## Lecture 11

### Introduction to Solid Modeling

Wednesday, February 23, 2000

William H. Hsu
Department of Computing and Information Sciences, KSU
http://www.cis.ksu.edu/~bhsu

Readings:
Sections 12.6-12.10, Foley *et al*
(Reference) 10.15-10.17 Hearn and Baker 2e
Slide Set 5, VanDam (8b, 11/09/1999)



## Lecture Outline

- **Readings**
  - Sections 12.6-12.10, Foley *et al*
  - Outside reading (optional): 10.15-10.17 Hearn and Baker 2e
  - Outside reading (required): Slide Set 8b, VanDam (11/09/1999)
- **Last Time**
  - Overview: data structures
  - Boolean set operations (12.2 FVD), primitive instancing (12.3 FVD), sweeps (12.4 FVD), boundary representations (B-reps, 12.5 FVD)
- **Today**
  - Spatial partitioning representations
    - Cell decomposition
    - (Planar and) Spatial occupancy: pixel, voxel
    - Hierarchical spatial occupancy: quadtrees, octrees; algorithms
  - Binary Space Partitioning (BSP) trees
  - Constructive Solid Geometry (CSG)
- **Next Class: Color Models; Visible Surface Determination (Intro)**



## Spatial Partitioning [1]: Cell Decomposition

- **Intuitive Idea**
  - Define set of primitive cells (typically parametric, often curved)
  - Difference from primitive instancing: "glue" primitive objects together
    - Glue operation (part of specification): non-intersecting "union"
    - Example: join two objects at specified faces
- **Tradeoffs**
  - Advantages
    - Results in *unambiguous* descriptions of complex objects
    - Admits additional specification (e.g., *how* object faces joined)
  - Disadvantages
    - *Descriptions not necessarily unique* (see Figure 12.19, FVD)
    - *May be difficult to validate* (model checking: many intersection tests needed)
- **When to Use**
  - Restrictive *constraint language* available (cuts down number of validation cases)
  - Example: *finite element analysis* ("glue spec" determines physical model)



## Spatial Partitioning [2]: Uniform and Hierarchical (Quadtree/Octree)

- **Intuitive Idea**
  - Special case of cell decomposition: *identical cells* arranged in *fixed, regular grid*
  - Cells: pixels (picture elements) for planar decomposition, voxels (volumetric elements) for spatial decomposition
  - Most common type: cubic voxel (decomposed object: cuberille)
- **Tradeoffs**
  - Advantages
    - Easy to perform cell classification (i.e., test whether inside or outside solid)
    - Easy to test adjacency of two objects
  - Disadvantages
    - No "partial" occupancy; many solids can only be approximated (*when*?)
    - Expensive to store; basic data structure admits high redundancy (*why*?)
- **When to Use**
  - Applications where volumetric data representation is needed
  - Examples: biomedicine (e.g., computerized axial tomography *aka* CAT); other nondestructive evaluation (NDE)



## Terminology

- **Modeling Solid Objects**
  - Data structures
    - Boundary representations (*aka* B-reps): describe solid in terms of *surfaces*
    - Spatial partitioning representations: describe solid in terms of *subparts*
  - Basic algorithms
    - Construction (aka composition): form new structure by composing primitives
    - Intersection: compute intersection point (if any) with ray, line, *other structure*
    - Point classification: tell whether query point lies inside or outside
- **Spatial Partitioning**
  - Cell decomposition: breaking complex object up into primitive *cells*
  - Planar and spatial occupancy
    - Voxel: volumetric unit (typically cubic, resulting in cuberille)
    - Hierarchical: variable-granularity decomposition, e.g., quadtrees and octrees
  - Binary Space Partitioning (BSP) tree: break space up into half-spaces
  - Constructive Solid Geometry (CSG): *combine* primitives using Boolean set operators and *modify* them using (unary) transformation operations



## Summary Points

- **Solid Modeling: Overview**
  - Data structures
    - Boundary representations (B-reps): last time
    - Spatial partitioning representations: today
  - Algorithms
    - Construction (composition)
    - Intersection, point classification
  - Know: difference between B-reps and spatial partitioning; pros and cons
- **Spatial Partitioning (Review Guide)**
  - Cell decomposition – know how to *obtain* for composite object (simple primitives)
  - Planar and spatial occupancy
    - Simple: uniform subdivision (grid / pixel, volumetric / voxel)
    - Hierarchical: quadtrees and octrees – know how to *obtain* for 2D, 3D scenes
  - Binary Space Partitioning (BSP) trees – know how to *obtain* for simple 2D object
  - Constructive Solid Geometry (CSG) – know *typical* primitives, how to combine
- **Next Class: Color Models; Visible Surface Data Structures**