Surface Rendering

• Introduce Mapping Methods
  - Texture Mapping
  - Environmental Mapping
  - Bump Mapping

• Go over strategies for
  - Forward vs backward mapping
The Limits of Geometric Modeling

- Although graphics cards can render over 10 million polygons per second, that number is insufficient for many phenomena:
  - Clouds
  - Grass
  - Terrain
  - Hair

Motivation: Modeling an Orange

- Consider the problem of modeling an orange (that is a SunKist, not the color)
- Start with an orange-colored sphere
  - Too simple
- Replace sphere with a more complex shape
  - To capture surface characteristics (small dimples)
  - Takes SO many polygons to model all the dimples
Motivation: Modeling an Orange (2)

• Take a picture of a real orange, scan it, and “paste” onto simple geometric model
  - This process is texture mapping
• Still might not be sufficient because resulting surface will be smooth
  - Need to change local shape
  - Bump mapping

Motivation: Modeling an Orange (3)

Detail surface variation
Noisy Surface
Patterned surface
Complex structure
Bumpy surface
Three Types of Mapping

• Texture Mapping
  - Uses images to fill inside of polygons

• Environmental (reflection mapping)
  - Uses a picture of the environment for texture maps
  - Allows simulation of highly specular surfaces

• Bump mapping
  - Emulates altering normal vectors during the rendering process

Texture Mapping

geometric model  texture mapped
Environment Mapping

Bump Mapping
Texture Mapping

• Geometry and lighting alone do not provide sufficient visible detail
• “Paste” 2D image onto 3D surface

• Surface appears much more complex than reality
Although the idea is simple---map an image to a surface---there are several tricky steps involved.
Mapping textures to models

- How do we find mapping?
- Consider going from texture coordinates to a point a surface
- Must define mapping

\[ x = x(s,t) \]
\[ y = y(s,t) \]
\[ z = z(s,t) \]

Texture "Painting"

- 3D Paint packages establishes an arbitrary mapping (model to map)
- Then records the texture as the user "paints" on the model
Rendering Textures

• In rendering, we perform backward mapping
  - Given a pixel, we want to know which point on an object it corresponds to and
  - Given a point on an object, we want to know which point in the texture it corresponds or:
    • Map of the form
      \[ s = s(x,y,z) \]
      \[ t = t(x,y,z) \]
    • These functions are difficult to find in general

Simple Backward mapping

• A direct solution to the simple mapping problem can be found analytically
• Example: mapping a cube
Simple Backward mapping

• Example: map to cylinder

\[ \begin{align*}
x &= r \cos 2\pi u \\
y &= r \sin 2\pi u \\
z &= v/h
\end{align*} \]
Maps rectangle in \( u,v \) space to cylinder of radius \( r \) and height \( h \) in world coordinates

Thus, \( s = u \) \hspace{1cm} \( t = v \)
maps from the texture space
Two-part (Backward) mapping

• One solution to the general mapping problem is a two-part mapping: first map the texture to a simple intermediate surface (like the cylinder)
• Then map to the object of interest

Two-Part Mapping

• Second, map from the intermediate object to the actual object by:
  - Normals from intermediate to actual
  - Normals from actual to intermediate
Two-Part Mapping

- In General, no direct method exists
- Best choice of secondary mapping depends on the model being mapped

Sampling

- Assume texture parameterized by $s,t$
Sampling the texture

• Associate texture coordinates with each vertex on the surface

Sampling the texture

• During drawing, look up color from texture using continuous texture coordinates
Sampling the texture

- Nearest neighbor
  - Blocky results
Nearest Sampling Example

Sampling

- Nearest neighbor
  - Blocky results
- Linear blending
  - Smooth appearance
Sampling

- Nearest neighbor
  - Blocky results
- Linear blending
  - Smooth appearance

Nearest Sampling Example
Linear Sampling Example

Other Problems with Texture Mapping

- Textures often composed of separate pieces called “charts”
- Boundaries don’t match
- Sampling issues

- Not bad for color
- Huge problem for displacement mapping
Other Uses of Texture Mapping

- Environment Mapping
- Bump/Normal Mapping
- Displacement Mapping
- ...

Any attribute of the surface position, normal, color, etc… can be placed in a texture

Environment Mapping

- Cheap attempt at modeling reflections
- Makes surfaces look metallic

- Use six textures to model faces of a cube
- Assume cube faces infinitely far away
- The normal (or reflected eye vector) is used to find which of the textures to use and what texture coordinate
Environment Mapping

Reflected ray: \( r = 2(n \cdot v)n - v \)

Texture is transferred in the direction of the reflected ray \( r \) from the environment map onto the object.

How to represent the map
How to represent the map
How to represent the map

- Store colors of every possible direction in texture maps
How to represent the map

- Store colors of every possible direction in texture maps
- Look up texture maps based on reflected vector

Environment Mapping
Environment Mapping

- Easy to use with simple projection for a box but can cause distortions
Bump/Normal Mapping

- Replace colors R,G,B with coordinates X,Y,Z
- Interpret pixels as adjusted normals
- Makes the shading look more complicated than geometry really is

After bump mapping

Bump Mapping (2D Example)

Surface with normals

Bump texture

“New” surface with texture
Bump Mapping

No bump mapping  With bump mapping

Bump/Normal Mapping Example
Bump Mapping combined with Texture
Displacement Mapping

• Offset geometry in direction of normal
• Encode offset inside texture
• Used to actually change the geometry and provide more detail (especially silhouette)

Bump/Normal Mapping Example
Displacement Mapping Example
More Examples

More Examples
Problems with Displacement Maps

Actually move geometry based on texture map – how is still open question in general

Expensive and difficult to implement in many rendering pipelines