Now to Texture mapping

Ubiquitous, and sometimes obvious

Image from http://www.minecraftercamp.com
Adding surface detail

• Surfaces in the world have appearance
  – They are seldom one flat color.
  – They have true texture – repeating patterns.
  – They have structured markings.
  – They have tiny changes in surface height.

• Purists view (don’t try this!)
  – Use ever more even smaller uniform triangles.

• Pragmatists view
  – Paint surfaces with images – texture mapping.
Texture Mapping

• Use projective geometry to compute where vertices appear in the image
• Apply shading to determine the color of pixels
  -- or --
• Map an existing texture onto a surface
  -- Textures supercede/augment the specification of surface material
  -- Leaves room for distinction diffuse vs. specular
Mapping

- Guess what? The underlying problem is to apply a geometric transformation
Mapping (II)

• Textures are color images
  – Logical texture coords run from (0,0) to (1,1)
  – Coordinates fixed regardless of image size

• Polygons are 2D surfaces in a 3D space

• The transformation from texture coordinates to surface coordinates is expressed as – you guessed it – a matrix
Texture Matrices

• Given vertices and the corresponding texture coordinates...

\[
\begin{bmatrix}
  t \\
  s
\end{bmatrix} = \begin{bmatrix}
  a & b & c \\
  d & e & f
\end{bmatrix} \begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix}
\]

- How many correspondences are needed?
- Maps from surface to texture
Worked Example - Before

• Consider point correspondences
  – Pairs of points in texture and 3-D coordinates
  – Three such pairs of points yield six constraints
  – Constraints match free variables – six.

• Specifically
  – Point (0,0) matches point (1,2,3)
  – Point (1,1) matches point (2,2,2)
  – Point (0,1) matches point (3,2,2)
Worked Example

Match #1
\[
\begin{vmatrix} 0 \\ 0 \end{vmatrix} = \begin{vmatrix} a & b & c \\ d & e & f \end{vmatrix} \begin{array}{c} 1 \\ 2 \\ 3 \end{array}
\]

Match #2
\[
\begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix} = \begin{vmatrix} a & b & c \\ d & e & f \end{vmatrix} \begin{array}{c} 2 \\ 2 \\ 2 \end{array}
\]

\[0 = a + b + c\]
\[1 = 2a + 2b + 2c\]
\[0 = 3a + 2b + 3c\]

3 equations for first texture coords

Linear Alg. Setup
\[
\begin{vmatrix} 0 \\ 1 \end{vmatrix} = \begin{vmatrix} 1 & 2 & 3 \\ 2 & 2 & 2 \end{vmatrix} \begin{array}{c} a \\ b \\ c \end{array}
\]
\[
\begin{vmatrix} 1 & 1 \\ 1 & 1 \end{vmatrix} = \begin{vmatrix} 1 & 2 & 3 \\ 2 & 2 & 2 \end{vmatrix} \begin{array}{c} d \\ e \\ f \end{array}
\]

Solved in Maple
\[a = 0, \quad b = \frac{3}{2}, \quad c = -1\]
\[d = \frac{1}{2}, \quad e = \frac{1}{2}, \quad f = -\frac{1}{2}\]
Texture Mapping (II)

- The fragment processor computes a reflectance color for every pixel.
- When textures are enabled,
  - The fragment processor also computes a texture value for every pixel.
  - Using the pixel to do texture mapping.
- These values are multiplied together to produce the final value.
Issue #1: Sampling

• The mapping from surface points to texture coordinates produces real values
Sampling

• Nearest-neighbor:
  – pick the closest texture pixel

• Bilinear:
  – linearly interpolate in both dimensions

• Bicubic:
  – fit a 3rd order surface to 16 surrounding points
  – Not as expensive as it sounds
Sampling (III)

• A better solution is for the texture map to be roughly the same size as the surface projection.

• A MipMap is an image pyramid built from a texture map
  – Example: if the texture is 64x64, the pyramid also includes 32x32, 16x16, etc.
Issue #2: Getting Textures

- **WEB!**
  - **millions** of textures – people use them for backgrounds of web pages a lot! You can download them in bulk packages, etc..

- **Build your own**
  - Make them “seamless”
    - When tiled, you cannot see the edges of the tiles.
Just for example ...
Texture Makers

• There are tons of them. Some examples:
  – http://www.backgroundmagic.com/software/BGM.zip
  – http://216.156.212.112/photoseam.exe
Example in SketchUp - Cube
Import a Texture Image
Place Texture on Face
View the Result
Position/Reset-Position
The Essence of Tiling
View the Result
Support for ‘Painting’ Textures
Final Result – Textured Cube
Alas - .obj support marginal

- Texture vertices do come through.
- File linkage to texture map does not.
More Examples

Surface

Textures
Texture Map from Image
Texture in Blender

Credit where credit is due – YouTube video: “Blender 2.8 Beginner Textures and Materials Tutorial”