Texture Mapping

Adapted from slides by Rich Riesenfeld

http://www.cs.utah.edu/classes/cs5600/
Texture Mapping

• Maps a pattern (texture) onto a surface
• Texels fill each pixel
• Texels selected from sample pattern (texture map)
• Pattern is often repeated
Texture Mapping Characteristics

- Too much *detail* to model geometrically, like grass, etc
- Pattern is repeated (*periodic*)
Texture Maps
Tiling textures
Tiling textures
Examples of Mapped Texture
Basic Concept  
(2D Texture maps)

• Relate a 2D image to a 3D model

• *Texture coordinates*
  – 2D coordinate $(u,v)$ that corresponds to a location in the texture image  
  – usually in range $[0,1]$
Elements of Texture Mapping

- Texture source function (1D, 2D or 3D)
- Inverse map:
  - texture location  surface location
- Typical texture sources
  - Procedure
  - Tabular data (texture image)
Texture Mapping Techniques

• 2D texture mapping: *paint 2D* pattern onto the surface

• *Environmental* (reflection) *mapping*

• *Bump mapping*: perturb surface normals to fool shading algorithms

• Procedural texture mapping
Need to Impose Parametrization

\[ (0,0,0) \quad (0,1,0) \quad (0,0,1) \quad (1,0,0) \quad (1,1,1) \]

\[ (h, \phi) \quad (\theta, \phi) \quad (\theta, 0) \quad (0,0.5) \quad (1,0.5) \]

\[ \phi \quad \theta \quad h \]
Using a planar projection
More Examples

Planar

Cubic

Cylindrical


http://astronomy.swin.edu.au/~pbourke/texture/texturemapping
Texture Mapped Teapot
Examples
Steps in Texture Mapping (OpenGL)

1. Create a texture object and specify a texture for that object
2. Indicate how the texture is to be applied to each pixel
3. Render the scene, supplying both texture and geometric coordinates
Mapping the $2D$ Texture to Surface

- The map: $2D$ texture$(s,t) \rightarrow 3D$ object$(x,y,z)$
- Mapping onto triangle is not difficult
- Mapping onto triangular mesh is more difficult (have to handle texture discontinuity)
- Mapping onto parametric surface is easier
- Alternative: use an intermediate parametric surface (cylinder, sphere)
Texture Mapping for Meshes

• Assign per-vertex texture coordinate
• During rasterization: interpolate texture coordinates at each pixel (similar to project 1)
• Lookup texture color via texture coordinate
Mapping Texture

Texture Space

Object Space

\[(s, t) = (0.2, 0.8)\]

\[(0.4, 0.2)\]

\[A\]

\[B\]

\[C\]

\[(0.8, 0.4)\]
Mapping Texture onto Parametric Surface

- Parametric surface:
  \[ S(u,v) = (x(u,v), y(u,v), z(u,v)) \]

- Use \((u,v)\) as texture coordinates
Using an intermediate surface

• Two-step mapping:
  – Map the texture to a simple intermediate surface (sphere, cylinder, cube)
  – Map the intermediate surface (with the texture) onto the surface being rendered
C : center of mesh
p : point on mesh
p' : projection of p onto sphere
p' = (p - c) / ||p - c|| (unit sphere)
spherical projection, cont'd

• starting with point $p$ on the surface
• project to point $p'$ on unit sphere:
  \[ p' = \frac{p - c}{|p - c|} \]
• now use $(x,y,z)$ coordinates of $p'$ to compute $(u,v)$ coordinates via the natural parameterization of the sphere
spherical projection, cont'd

• details of last step explained in project 2
MIP Mapping (*multum in parvo*)

“Many things in a small place”
Mipmapped Textures

• Mipmapping:
  – prefILTERED texture maps
  – decreasing resolutions
  – used to combat aliasing

• OpenGL supports mipmapping
Aliasing

classic (left); antialiased (middle, right)  
(from wikipedia)
Bump Mapping

smooth silhouettes
Bump mapping

- 2D texture maps don't interact with lighting
- Bump mapping: use texture map to define perturbed surface normals
- Compute lighting using perturbed normals
More Examples

Texture Mapping as a Fundamental Drawing Primitive
Paul Haeberli and Mark Segal

Contours indicate equidistance from reference plane

Environment Mapping
Projective Texture Mapping