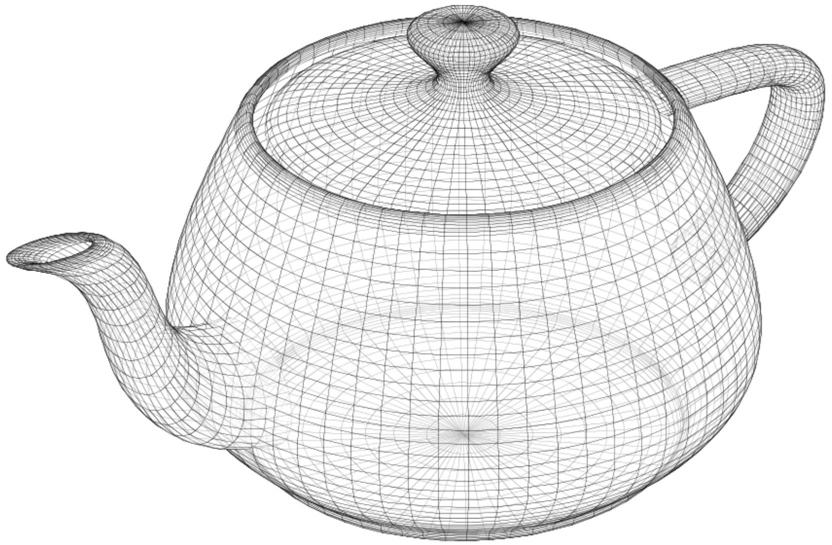
Texture Mapping



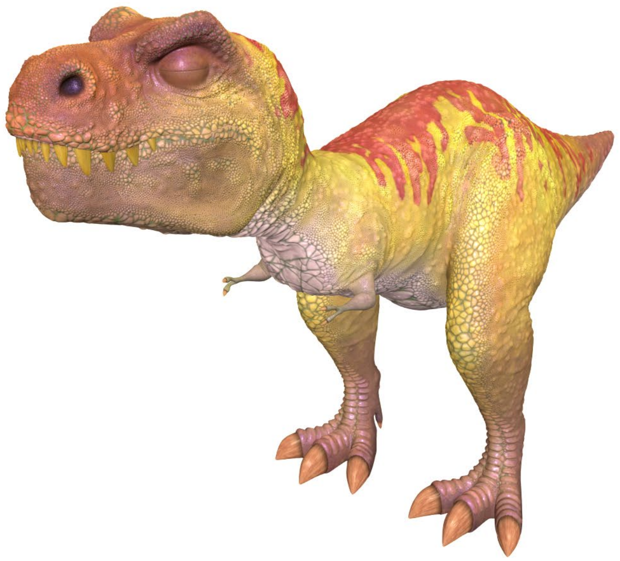
Interactive Computer Graphics

Professor Eric Shaffer

"All it takes is for the rendered image to look right."
—Jim Blinn

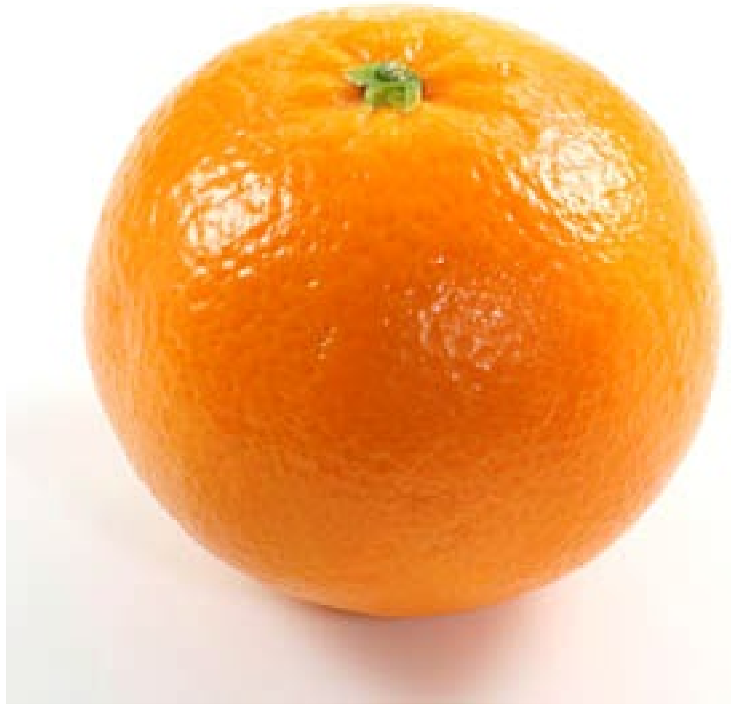
The Limits of Geometric Modeling

- Graphics cards can render over 20 billion polygons per second
  (NVIDIA 2080 TI in 2018)
- That number is still insufficient for many visual phenomena
- Consider rendering a herd of 100s of bumpy-skinned dinosaurs



Or Consider Modeling an Orange

- Start with an orange-colored sphere
  - Too simple
- Replace sphere with a more complex shape
  - Does not capture surface characteristics (small dimples)
  - Takes too many polygons to model all the dimples



Modeling an Orange

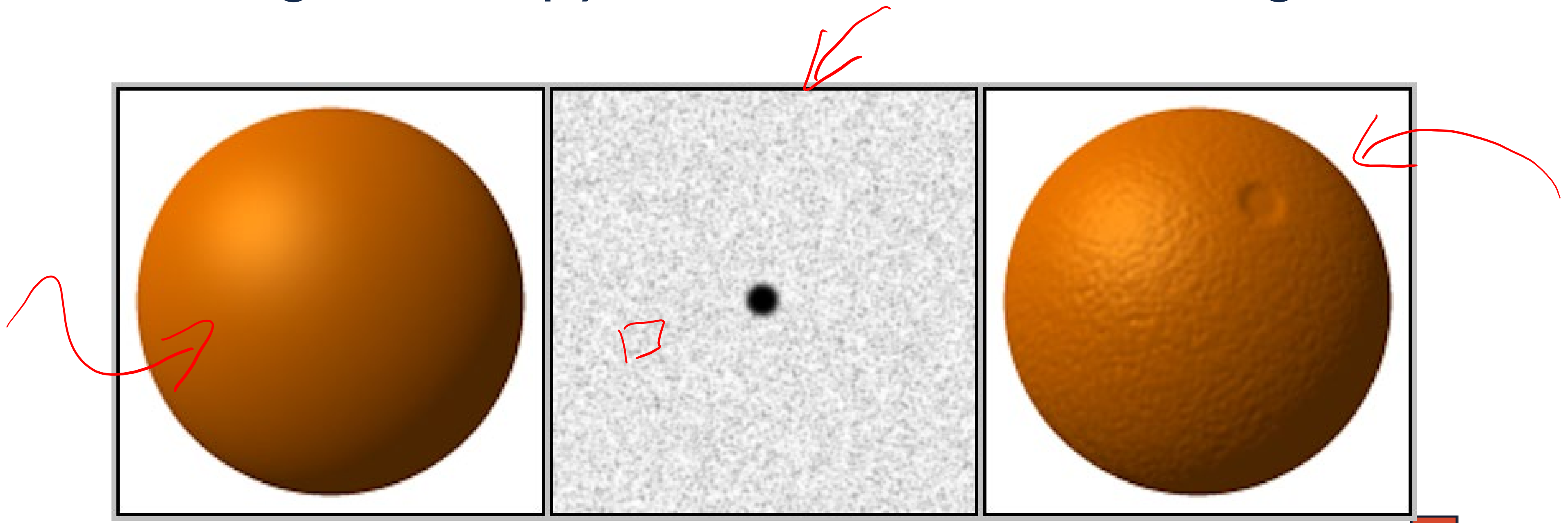
- Take a picture of a real orange
- “paste” pixels of the image onto simple geometric model
  - This process is known as texture mapping
  - Specifically *image texturing*
- Still might be problematic…
  - Looking at the orange in a rendered scene
  - How could you tell the colors on the orange aren’t generated by shading?

Modeling an Orange

Another alternative would be ***bump mapping***

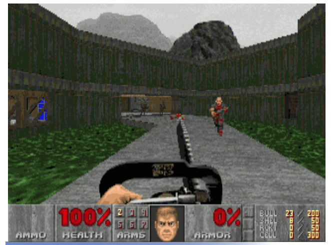
Use an image that specifies the normal to use to shade the surface

- This way, can render an "bumpy" surface during shading
- Without modeling the bumpy surface with lots of triangles



Some Types of Texture Mapping

- Image Texturing
  - Uses images to fill inside of polygons
- Environment Mapping
  - Uses a picture of the environment for texture maps
  - Allows simulation of mirror-like surfaces
- Bump mapping
  - Alters normal vectors during the rendering process
  - Generates a bumpy looking surface



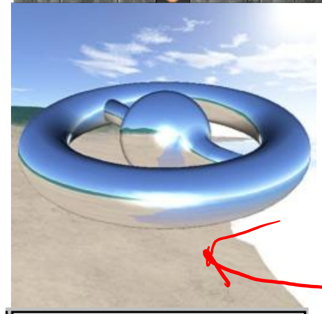
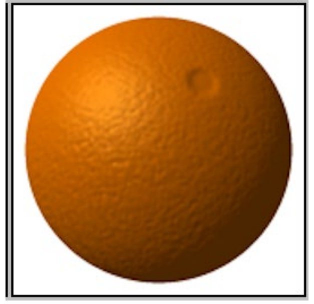
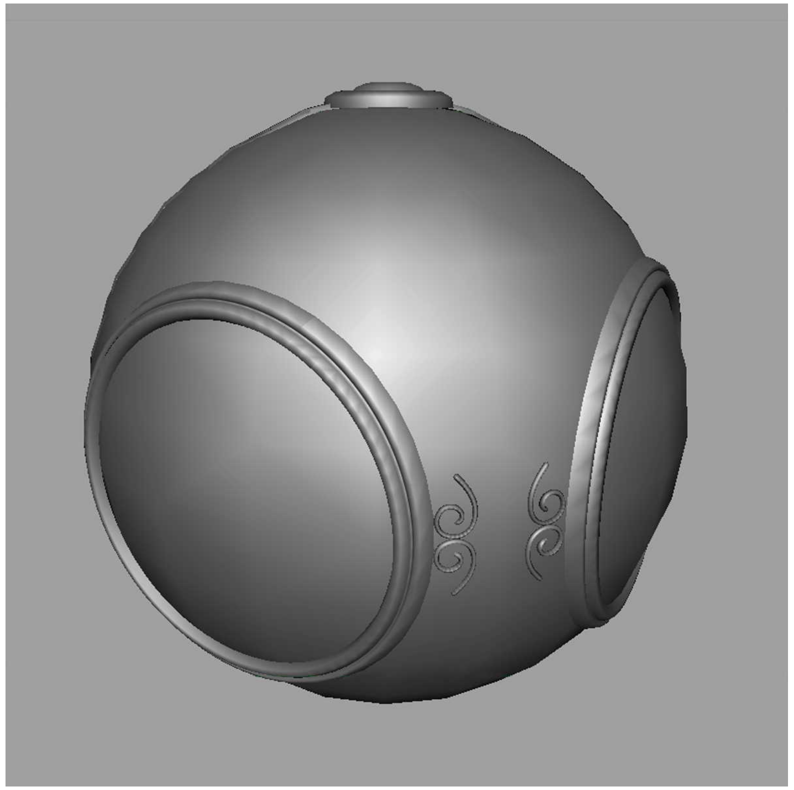
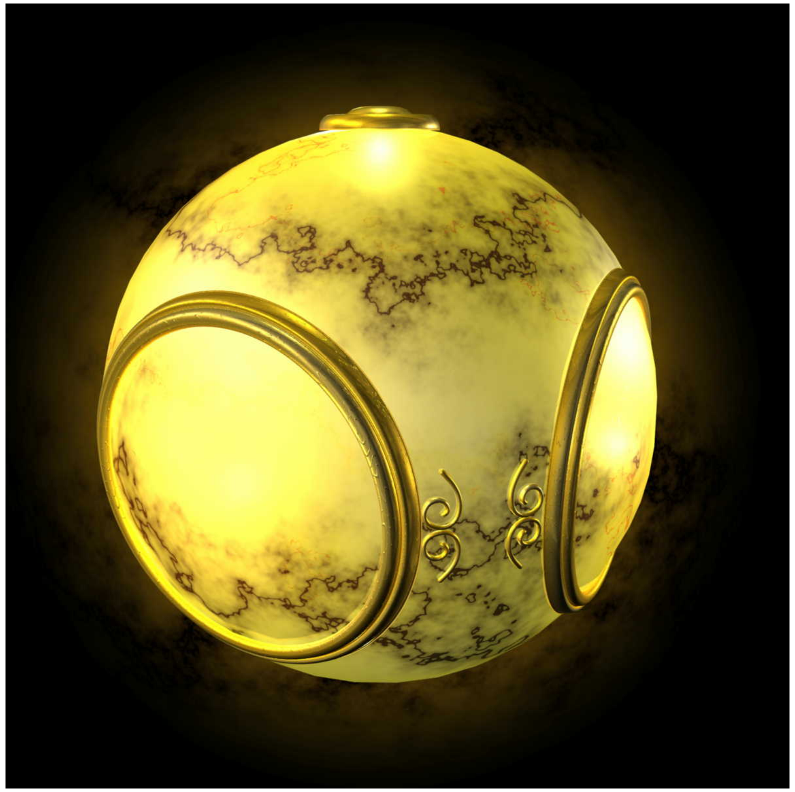
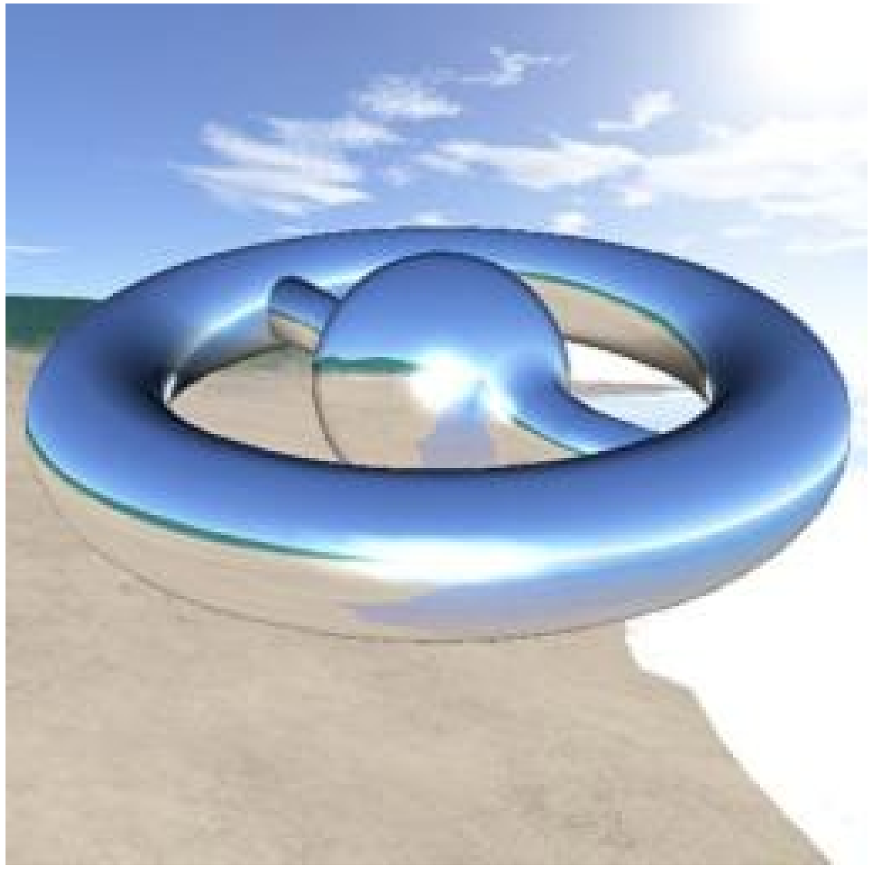



Image Texturing

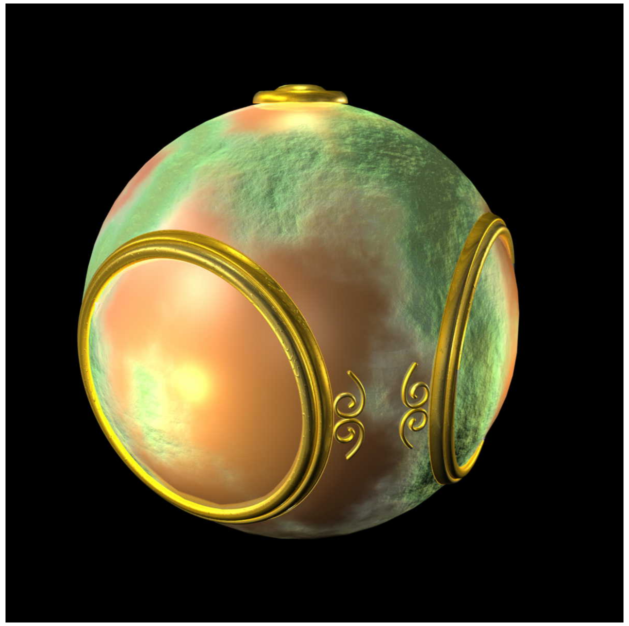

geometric model

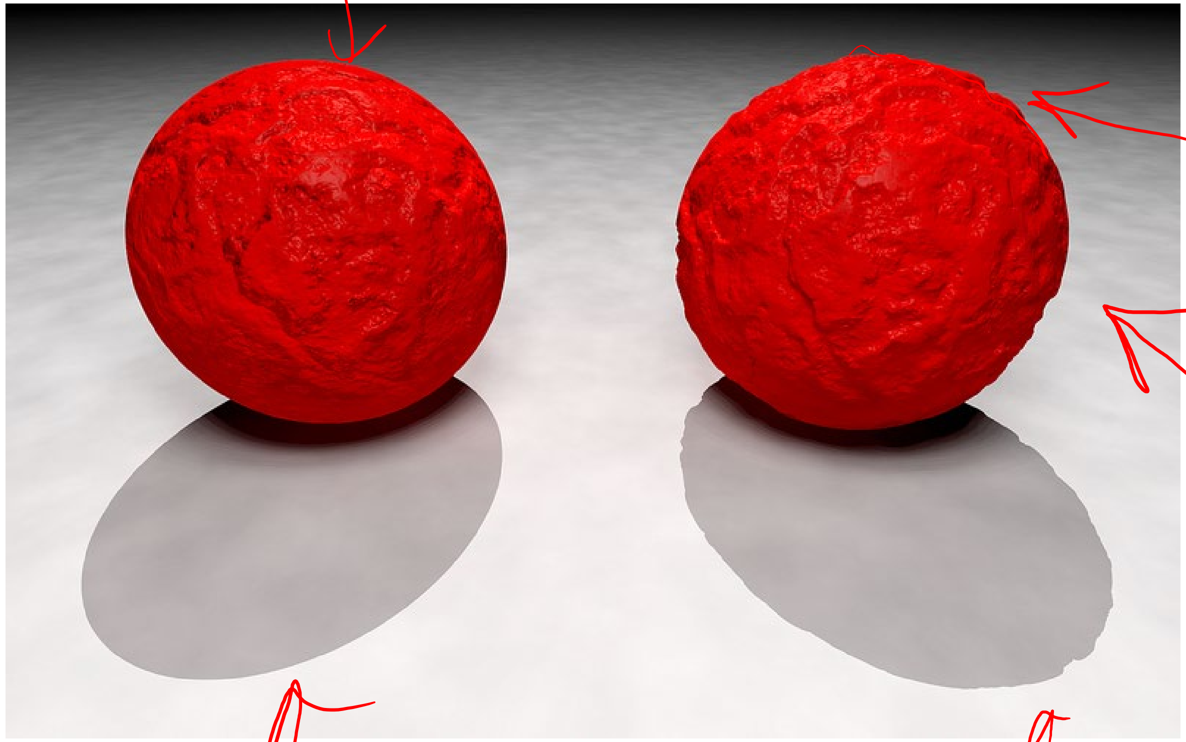

texture mapped

Environment Mapping



Bump Mapping





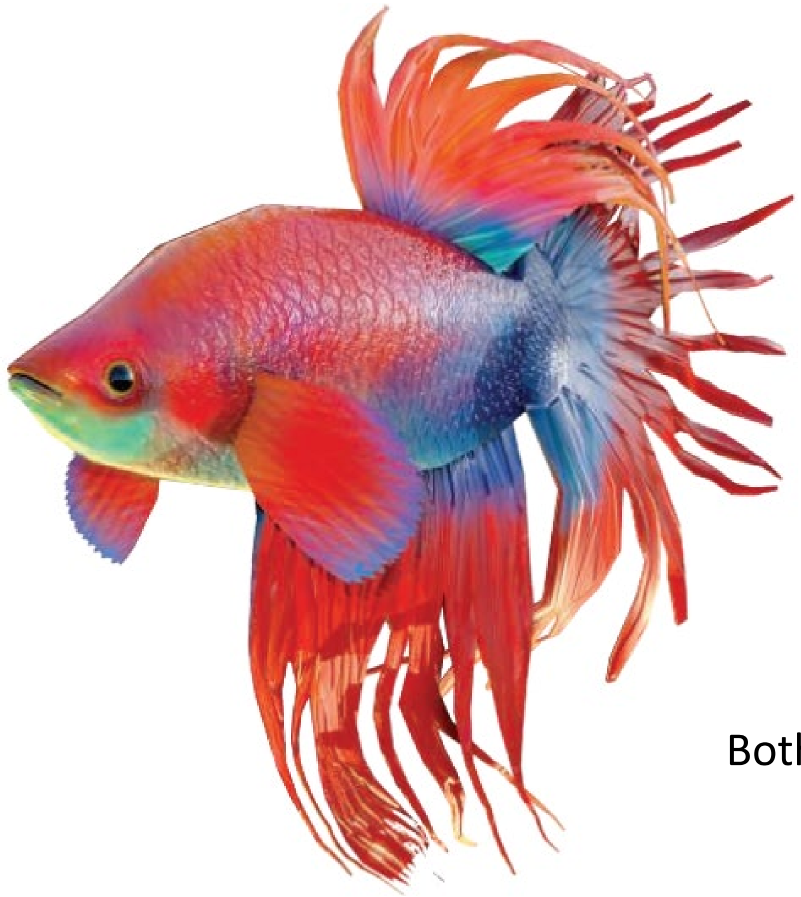
ILLINOIS

So What is Texture?

"A surface's texture is its look and feel—just think of the texture of an oil painting. In computer graphics, texturing is a process that takes a surface and modifies its appearance at each location using some image, function, or other data source. As an example, instead of precisely representing the geometry of a brick wall, a color image of a brick wall is applied to a rectangle, consisting of two triangles. When the rectangle is viewed, the color image appears where the rectangle is located. Unless the viewer gets close to the wall, the lack of geometric detail will not be noticeable."

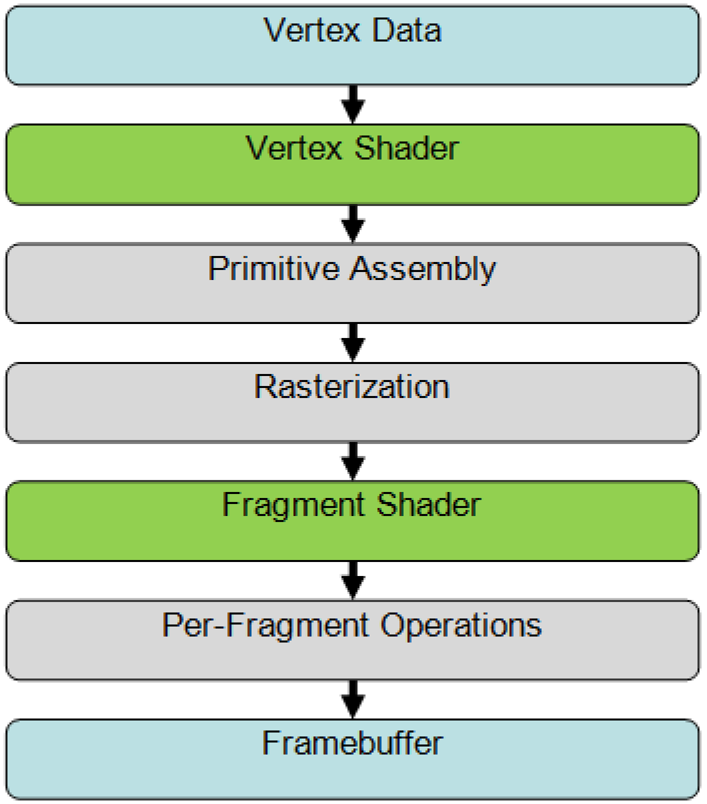
-- Akenine-Moeller, Tomas; Haines, Eric; Hoffman, Naty.
Real-Time Rendering, Fourth Edition

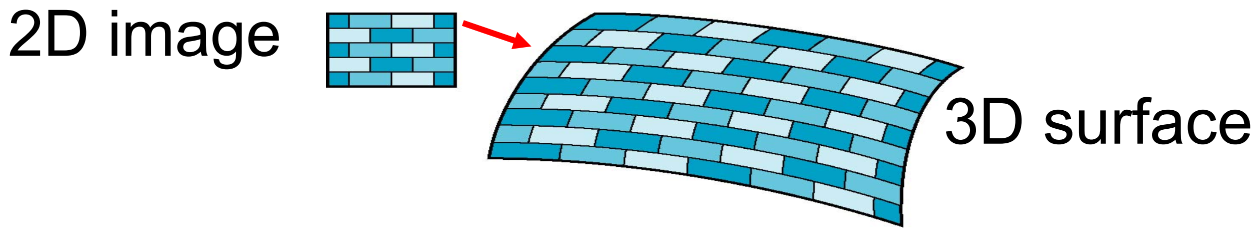
For example



Both image texturing and bump mapping are used here

Texturing Mapping and the Pipeline





In the simplest form texture mapping is a process in which

- a fragment on a surface to be rendered
- is mapped to a color in a texture (an image)
- and then this color (called a texel) is used in shading

Mapping is implemented in the fragment shader

- Efficient because few polygons make it past the clipper

Texture Mapping

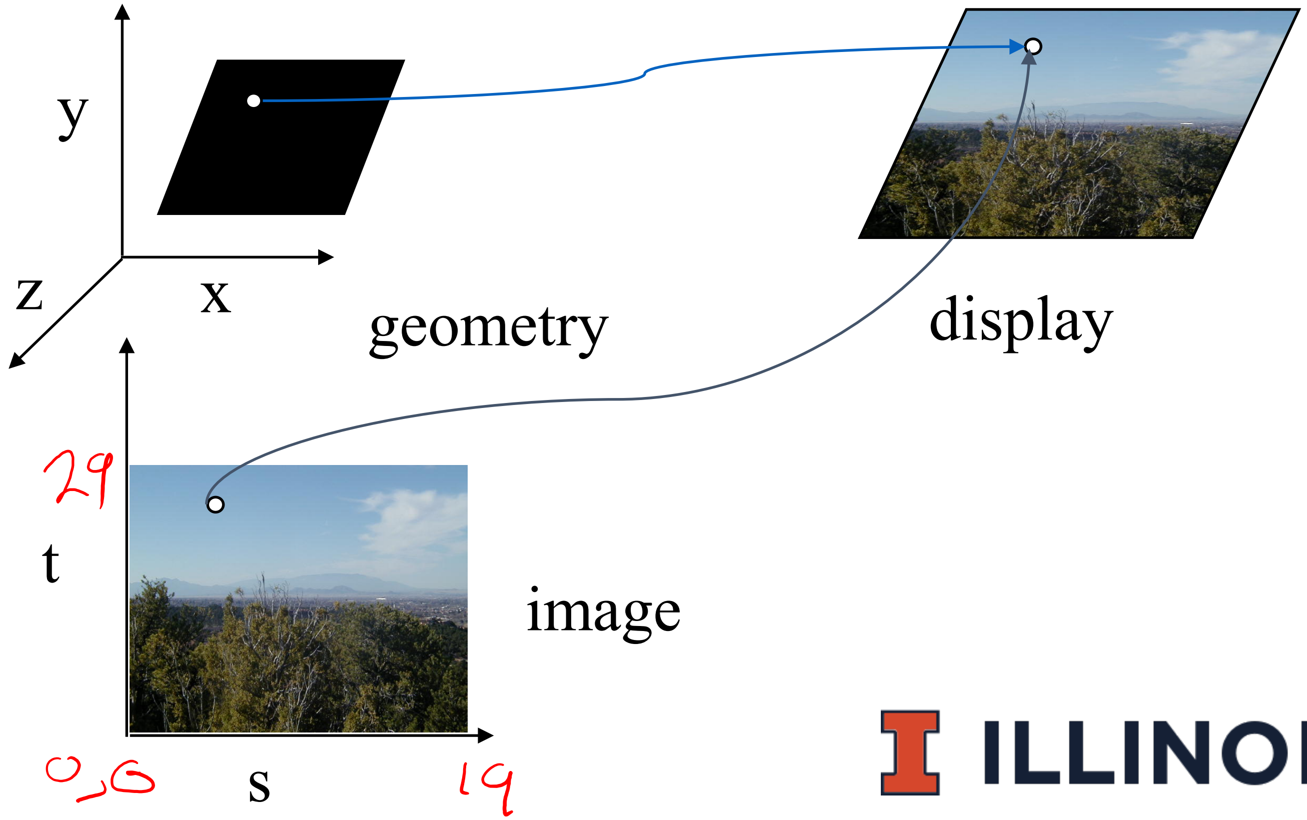
Mapping means we need a function

- Given a point on a surface (a fragment)
- We want to know to which texel in the texture it corresponds to

Need a map of the form

$$s = s(x,y,z)$$
$$t = t(x,y,z)$$

We will use the word **texel** to indicate a pixel in the texture image…so it doesn't get confused with a pixel in our final rendered image



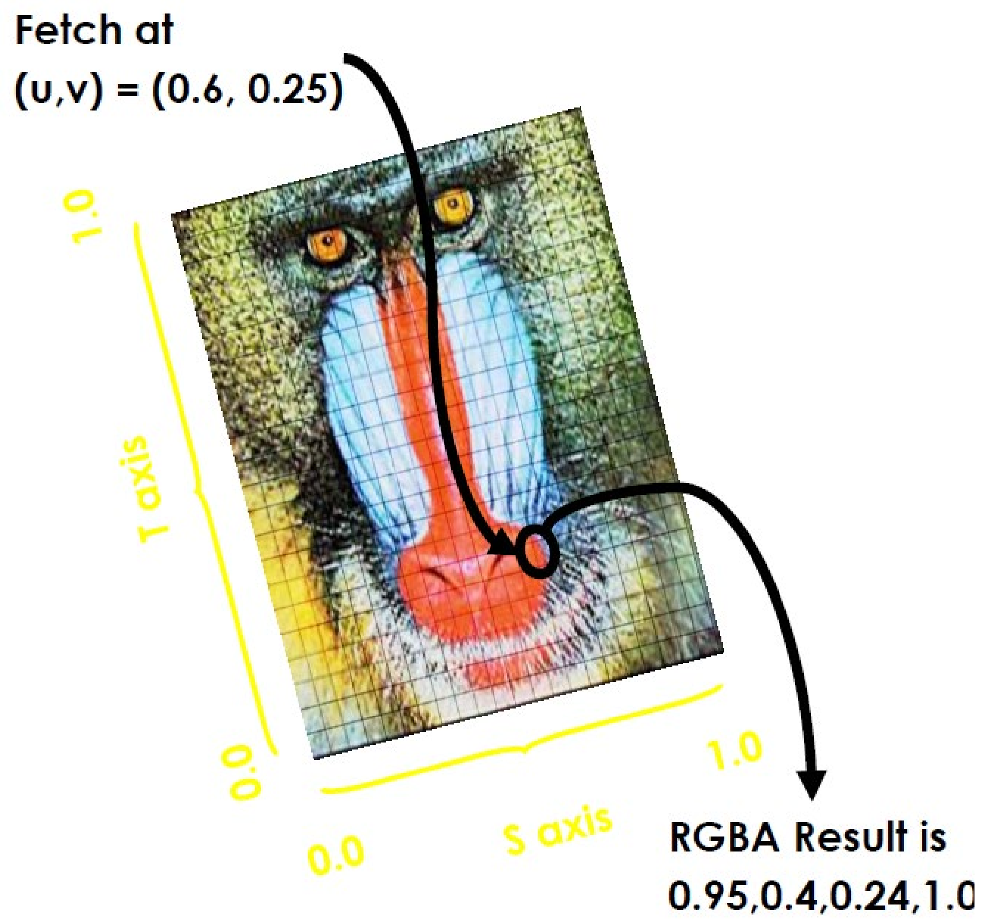
Specifying a Texture Mapping Function

- You need to come up with a mapping from

  surface vertices → texture space

- Specify mapping using (u,v) vertex attributes
- The (u,v) coordinates map into a parametric texture space

- Generally (u,v) is in [0,1] x [0, 1]
  - We can allow coordinates to go outside that range...
  - if we tell WebGL how to handle that event
    ...more on that later



ILLINOIS

A Word About Coordinates

We will use the following conventions

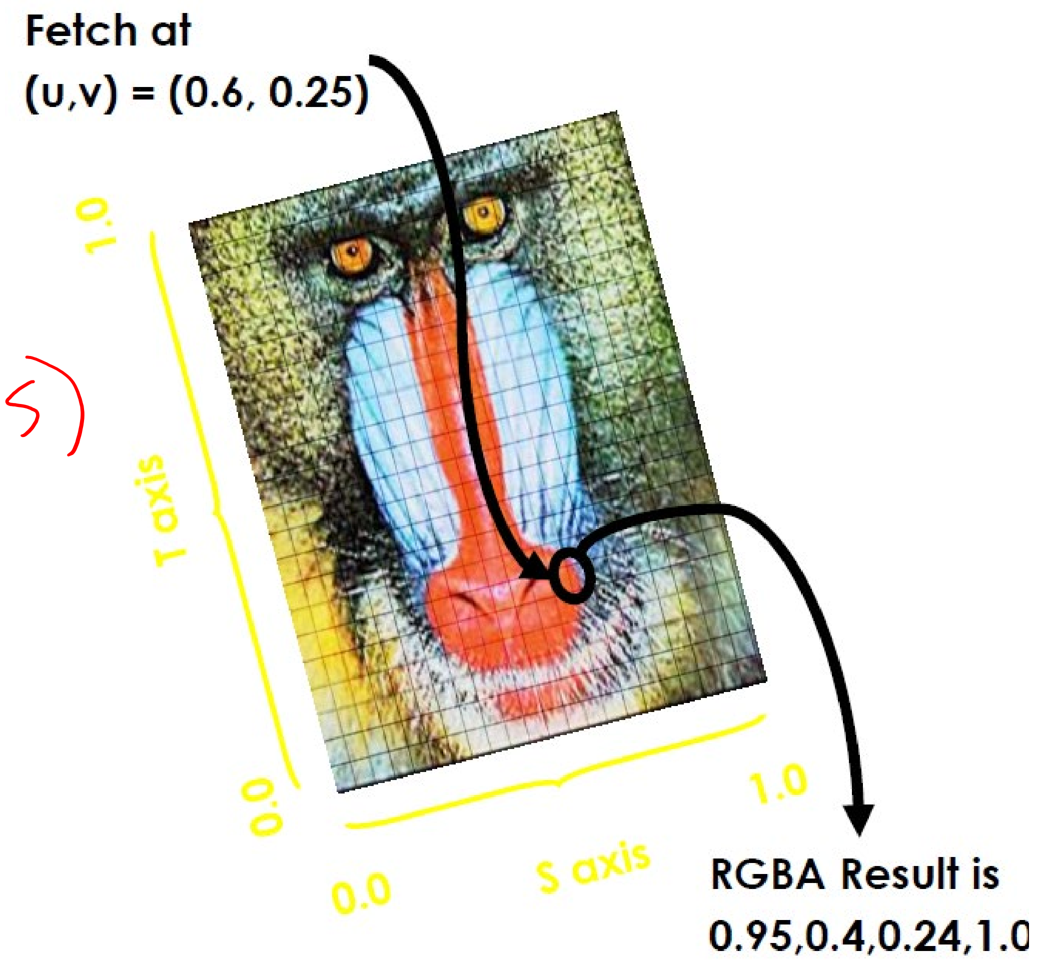
- (u,v) are the texture coordinates in the parametric space
  - Typically in [0,1]x[0,1]

(0.5, 0.5)

$\rightarrow$ (15, 15)

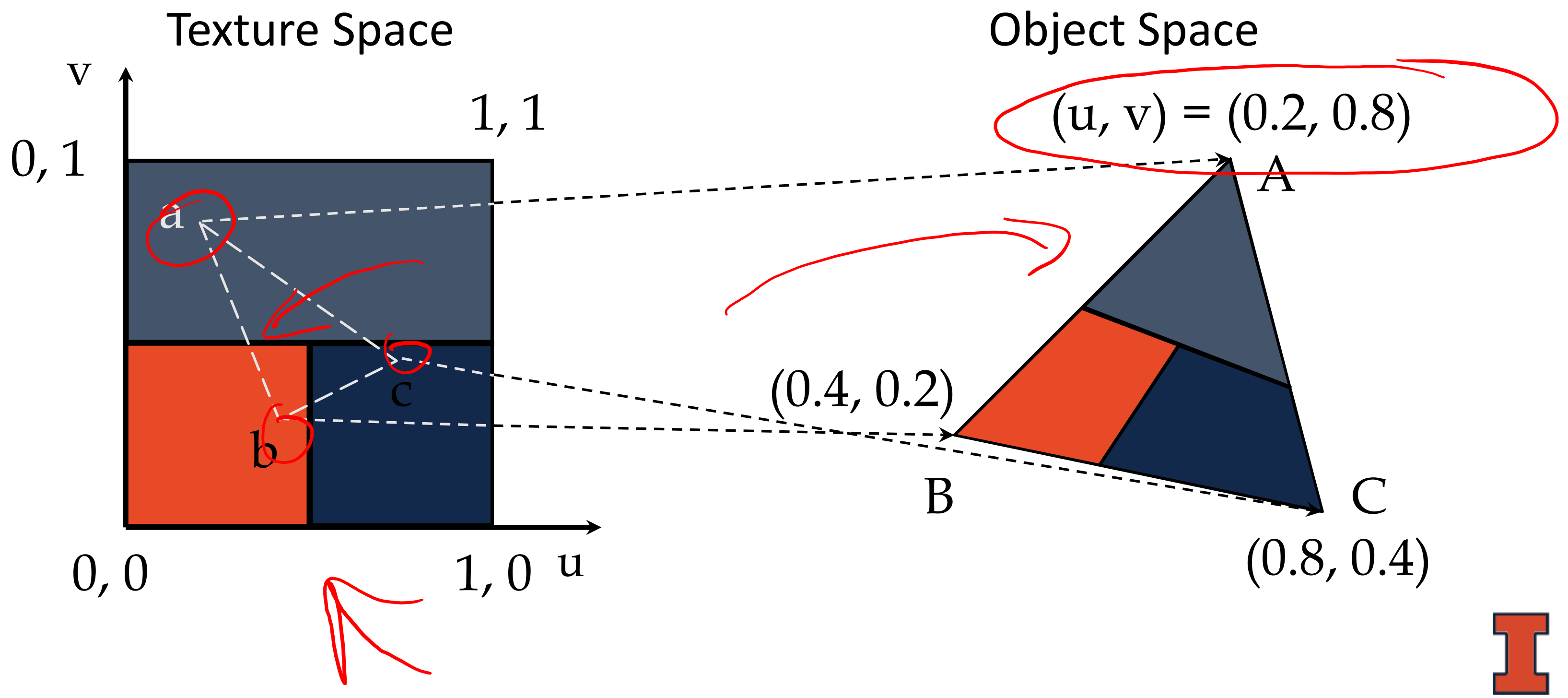
- (s,t) are the texel coordinates
  - For example in a 32x32 image (s,t) would be in [0,31]x[0,31]
  - (s,t) are floating point coordinates…
  - …we may not map directly onto an integer texel location
  - …and then we need to figure out how to choose the best texel

- Some websites/books/posts use (s,t) to denote parametric coordinates…so be aware….



Mapping a Texture

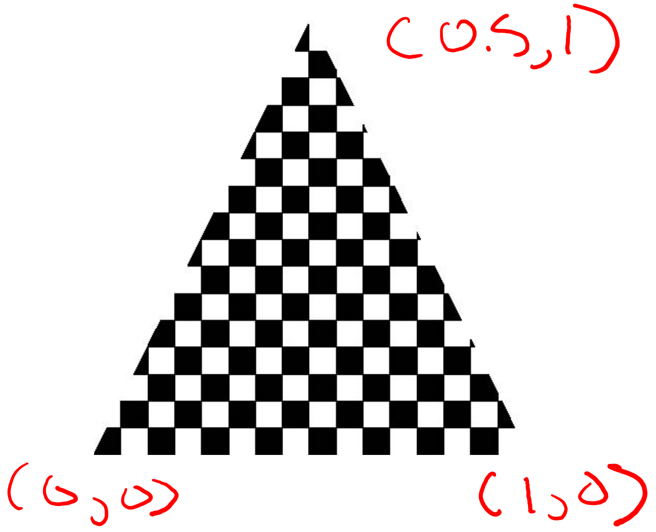
- Based on parametric texture coordinates
- Specify as a 2D vertex attribute



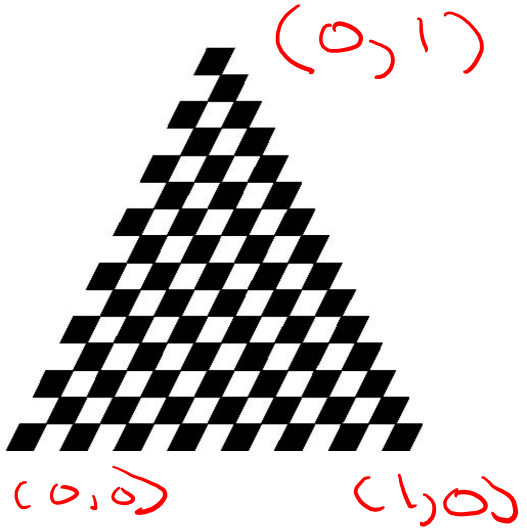
Mapping a Texture

Can be distortions

good selection
of tex coordinates



poor selection
of tex coordinates



texture stretched
over trapezoid
showing effects of
bilinear interpolation