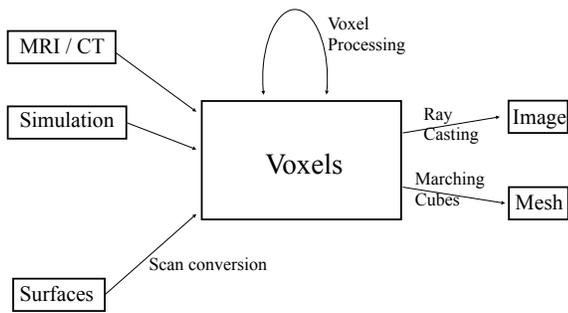
Engine Block
Stanford University



Visible Human
(National Library of Medicine)

12

Voxels

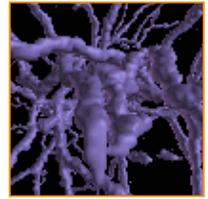


13

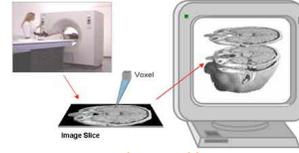
Voxel Acquisition



- Scanning devices
 - MRI
 - fMRI
 - CAT
 - etc.



SUNY Stony Brook



www.volumegraphics.com

14

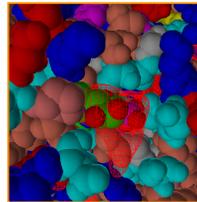
Voxel Acquisition



- Simulation
 - Models
 - FEM
 - etc.



Airflow Inside a Thunderstorm
(Bob Wilhelmson, University of Illinois at Urbana-Champaign)

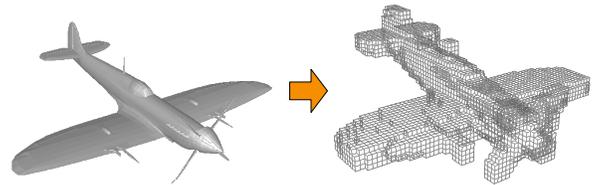


15

Voxel Acquisition



- Voxelization of surfaces

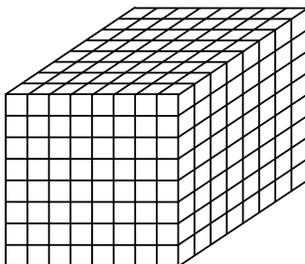


16

Voxel Storage



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

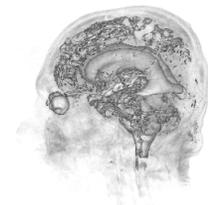


17

Voxel Processing



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



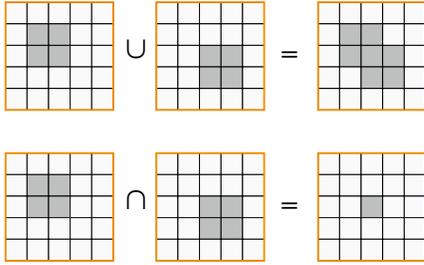
www.volumegraphics.com

18

Voxel Boolean Operations



- Compare objects voxel by voxel
 - Trivial

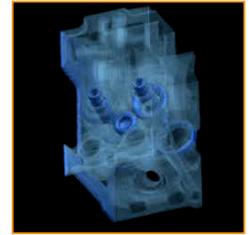
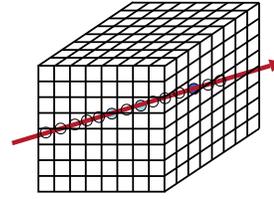


19

Voxel Display



- Ray casting
 - Integrate density along rays through pixels



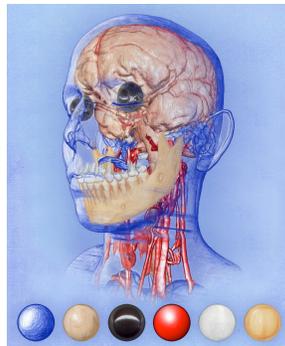
Engine Block
Stanford University

20

Voxel Display



- Extended ray-casting
 - Complex transfer functions
 - Map voxel densities to materials
 - Evaluating of "normals" at material transitions



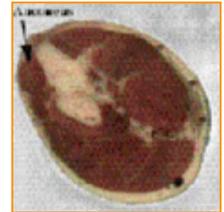
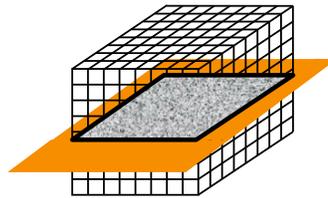
Bruckner et al. 2007

21

Voxel Display



- Slicing
 - Draw 2D image resulting from intersecting voxels with a plane



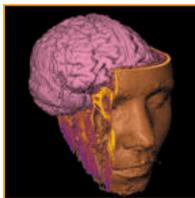
Visible Human
(National Library of Medicine)

22

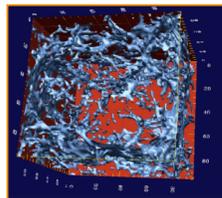
Voxel Display



- Isosurface rendering
 - Render surfaces bounding volumetric regions of constant value (e.g., density)



SUNY Stony Brook



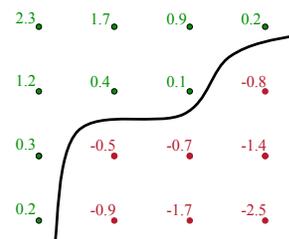
Princeton University

23

Voxel Display

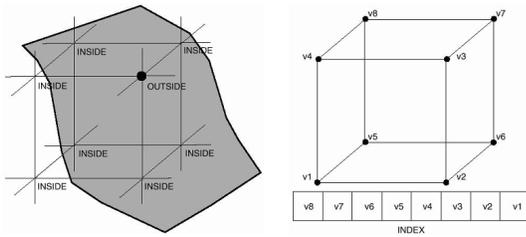


- Isosurface rendering
 - Interpolate samples stored on regular grid
 - Isosurface at $f(x,y,z) = 0$ defines surface



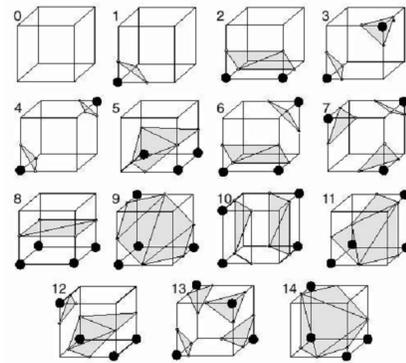
24

Marching Cubes



25

Marching Cubes (15 Cases)

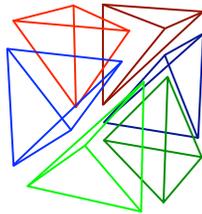


26

Marching Tetrahedra



- Decompose each voxel into 6 tetrahedra



- Mirror decomposition between adjacent voxels for consistent tetrahedrization

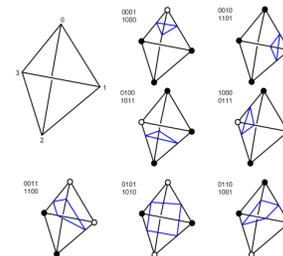
27

27

Marching Tetrahedra



- Requires less cases than marching cubes:



- Easier implementation, but extracted surface less regular

28

28

Voxels



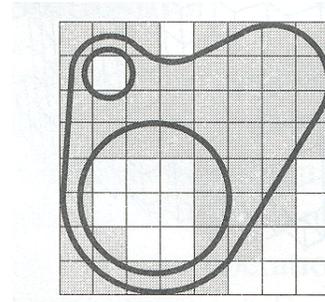
- Advantages
 - Simple, intuitive, unambiguous
 - Same complexity for all objects
 - Natural acquisition for some applications
 - Trivial boolean operations
- Disadvantages
 - Approximate
 - Not affine invariant
 - Expensive display
 - Large storage requirements

29

Voxels



- What resolution should be used?



FvDFH Figure 12.21

30

Solid Modeling Representations



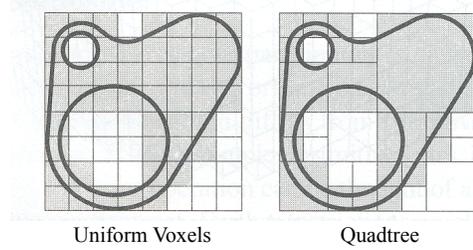
- Voxels
- **Quadtrees & Octrees**
- KD and BSP Trees
- Constructive solid geometry

31

Quadtrees & Octrees



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



Uniform Voxels

Quadtree

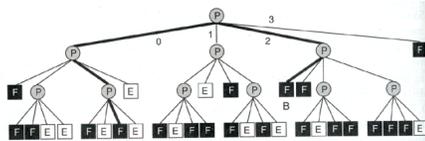
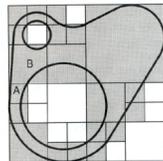
FvDFH Figure 12.21

32

Quadtree Processing



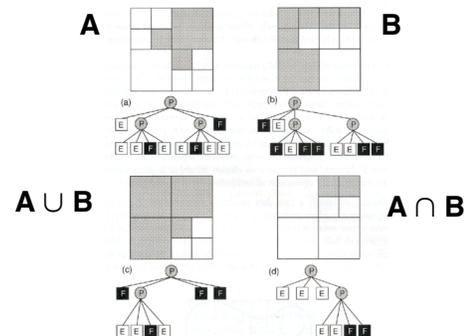
- Hierarchical versions of voxel methods
 - Finding neighbor cell requires traversal of hierarchy ($O(1)$)



FvDFH Figure 12.25

33

Quadtree Boolean Operations



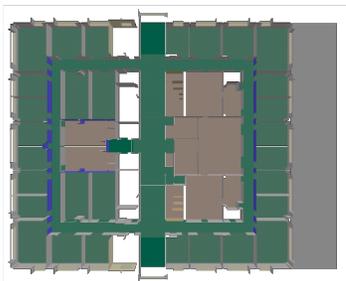
FvDFH Figure 12.24

34

Quadtrees



- How concise/accurate are they?



35

Solid Modeling Representations



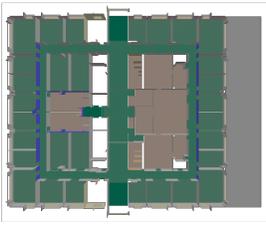
- Voxels
- Quadtrees & Octrees
- **KD and BSP Trees**
- Constructive solid geometry

36

KD Trees



- Recursive binary partition of space by axis aligned planes

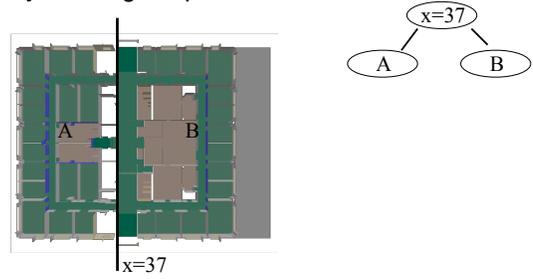


37

KD Trees



- Recursive binary partition of space by axis aligned planes

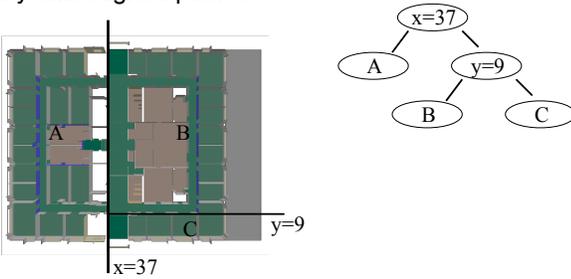


38

KD Trees



- Recursive binary partition of space by axis aligned planes

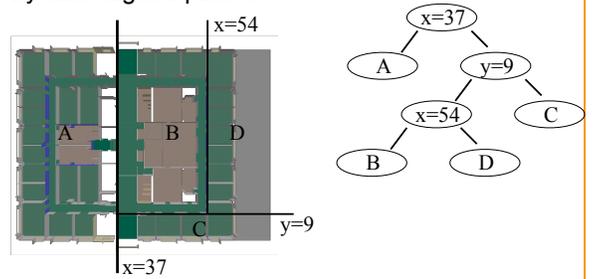


39

KD Trees



- Recursive binary partition of space by axis aligned planes

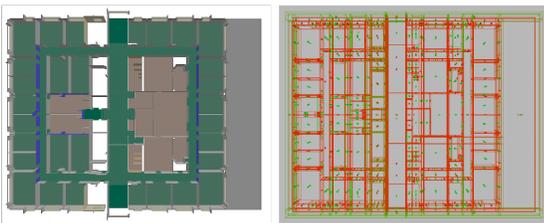


40

KD Trees



- Recursive binary partition of space by axis aligned planes

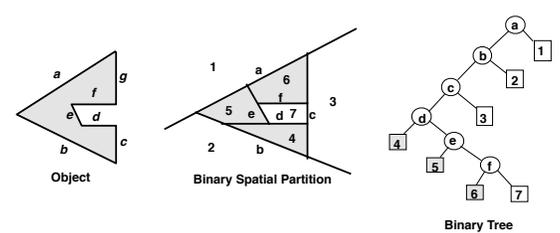


41

BSP Trees



- Recursive binary partition of space by arbitrary planes



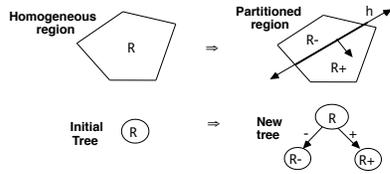
Naylor

42

BSP Trees



- Single geometric operation
 - Partition a convex region by a hyperplane
- Single combinatorial operation
 - Two child nodes added as leaf nodes



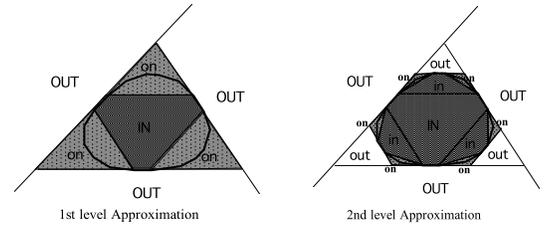
Naylor

43

BSP Trees



- Fast algorithms exploit
 - visibility ordering (later)
 - hierarchy of convex regions



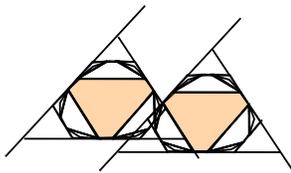
Naylor

44

BSP Trees



- Fast algorithms exploit
 - visibility ordering (later)
 - hierarchy of convex regions



Naylor

45

BSP Trees



- Heavily used in 3D computer games
- Static "level" geometry represented as BSP tree
- Used for
 - Collision detection
 - Visibility ordering
 - Point-to-point visibility
 - ...



Doom, 1993 by ID software

46

46

Solid Modeling Representations



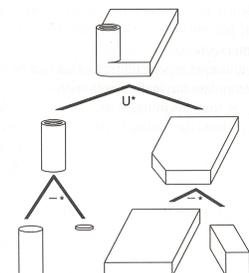
- Voxels
- Quadrees & Octrees
- KD and BSP Trees
- **Constructive solid geometry**

47

Constructive Solid Geometry (CSG)



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



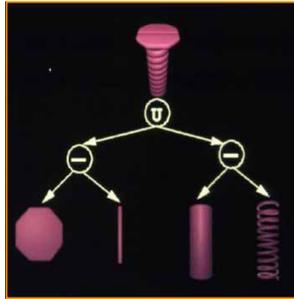
FvDFH Figure 12.27

48

CSG Acquisition



- Interactive modeling programs
 - Intuitive way to design objects



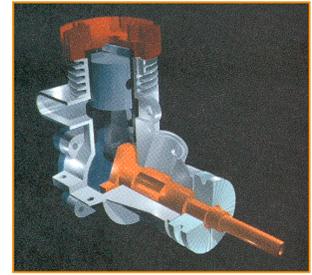
SUNY Stoney Brook

49

CSG Acquisition



- Interactive modeling programs
 - Intuitive way to design objects



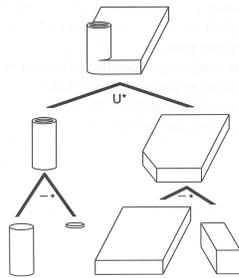
H&B Figure 9.9

50

CSG Boolean Operations



- Create a new CSG node joining subtrees
 - Union
 - Intersection
 - Difference



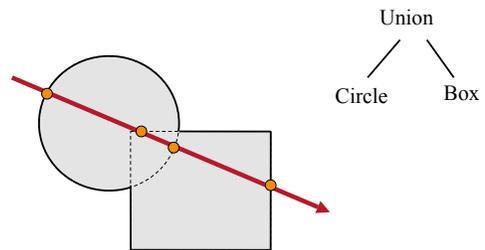
FvDFH Figure 12.27

51

CSG Display & Analysis



- Ray casting



52

Summary



	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 operations	Yes	Yes	Yes	Yes
Efficient display	No	No	Yes	No

53