Hidden Surface Removal (visibility)
Adam Finkelstein
Princeton University
COS 426, Fall 2001

Overview

• Motivation
• Algorithms for HSR
  o Back-face detection
  o Depth sort
  o Ray casting
  o Scan-line
  o Z-buffer
  o Area subdivision
• Tradeoffs

Motivation

• Surfaces may be back-facing.
• Surfaces may be occluded.
• Surfaces may overlap in the image plane.
• Surfaces may intersect.

3D Rendering Pipeline

Somewhere in here we have to decide which objects are visible, and which objects are hidden.

Overview

• Motivation
• Algorithms for HSR
  o Back-face detection
  o Depth sort
  o Ray casting
  o Scan-line
  o Z-buffer
  o Area subdivision
• Tradeoffs

Back-face detection

Q: When does this method break down?
A: More than one object. Object not closed. Interreflect?

Q: How do we test for back-facing polygons?
A: Dot product of the normal and view directions.
A polygon is back-facing if $V \cdot N > 0$.

Depth sort

“Painter’s algorithm”
- Sort surfaces in order of decreasing maximum depth
- Scan convert surfaces in back-to-front order

BSP Tree

- Binary space partition with solid cells labeled
  - Constructed from polygonal representations
  - Provides linear-time depth sort for arbitrary view

Ray Casting

- Fire a ray for every pixel
  - If ray intersects multiple objects, take the closest

Ray Casting Pipeline

- $O(p \log n)$ for $p$ pixels
- May (or not) use pixel coherence
- Simple, but generally not used
Z-Buffer

- Store color & depth of closest object for every pixel
  - Update only pixels whose depth is closer than in buffer
  - Depths are interpolated from vertices, just like colors

Scan-Line Algorithm

- For each scan line, construct spans
  - Sort by depth

Area Subdivision

Warnock's algorithm
  - Fill area if:
    - All surfaces are outside area, or
    - Only one surface intersects area, or
    - One surface occludes other surfaces in area
  - Otherwise, subdivide

3D Rendering Pipeline

Z-buffer comments
  - Polygons rasterized in any order
  - Requires lots of memory
    - 1K x 1K x 24 bits
    - Was expensive, cheap now
  - Subject to aliasing (A-buffer)
  - Commonly in hardware

Scan-line comments
  - Fully compute only visible pixels
  - Coherence among along scans
  - Commonly in software

Area subdivision comments
  - Augments scan conversion
  - Polygon coherence
  - Commonly in software
Algorithms for HSR

- Back-face detection
- Depth sort
- Ray casting
- Scan-line
- Z-buffer
- Area subdivision

• Where in pipeline?
• Hardware / Software?
• Trends in hardware.

Conclusions