Handling complexity

- Level of detail variation
- Progressive refinement
- Handling very complex scenes
- Frameless rendering

Toward Uniform Frame Times

- Interactive/Real-time limits frame time
- Smooth motion etc. requires uniform frame time
- Polygon-bound systems can use multiple LODs
  - LOD = Level of Detail
- Flight simulator systems use LODs heavily
  - Mostly hand-modeled
  - Predictable scene complexity
- Cell-based visibility for indoor scenes (from ‘60s)
- Additional methods needed where complexity varies from frame to frame

Funkhouser and Sequin ‘93

- Visibility culling (view frustum and cells)
- Static level-of-detail thresholds (distance, size)
  - Blend between LODs to avoid pops (chair in paper)
- Dynamic LOD using previous frame time - reactive
- Predictive frame time cost heuristic
  - Based on primitive count and pixel estimate
  - Measurements give rough values (chair example)
  - rendering style also variable
- Benefit heuristic
  - Object importance, position in frame
  - Motion, and LOD hysteresis

More Funkhouser and Sequin ‘93

- Path through 87k polygon database (in paper)
- Sudden changes in scene complexity at turns
- Static LOD - high frame time variance
- Feedback - muffs sudden complexity changes
- Optimization - does best (0.13 - 0.075s)
- Video from Siggraph ‘93

Automatic Object Simplification

- Merge polygons with small dihedral angles
- Join vertices when edge gets short
- Drop polygons when degenerate
- Drop vertices in middle of small shallow clusters
- Error criteria for surfaces
- Issues for spindly objects screen doors, etc.
- Combining parts, filling holes, etc.

Progressive Meshes

- Hoppe ‘96, ‘97
- Retain simplification schema
- Replay to build out from simplified object
- Smooth morph between levels (geomorph)
- Retain color variation using texture (Certain et al ‘96)

Adding view dependence

- Hoppe ‘97
  - View frustum determines refinement
  - Angle to viewer varies refinement
- Luebke and Erikson ‘97
  - Progressive refinement in large database
  - Space division polygon collapse scheme

Appearance-Preserving Simplification

- Cohen, Olano, Manocha ‘98
- Use error-bounded geometry simplification
- Add error-bounded texture distortion
- Use with bump-mapped texture (Pixelflow)
- What do you do when bumps on bump maps get too small?

For Very High Depth Complexity

- Hierarchical depth buffer (Greene ‘93)
  - Octree spatial subdivision
  - Pyramid multiresolution depth buffer
  - Temporal coherence by saving visible octree nodes
- Antialiasing, etc. with Coverage masks
  - Greene ‘96 (-7:30 on ‘96 tape)
  - Adds “triage” coverage masks, to subpixels
  - 8x8 hierarchy, 512 to 64 to 8 to 1/8 (4096 grid)

Frameless rendering

- Bishop ‘94
  - Update pixels at random with latest info
- Shirley et al ‘99
  - Interactive ray tracing on 64-processor SGI
  - Frameless technique improves interaction

References

- [Funkhouser93] Thomas A Funkhouser and Carlo H. Sequin, "Adaptive Display Algorithm for Interactive Frame Rates During Visualization of Complex Virtual Environments", Proc. Siggraph '93, August 1993, pp. 247-254.
- [Hoppe96] Hughes Hoppe, "Progressive Meshes", Proc. Siggraph '96, August 1996, pp. 99-108.
- [Hoppe97] Hughes Hoppe, "View-Deoendent Refinement of Progressive Meshes", Proc. Siggraph '97, August 1997, pp. 189-198.
- [Certain96] Andrew Certain, Jovan Popovich, Tony DeRose, Tom Duchamp, David Salesin, Werner Stuetzle, "Interactive Multiresolution Surface Viewing", Proc. Siggraph '96, August 1996, pp. 91-98
- [Luebke97] David Luebke,and Carl Erikson, "View-Dependent Simplification of Arbitrary Polygonal Environments", Proc. Siggraph '97, August 1997, pp. 199-208.
- [Cohen98] Jonathon Cohen, Marc Olano, and Dinesh Manocha, "Appearance-Preserving Simplification", Proc. Siggraph '98, July 1998, pp. 115-114.

More References

- [Greene93] Ned Greene, Michael Kass, Gavin Miller, "Hierarchical Z-Buffer Visibility", Proc. Siggraph '93, August 1993, pp. 231-238.
- [Greene96] Ned Greene, "Hierarchical Tiling with Coverage Masks", Proc. Siggraph '96, August 1996, pp. 65-74
- [BIshop94] Gary Bishop, Henry Fuchs, Leonard MacMillan and Ellen Zagier, "Frameless Rendering: Double Buffering Considered Harmful", Proc. Siggraph '94, Computer Graphics, 28(2), July 1994, pp. 175-176.
- [Parker99] Steven Parker, Wiliam Martin, P-P. J. Sloan, Peter Shirley, Brian Smits and Charles Hansen, "Interactive Ray Tracing", Proc. 1999 Symposium on Interactive 3D Graphics, April 1999, pp. 119-126.