Digital Geometry Processing

Many possible surface representations
- Polygonal meshes
- Parametric surfaces
- Subdivision surfaces
- Implicit surfaces
- etc.

Let's focus on 3D polygonal meshes
- Simple, common representation
- Rendering with hardware support
- Output of many acquisition tools
- Input to many simulation/analysis tools

3D Polygonal Meshes
Set of polygonal faces representing a 2D surface embedded in 3D
Polygonal Mesh Acquisition

Interactive modeling
- Polygon editors
- Interchange formats

Scanners
- Laser range scanners
- CAT, MRI, etc. (isosurfaces)

Simulations
- Physical processes

Digital Michelangelo Project

Princeton Shape Benchmark

Stanford
Polygonal Mesh Acquisition

Interactive modeling
- Polygon editors
- Interchange formats

Scanners
- Laser range scanners
- CAT, MRI, etc. (isosurfaces)

Simulations
- Physical processes

Outline

Acquisition
Processing
Representation

Polygonal Mesh Processing

Storage
- Compression
- Transmission

Analysis
- Parameterization
- Differential geometry
- Feature detection
- Segmentation

Editing
- Smoothing, sharpening, etc.
- Deformation
- Completion

Lossy Compression (Simplification)
Polygonal Mesh Processing

Storage
- Compression
- Transmission

Analysis
- Parameterization
- Differential geometry
- Feature detection
- Segmentation

Editing
- Smoothing, sharpening, etc.
- Deformation
- Completion

Sheffer
Polygonal Mesh Processing

Storage
- Compression
- Transmission

Analysis
- Parameterization
- Differential geometry
- Feature detection
- Segmentation

Editing
- Smoothing, sharpening, etc.
- Deformation
  - Completion

Outline

Acquisition
Processing
Representation

Polygon Mesh Representation

Data structures determine algorithms
- Data structure must support key operations of algorithm efficiently

Examples:
- Drawing a mesh
- Removing a vertex
- Computing per-vertex normals

Important properties of mesh representation?

Different data structures for different algorithms

Polygon Mesh Representation

Important properties of mesh representation?
- Efficient traversal of topology
- Efficient use of memory

Possible data structures
- List of independent faces
- Vertex and face tables
- Adjacency lists
- Winged edge
- Half edge
- etc.
Independent Faces

Each face lists vertex coordinates
- Redundant vertices
- No topology information

Vertex and Face Tables

Each face lists vertex references
- Shared vertices
- Still no topology information

Adjacency Lists

Store all vertex, edge, and face adjacencies
- Efficient topology traversal
- Extra storage

Partial Adjacency Lists

Can we store only some adjacency relationships and derive others?

Winged Edge

Adjacency encoded in edges
- All adjacencies in O(1) time
- Little extra storage (fixed records)
- Arbitrary polygons

Winged Edge

Example:
Half Edge

Adjacency encoded in edges
- All adjacencies in $O(1)$ time
- Little extra storage (fixed records)
- Arbitrary polygons

Similar to winged-edge, except adjacency encoded in half-edges

Summary

Polygonal mesh overview
- Acquisition
- Processing
- Representation

Summary

Do polygonal mesh reps have these properties?
- Easy to acquire
- Accurate
- Concise
- Efficient display
- Efficient intersections
- Efficient deformations
- Efficient topology changes
- Guaranteed validity
- Guaranteed smoothness
- Intuitive editing controls

Summary

Next time: Laplacian Surface Editing