Shape Representation

Dr. Marios Papas
marios@disneyresearch.com
Why shape representation?

Computer Graphics

general shape description, motion/animation

appearance, texture mapping

rendering
Digital shapes are pervasive

Games/Movies

Engineering/Product design

Medicine/Biology

Architecture
Shape representation: origin- and application-dependent

- Acquired real-world objects:
  - Discrete sampling
  - Points, meshes

- Modeling “by hand”:
  - Higher-level representations, amendable to modification, control
  - Parametric surfaces, subdivision surfaces, implicits

- Procedural modeling
  - Algorithms, grammars
Similar to the 2D image domain

• Acquired digital images:
  – Discrete sampling
  – Pixels on a grid

• Painting “by hand”:
  – Strokes + color/shading
  – Vector graphics
  – Controls for editing
Similar to the 2D image domain

• Acquired digital images:
  – Discrete sampling
  – Pixels on a grid

• Painting “by hand”:
  – Strokes + color/shading
  – Vector graphics
  – Controls for editing
Representation considerations

• How should we represent geometry?
  – Needs to be stored in the computer
  – Creation of new shapes
    • Input metaphors, interfaces…
  – What operations do we apply?
    • Editing, simplification, smoothing, filtering, repair…
  – How to render it?
    • Rasterization, raytracing…
Shape representations

- Points
- Polygonal meshes
Shape representations

- Parametric surfaces
- Implicit functions
Parametric Curves and Surfaces
Smooth shape representation
Parametric representation

- Range of a function $s : X \rightarrow Y, \ X \subseteq \mathbb{R}^m, \ Y \subseteq \mathbb{R}^n$
  - Planar curve: $m = 1, n = 2$
    $$s(t) = (x(t), y(t))$$
  - Space curve: $m = 1, n = 3$
    $$s(t) = (x(t), y(t), z(t))$$
Parametric representation

- Range of a function \( s : X \rightarrow Y, \ X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n \)

  - Surface in 3D: \( m = 2, n = 3 \)

\[
s(u, v) = (x(u, v), y(u, v), z(u, v))
\]
Parametric representation

- Range of a function \( s : X \to Y, \quad X \subseteq \mathbb{R}^m, Y \subseteq \mathbb{R}^n \)

- Volumetric density in 3D: \( m = 3, n = 1 \)

http://vis.lbl.gov/

Bruckner et al.
Parametric curves

- Explicit curve/circle in 2D

\[ p : \mathbb{R} \rightarrow \mathbb{R}^2 \]
\[ t \mapsto p(t) = (x(t), y(t)) \]

\[ p(t) = r \left( \cos(t), \sin(t) \right) \]
\[ t \in [0, 2\pi) \]
Parametric curves

- Bezier curves

\[ s(t) = \sum_{i=0}^{n} p_i B_i^n(t) \]

Curve and control polygon

\[ B_i^n(t) = \binom{n}{i} t^i (1 - t)^{n-i} \]

Basis functions
Parametric surfaces

- Sphere in 3D

\[ s : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \]

\[ s(u, v) = r \left( \cos(u) \cos(v), \sin(u) \cos(v), \sin(v) \right) \]

\[ (u, v) \in [0, 2\pi) \times [-\pi/2, \pi/2] \]
Parametric surfaces

• Curve swept by another curve

\[ s(u, v) = \sum_{i,j} p_{i,j} B_i(u) B_j(v) \]

• Bezier surface:

\[ s(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} p_{i,j} B_i^m(u) B_j^n(v) \]
Tangents and normal

\[ s_u = \frac{\partial s(u, v)}{\partial u} \]

\[ s_v = \frac{\partial s(u, v)}{\partial v} \]

\[ n = \frac{s_u \times s_v}{\|s_u \times s_v\|} \]

Tangent plane is normal to \( n \)
Parametric curves and surfaces

• Advantages
  – Easy to generate points on the curve/surface
  – Analytic formulas for derivatives

• Disadvantages
  – Hard to determine inside/outside
  – Hard to determine if a point is on the curve/surface
Implicit Curves and Surfaces
Implicit curves and surfaces

• Kernel of a scalar function \( f : \mathbb{R}^m \rightarrow \mathbb{R} \)
  – Curve in 2D: \( S = \{ x \in \mathbb{R}^2 | f(x) = 0 \} \)
  – Surface in 3D: \( S = \{ x \in \mathbb{R}^3 | f(x) = 0 \} \)

• Space partitioning

\[
\begin{align*}
\{ x \in \mathbb{R}^m | f(x) > 0 \} & \quad \text{Outside} \\
\{ x \in \mathbb{R}^m | f(x) = 0 \} & \quad \text{Curve/Surface} \\
\{ x \in \mathbb{R}^m | f(x) < 0 \} & \quad \text{Inside}
\end{align*}
\]
Implicit curves and surfaces

• Kernel of a scalar function \( f : \mathbb{R}^m \rightarrow \mathbb{R} \)
  – Curve in 2D: \( S = \{ x \in \mathbb{R}^2 | f(x) = 0 \} \)
  – Surface in 3D: \( S = \{ x \in \mathbb{R}^3 | f(x) = 0 \} \)

• Zero level set of a signed distance function
Implicit curves and surfaces

- Implicit circle and sphere

\[ f(x, y) = x^2 + y^2 - r^2 \]

\[ f(x, y, z) = x^2 + y^2 + z^2 - r^2 \]
Implicit curves and surfaces

- The normal vector to the surface (curve) is given by the gradient of the implicit function

\[ \nabla f(x, y, z) = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right)^T \]

- Example

\[ f(x, y, z) = x^2 + y^2 + z^2 - r^2 \]

\[ \nabla f(x, y, z) = (2x, 2y, 2z)^T \]
Boolean set operations

• Union: \( \bigcup_{i} f_i(x) = \min f_i(x) \)

• Intersection: \( \bigcap_{i} f_i(x) = \max f_i(x) \)
Boolean set operations

- Positive = outside, negative = inside
- Boolean subtraction: \( h = \max(f, -g) \)

<table>
<thead>
<tr>
<th></th>
<th>( f &gt; 0 )</th>
<th>( f &lt; 0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( g &gt; 0 )</td>
<td>( h &gt; 0 )</td>
<td>( h &lt; 0 )</td>
</tr>
<tr>
<td>( g &lt; 0 )</td>
<td>( h &gt; 0 )</td>
<td>( h &gt; 0 )</td>
</tr>
</tbody>
</table>

- Much easier than for parametric surfaces!
Smooth set operations

• In many cases, smooth blending is desired
  – Pasko and Savchenko [1994]

\[
f \cup g = \frac{1}{1+\alpha} \left( f + g - \sqrt{f^2 + g^2 - 2\alpha fg} \right)
\]

\[
f \cap g = \frac{1}{1+\alpha} \left( f + g + \sqrt{f^2 + g^2 - 2\alpha fg} \right)
\]
Smooth set operations

- **Examples**

  - For $\alpha = 1$, this is equivalent to $\min$ and $\max$

  \[
  \lim_{\alpha \to 1} f \cup g = \frac{1}{2} \left( f + g - \sqrt{(f - g)^2} \right) = \frac{f + g}{2} - \frac{|f - g|}{2} = \min(f, g)
  \]

  \[
  \lim_{\alpha \to 1} f \cap g = \frac{1}{2} \left( f + g + \sqrt{(f - g)^2} \right) = \frac{f + g}{2} + \frac{|f - g|}{2} = \max(f, g)
  \]
Designing with implicit surfaces

- Zero set or other level set of a function:

\[ f(p) = \|p\|^2 - r^2 \]

\[ f(p) = e^{-\|p\|^2/r^2} \text{ at } e^{-1} \]
Designing with implicit surfaces

• With smooth fall-off functions, adding implicit functions generates a blend:

\[
f(p) = e^{-\|p-p_1\|^2} + e^{-\|p-p_2\|^2}
\]

• Called “Metaballs” or “Blobs”
Designing with implicit surfaces

• With smooth fall-off functions, adding implicit functions generates a blend:

\[ f(p) = e^{-\|p - p_1\|^2} + e^{-\|p - p_2\|^2} \]

• Called “Metaballs” or “Blobs”
Blobs

• Suggested by Blinn [1982]
  – Defined implicitly by a potential function around a point $p_i$: $f(p) = b_i e^{-a_i \|p-p_i\|^2}$
  – Set operations by simple addition/subtraction
Blobs in Films
Procedural Implicits

- Combine multiple operations into a tree
  [Wyvill et al. 1999]
Implicit curves and surfaces

• Advantages
  – Easy to determine inside/outside
  – Easy to determine if a point is on the curve/surface

• Disadvantages
  – Hard to generate points on the curve/surface
  – Does not lend itself to (real-time) rendering
## Surface representation zoo!

<table>
<thead>
<tr>
<th>Parametric</th>
<th>Implicit</th>
<th>Discrete/Sampled</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="spline.png" alt="Spline example" /></td>
<td><img src="implicit.png" alt="Implicit example" /></td>
<td><img src="discrete.png" alt="Discrete example" /></td>
</tr>
<tr>
<td>• Splines, tensor-product surfaces</td>
<td>• Metaballs/blobs</td>
<td>• Meshes</td>
</tr>
<tr>
<td>• Subdivision surfaces</td>
<td>• Distance fields</td>
<td>• Point set surfaces</td>
</tr>
<tr>
<td></td>
<td>• Procedural, CSG</td>
<td></td>
</tr>
</tbody>
</table>

- Parametric: Splines, tensor-product surfaces, Subdivision surfaces
- Implicit: Metaballs/blobs, Distance fields, Procedural, CSG
- Discrete/Sampled: Meshes, Point set surfaces
Literature

- Farin: *Curves and Surfaces for CAGD*, Morgan Kaufman, 2002
  The “bible” on parametric curves and surfaces

- Botsch et al., *Polygon Mesh Processing*, CRC Press, 2010
  Overview of shape representations and in-depth about meshes