There are limits to geometric modeling

Although modern GPUs can render millions of triangles/sec, that’s not enough sometimes...
Use texture mapping to increase realism through detail

This image is just 8 polygons!

Add visual complexity.

http://www.siggraph.org/education/materials/HyperGraph/mapping/r_wolfe/r_wolfe_mapping_1.htm
No texture

With texture
Pixar - Toy Story
Store 2D images in buffers and lookup pixel reflectances

Textures can be anything that you can lookup values in -- photo, procedurally generated, or even a function that computes a value on the fly
3D solid textures
Other uses of textures...

Light maps
Shadow maps
Environment maps
Bump maps
Opacity maps
Animation

[Angel and Shreiner]
[Stam 99]
Texture mapping in the OpenGL pipeline

- Geometry and pixels have separate paths through pipeline
- meet in **fragment processing** - where textures are applied
- texture mapping applied at end of pipeline - efficient since relatively few polygons get past clipper
uv Mapping

- Texture is parameterized by \((u,v)\)
- Assign polygon vertices texture coordinates
- Interpolate within polygon

Texture coordinates are per-vertex data – a position in the \((u,v)\) space can interpolate tex coordinates with barycentric coordinates.
Texture Calibration
The major issues in texture mapping...

- What should the actual mapping be?

   easy: rectangular surface
   harder: parametric surface

Teapot: Which image looks better? The image on the left uses **object coordinates** in the texture mapping – this makes more sense. The image on the right uses **world coordinates** – texture ends up changing relative to the object. We want a nice map that doesn’t look distorted.
Given a point on the object \((x,y,z)\), what point \((u,v)\) in the texture we use?
Example: planar mapping

[Image of a 3D model with colorful, checkered objects and an XYZ axis]

[Rosalee Wolfe]
Intermediate surfaces

First map the texture to a simpler, intermediate surface