3D Object Representations

- Points
  - Range image
  - Point cloud

- Surfaces
  - Polygonal mesh
  - Subdivision
  - Parametric
  - Implicit

- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep

- High-level structures
  - Scene graph
  - Application specific

Surfaces

- What makes a good surface representation?
  - Accurate
  - Concise
  - Intuitive specification
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections

Parametric Surfaces

- Boundary defined by parametric functions:
  - \( x = f_x(u,v) \)
  - \( y = f_y(u,v) \)
  - \( z = f_z(u,v) \)

- Example: ellipsoid
  - \( x = r_c \cos \phi \cos \theta \)
  - \( y = r_c \cos \phi \sin \theta \)
  - \( z = r_s \sin \phi \)

- Example: surface of revolution
  - Take a curve and rotate it about an axis

Parametric & Implicit Surfaces

Adam Finkelstein & Tim Weyrich
Princeton University
COS 426, Spring 2008
Parametric Surfaces

- Example: swept surface
  - Sweep one curve along path of another curve

Demetri Terzopoulos

Parametric Surfaces

- Example: swept surface
  - Making sea shells

Fowler

Parametric Surfaces

- How do we describe arbitrary smooth surfaces with parametric functions?
  
H&B Figure 10.46

Parametric Patches

- Each patch is defined by blending control points
  
FvDFH Figure 11.44

Parametric Patches

- Point Q(u,v) on the patch is the tensor product of parametric curves defined by the control points
  
Watt Figure 6.21

Piecewise Polynomial Parametric Surfaces

- Surface is partitioned into parametric patches:
  
Watt Figure 6.25

Same ideas as parametric splines!
Parametric Patches

• Point \( Q(u,v) \) on the patch is the tensor product of parametric curves defined by the control points.

\[ Q(u,v) = U M \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P_{41} & P_{42} & P_{43} & P_{44} \end{bmatrix} M^T V^T \]

where \( U = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \) and \( V = \begin{bmatrix} v^3 & v^2 & v & 1 \end{bmatrix} \).

B-Spline Patches

\[ Q(u,v) = U M_{b-spline} \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P_{41} & P_{42} & P_{43} & P_{44} \end{bmatrix} M_{b-spline}^T V \]

where \( M_{b-spline} = \begin{bmatrix} \frac{1}{6} & \frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\ \frac{1}{2} & -1 & \frac{1}{2} & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ \frac{1}{6} & \frac{1}{3} & \frac{1}{6} & 0 \end{bmatrix} \).
**Bezier Patches**

\[ Q(u, v) = U \mathbf{M}_{\text{Bezier}} \mathbf{V} \]

- \[ \mathbf{M}_{\text{Bezier}} = \begin{bmatrix}
-1 & 3 & -3 & 1 \\
3 & -6 & 3 & 0 \\
-3 & 3 & 0 & 0 \\
1 & 0 & 0 & 0
\end{bmatrix} \]

- Properties:
  - Interpolates four corner points
  - Convex hull
  - Local control

**Bezier Surfaces**

- Continuity constraints are similar to the ones for Bezier splines

**B-Spline Patches**

\[ Q(u, v) = U \mathbf{M}_{\text{B-Spline}} \mathbf{V} \]

- \[ \mathbf{M}_{\text{B-Spline}} = \begin{bmatrix}
-\frac{1}{6} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{6} \\
\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & 0 \\
-\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\frac{1}{6} & \frac{1}{3} & \frac{1}{3} & 0
\end{bmatrix} \]

- \[ \mathbf{M}_{\text{B-Spline}} = \begin{bmatrix}
\frac{1}{2} & -1 & \frac{1}{2} & 0 \\
-\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\frac{1}{6} & -\frac{1}{3} & \frac{1}{3} & 0
\end{bmatrix} \]

- \[ \mathbf{M}_{\text{B-Spline}} = \begin{bmatrix}
\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & 0 \\
-\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\frac{1}{6} & -\frac{1}{3} & \frac{1}{3} & 0
\end{bmatrix} \]

- \[ \mathbf{M}_{\text{B-Spline}} = \begin{bmatrix}
\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & 0 \\
-\frac{1}{2} & 0 & \frac{1}{2} & 0 \\
\frac{1}{6} & -\frac{1}{3} & \frac{1}{3} & 0
\end{bmatrix} \]
Parametric Surfaces

• Advantages:
  ▪ Easy to enumerate points on surface
  ▪ Possible to describe complex shapes

• Disadvantages:
  ▪ Control mesh must be quadrilaterals
  ▪ Continuity constraints difficult to maintain
  ▪ Hard to find intersections

3D Object Representations

• Points
  ▪ Range image
  ▪ Point cloud

• Surfaces
  ▪ Polygonal mesh
  ▪ Subdivision
  ▪ Parametric
  ▪ Implicit

• Solids
  ▪ Voxel
  ▪ BSP tree
  ▪ CSG
  ▪ Sweep

• High-level structures
  ▪ Scene graph
  ▪ Application specific

Implicit Surfaces

• Represent surface with function over all space

Implicit Surfaces

• Surface defined implicitly by function:
  ▪ \( f(x, y, z) = 0 \) (on surface)
  ▪ \( f(x, y, z) < 0 \) (inside)
  ▪ \( f(x, y, z) > 0 \) (outside)

\[ f(x, y) = 0 \text{ on curve} \]
\[ f(x, y) < 0 \text{ inside} \]
\[ f(x, y) > 0 \text{ outside} \]

Implicit Surfaces

• Normals defined by partial derivatives
  ▪ \( \text{normal}(x, y, z) = (df/dx, df/dy, df/dz) \)
Implicit Surface Properties

(1) Efficient check for whether point is inside
- Evaluate \( f(x,y,z) \) to see if point is inside/outside/on

\[
\left( \frac{x}{r_x} \right)^2 + \left( \frac{y}{r_y} \right)^2 + \left( \frac{z}{r_z} \right)^2 - 1 = 0
\]

H&B Figure 10.10

(2) Efficient surface intersections
- Substitute to find intersections
- Ray: \( P = P_0 + tv \)
- Sphere: \( |P - O| ^ 2 - r ^ 2 = 0 \)

Substituting for \( P \), we get:

\[
|P_0 + tv - O| ^ 2 - r ^ 2 = 0
\]

Solve quadratic equation:

\[
a t^2 + bt + c = 0
\]

where:

\[
a = 1 \\
b = 2 V \cdot (P_0 - O) \\
c = |P_0 - O|^2 - r^2 = 0
\]

Implicit Surface Properties

(3) Efficient boolean operations (CSG)
- Union, difference, intersect

Union
Difference

(4) Efficient topology changes
- Surface is not represented explicitly!

Comparison to Parametric Surfaces

- Implicit
  - Efficient intersections & topology changes

- Parametric
  - Efficient "marching" along surface & rendering

\[
p = (\cos(\alpha), \sin(\alpha)), \alpha \in [0, 2\pi]
\]

\[
equangular parametric (transcendental trigonometric) \quad p = (\cos(\alpha), \sin(\alpha)), \alpha \in [0, 2\pi]
\]

\[
non-equangular parametric (rational) \quad p = ((1-t^\alpha)(1+t^\beta), 2t(1+t^\beta)), t \in [-1, 1]
\]

\[
\text{Implicit: } P \cdot P' - 1 = 0
\]
Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Blobby models
  - Skeletons
  - Procedural
  - Samples
  - Variational

Algebraic Surfaces

- Implicit function is polynomial
  - \( f(x,y,z) = ax^d + by^d + cz^d + dx^d-1y + dx^d-1z + dy^d-1x + ... \)
  - \( \left( \frac{x}{r_x} \right)^2 + \left( \frac{y}{r_y} \right)^2 + \left( \frac{z}{r_z} \right)^2 - 1 = 0 \)

Algebraic Surfaces

- Higher degree algebraics
  - Cubic
  - Quartic
  - Degree six

Algebraic Surfaces

- Most common form: quadrics
  - \( f(x,y,z) = ax^2 + by^2 + cz^2 + 2dxy + 2eyz + 2fxz + 2gx + 2hy + 2jz + k \)
  - Examples
    - Sphere
    - Ellipsoid
    - Torus
    - Paraboloid
    - Hyperboloid

Algebraic Surfaces

- Function extends to infinity
  - Must trim to get desired patch (this is difficult!)
Algebraic Surfaces

- Equivalent parametric surface
  - Tensor product patch of degree m and n curves yields algebraic function with degree 2mn

Bicubic patch has degree 18!

Algebraic Surfaces

- Intersection
  - Intersection of degree m and n algebraic surfaces yields curve with degree mn

Intersection of bicubic patches has degree 324!

Implicit Surface Representations

- How do we define implicit function?
  - Algebras
  - Blobby models
  - Skeletons
  - Procedural
  - Samples
  - Variational

Blobby Models

- Implicit function is sum of spherical basis functions

Blobby Models

- Sum of two blobs

Blobby Models

- Sum of four blobs
Blobby Models: radial basis func

- Blobby molecules
  \[ D(r) = a e^{-\alpha r} \]

- Meta balls
  \[ D(r) = \begin{cases} \frac{3r^2}{2} - \frac{r^3}{3} & r < a \varepsilon \sigma b \varepsilon b \\ 0 & r \geq a \varepsilon \sigma b \varepsilon b \end{cases} \]

- Soft objects
  \[ D(r) = \begin{cases} \frac{4r^6}{9a^6} - \frac{15r^5}{30a^5} + \frac{15r^4}{16a^4} - \frac{11r^3}{30a^3} + \frac{22r^2}{90a^2} - \frac{3r}{10a} & r < a \varepsilon b \varepsilon b \\ 0 & r \geq a \varepsilon b \varepsilon b \end{cases} \]

Blobby Model of Face

(a) \( N = 1 \)
(b) \( N = 2 \)
(c) \( N = 10 \)
(d) \( N = 35 \)
(e) \( N = 70 \)
(f) \( N = 243 \)

Blobby Model of Head

(a) \( N = 1 \)
(b) \( N = 2 \)
(c) \( N = 20 \)
(d) \( N = 60 \)
Blobby Model of Head

Blobby Models

Objects resulting from CSG of implicit soft objects and other primitives

Implicit Surface Representations

• How do we define implicit function?
  o Algebraics
  o Blobby models
  ➔ Skeletons
  o Procedural
  o Samples
  o Variational

Skeletons

• Bulge problem

Skeletons

• Bulge problem

Skeletons

• Convolution surfaces

Bloomenthal
Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Blobby models
  - Skeletons
  - Procedural
  - Samples
  - Variational

Procedural Implicits

- $f(x,y,z)$ is result of procedure
  - Example: Mandelbrot set

Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Blobby models
  - Skeletons
  - Procedural
  - Samples
  - Variational

Sampled Functions

- Most common example: voxels
  - Interpolate samples stored on regular grid
  - Isosurface at $f(x,y,z) = 0$ defines surface

Sampled Functions

- Acquired from simulations or scans
  - Airflow Inside a Thunderstorm (Bob Wilhelmson, University of Illinois at Urbana-Champaign)
  - Visible Human (National Library of Medicine)

Implicit Surface Representations

- How do we define implicit function?
  - Algebraics
  - Blobby models
  - Skeletons
  - Procedural
  - Samples
  - Variational
Variational Implicit Surfaces

Example Implicit Surface

Implicit Surface Summary

- Advantages:
  - Easy to test if point is on surface
  - Easy to compute intersections/unions/differences
  - Easy to handle topological changes

- Disadvantages:
  - Indirect specification of surface
  - Hard to describe sharp features
  - Hard to enumerate points on surface
    » Slow rendering

Summary

<table>
<thead>
<tr>
<th>Feature</th>
<th>Polygonal Mesh</th>
<th>Implicit Surface</th>
<th>Parametric Surface</th>
<th>Subdivision Surface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accurate</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Concise</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Intuitive specification</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Local support</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Affine invariant</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Arbitrary topology</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Guaranteed continuity</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Natural parameterization</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Efficient display</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Efficient intersections</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>