Geometric Modeling

- How to design a graphical model?
- Design

- How to create a digital description of a real-world object?
- Digitize

By other names

- Data Generation
- CAD: Computer-Aided Design
- CAAD: Computer-Aided Architectural Design
- Geometric Modeling
- Computational Geometry

Approaches

- interactive design
- procedural composition & construction
- CSG: constructive solid geometry & boolean operators
- subdivision surfaces
- fractals
- isosurfaces of implicit functions

Geometric Modeling Basics

- types of polygons & polyhedra
- computing normals
- geometric computations
- OpenGL concerns
- procedural approaches
types of polygons & polyhedra
- triangle – fixed length polygons
- convex polygon / polyhedron
- concave polygon / polyhedron
- non-planar polygon
- sliver triangles – error-prone normals

Polygonal processing
- triangulation
- converting to convex polyhedra
- intersection testing
- closure testing
- well-formed polyhedron:
  - closed,
  - non-self intersecting
  - 2D manifold

Data structures
- face-based
- winged edge

computing normals
- cross-product of edges
- ordered edges at convex corner
  \[ N = (v_0 - v_1) \times (v_2 - v_1) \]
- summation method
  \[ N_x = \sum (z_i + z_{i+2}) \times (y_{i+1} - y_i) \]
  \[ N_y = \sum (x_i + x_{i+2}) \times (z_{i+1} - z_i) \]
  \[ N_z = \sum (y_i + y_{i+2}) \times (x_{i+1} - x_i) \]
**computing vertex normals**

- for each vertex, $i$, $n[i] = (0,0,0)$
- for each face, $j$
  - Compute the normal, $\text{nrml}$, for the face
  - For each vertex, $i$, of the face $n[i] += \text{nrml}$
- for each vertex, $i$, normalize $n[i]$

**OpenGL concerns**

- vertex normals
- convex polygons
  - triangle strip
  - quad strips
  - triangle fans

**geometric computations**

- topology: genus, holes, & handles
- Euler's formula: $E+2 = F+V+2G$
- Convex hull
- distance to, contained in, intersected by
- meshing & simplification
- LoD: level of detail representations

**Procedural Approaches**

- composition from primitive shapes
- extrusion
- solid of revolution
- lofting
- sweep operator
Scene Graph

Hierarchical representation
File format to record scene description

- Describe a scene
  - transformations
  - primitive objects
  - attributes
  - camera
  - lights

Scene Graph

```
Scene {
  transform {
    attribute
    primitive
  }
  transform {
    attribute
    primitive
  }
}
```