

Texture Mapping & Other Fun Stuff

The Problem:

- Don't want to represent all this detail with geometry

Procedural Solid (3D) Textures

- Write a function: $f(x,y,z) \rightarrow \text{color}$
- non-intuitive
- difficult to match existing texture

Image removed due to copyright considerations.

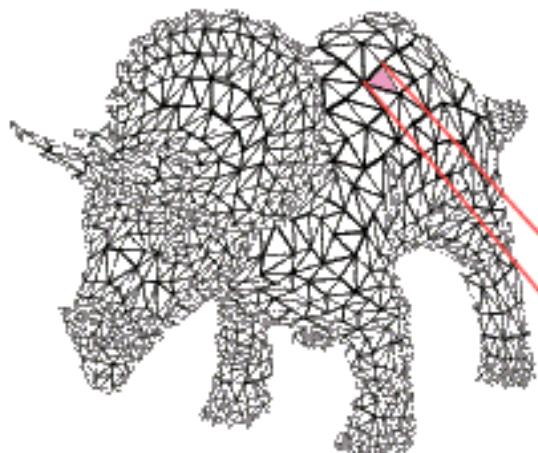
Ken Perlin, SIGGRAPH '85.

MIT EECS 6.837, Durand and Cutler

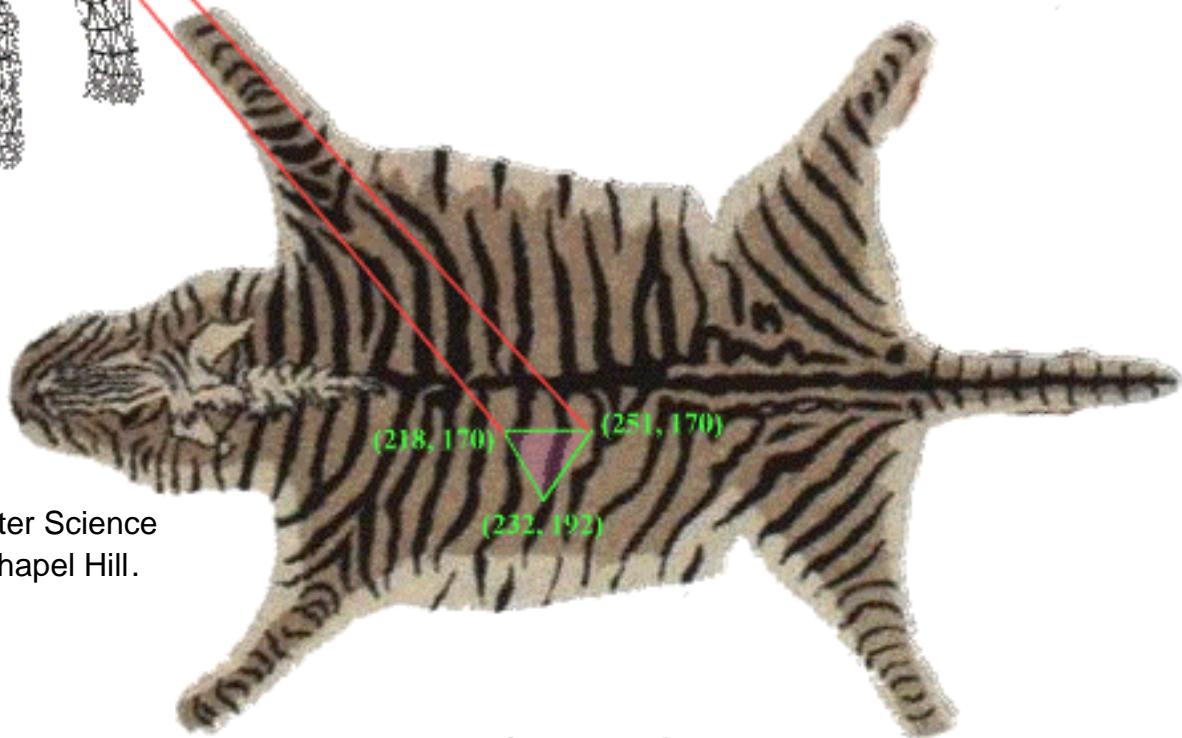
Today

- 2D Texture Mapping
 - Perspective Correct Interpolation
 - Illumination
 - Texture Mapping Difficulties
 - Projective Texturing
- Other Mapping Techniques

Photo-textures



For each triangle in the model establish a corresponding region in the phototexture



Courtesy of Leonard McMillan, Computer Science
at the University of North Carolina in Chapel Hill.
Used with permission.

*During rasterization interpolate the
coordinate indices into the texture map*

Texture Mapping

- Like wallpapering or gift-wrapping with stretchy paper
- Curved surfaces require extra stretching or cutting

Texture Coordinates

- Specify a texture coordinate (u,v) at each vertex
- Canonical texture coordinates $(0,0) \rightarrow (1,1)$
- Often the texture size is a power of 2
(but it doesn't have to be)
- How can we tile
this texture?



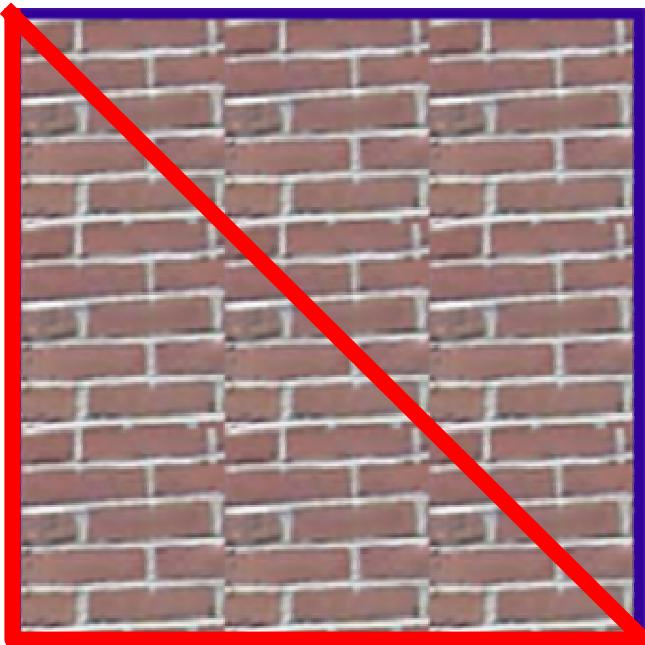
$(0,0)$

$(1,0)$

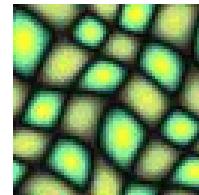
Tiling Texture



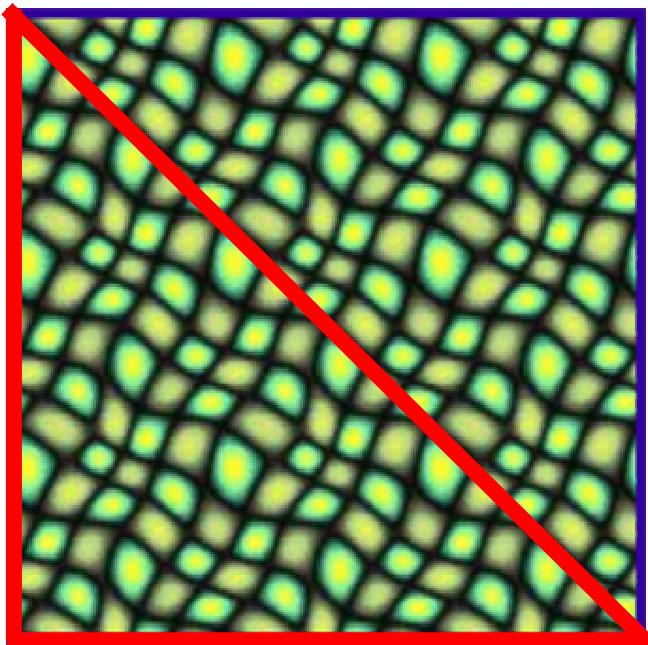
(0,3)



tiles with
visible seams



(0,3)



seamless tiling
(repeating)

Texture Coordinates

- Specify a texture coordinate (s,t) at each vertex
- Canonical texture coordinates $(0,0) \rightarrow (1,1)$
- Can we just linearly interpolate the values in screen space?

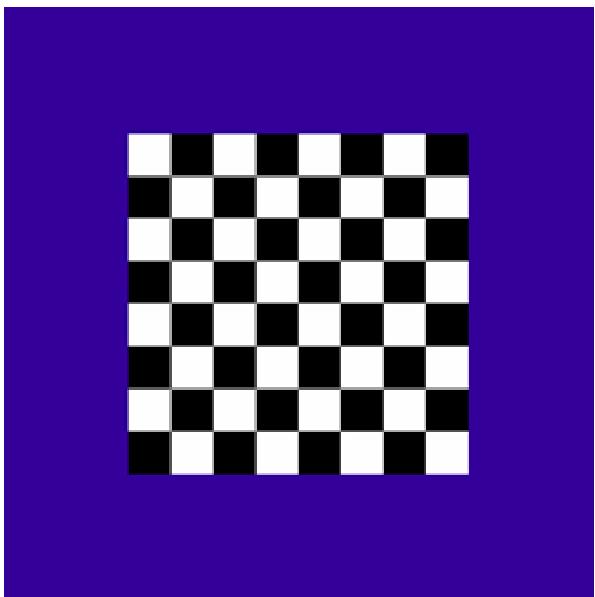
$(0,1)$



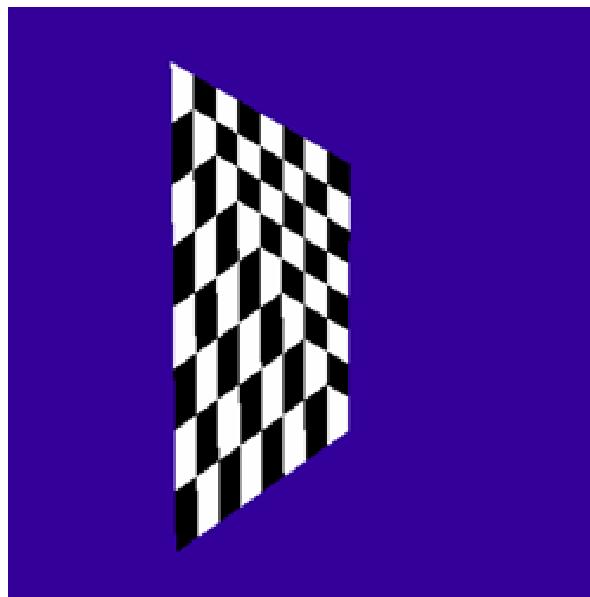
$(0,0)$

$(1,0)$

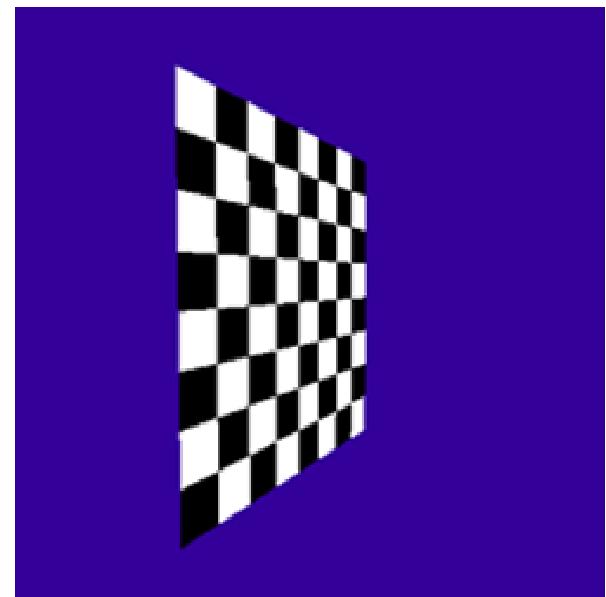
What Goes Wrong?



texture source



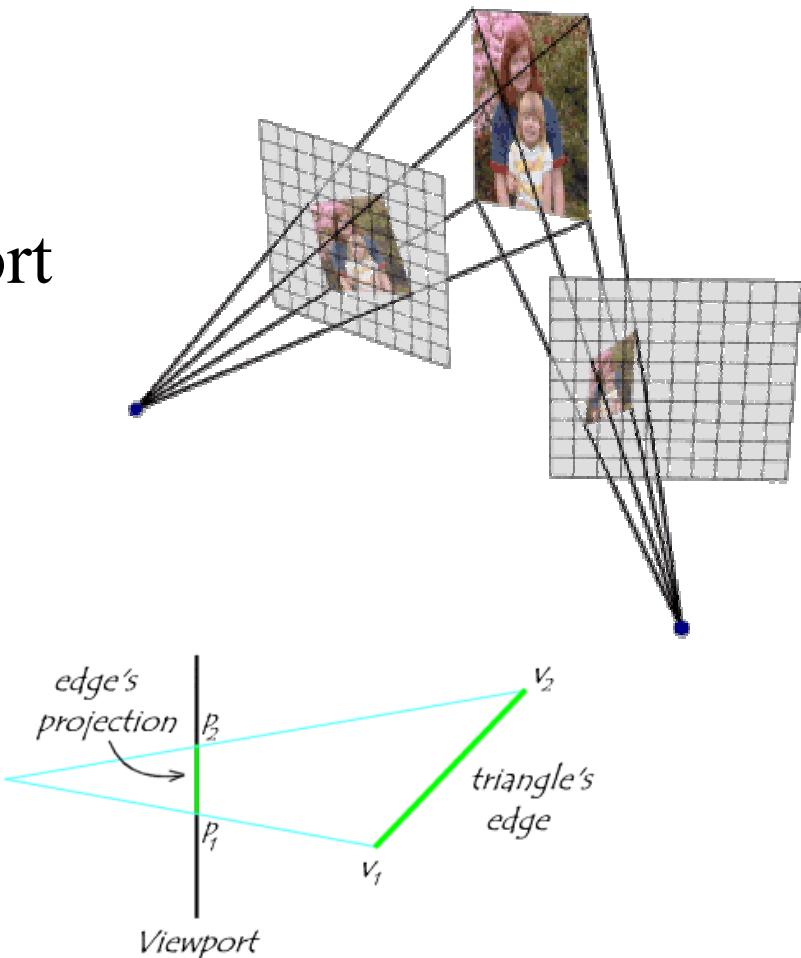
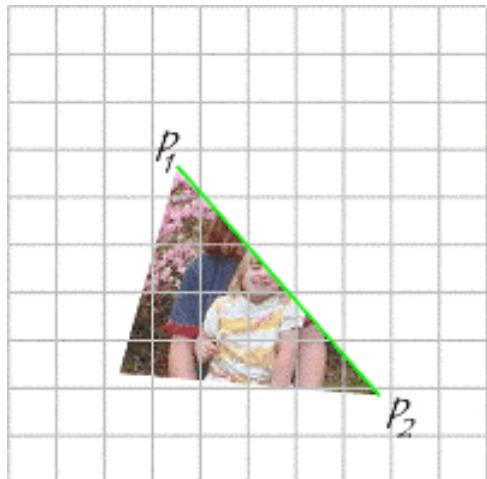
what we get



what we want

Looking at One Edge

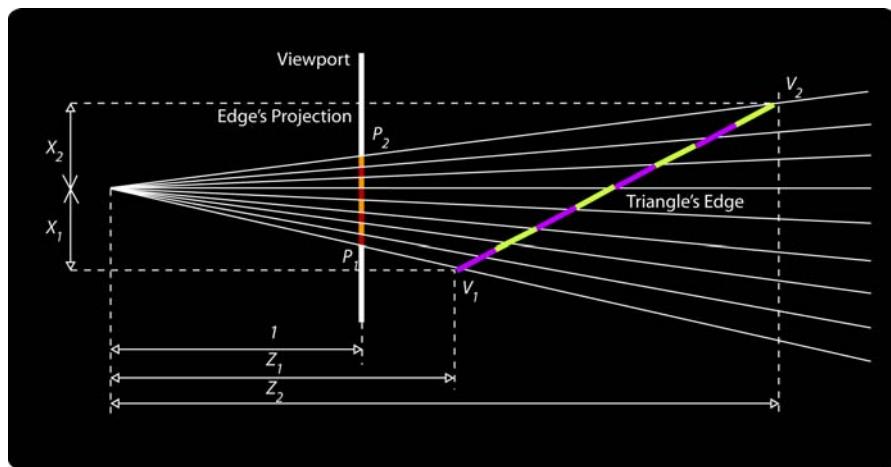
- Consider one edge from a given triangle. This edge and its projection onto our viewport lie in a single common plane illustrated below:



Courtesy of Leonard McMillan, Computer Science
at the University of North Carolina in Chapel Hill.
Used with permission.

Visualizing the Problem

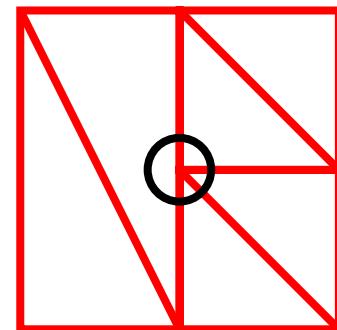
Let's assume that the viewport is located 1 unit away from the center of projection.



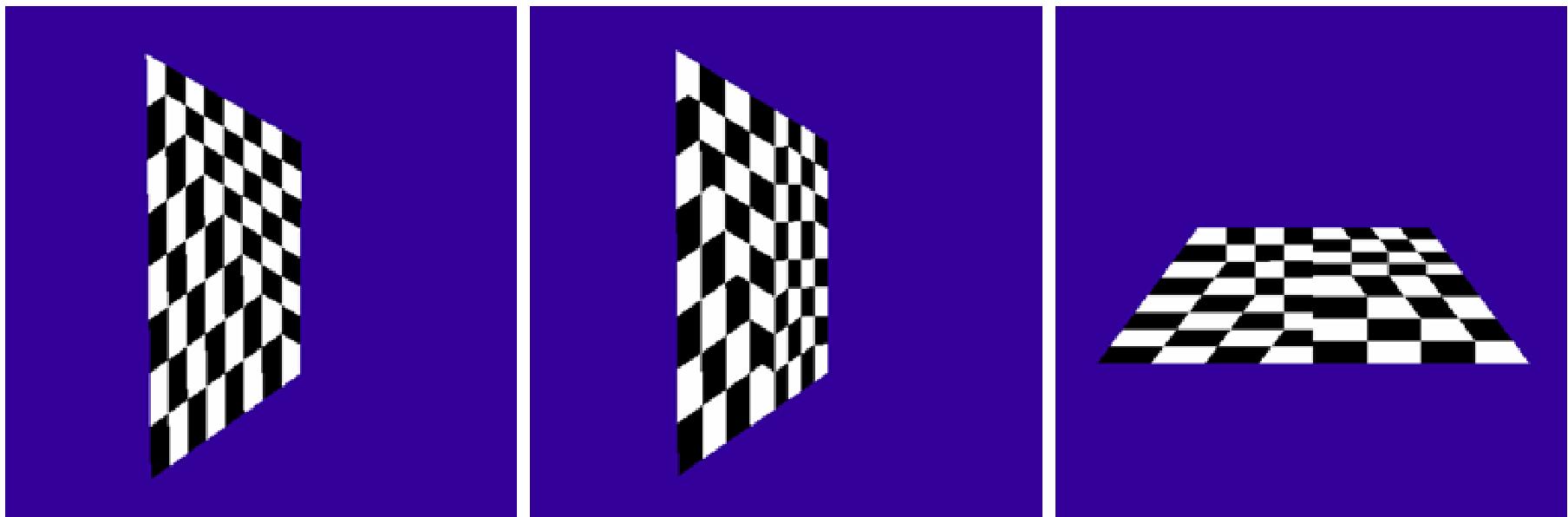
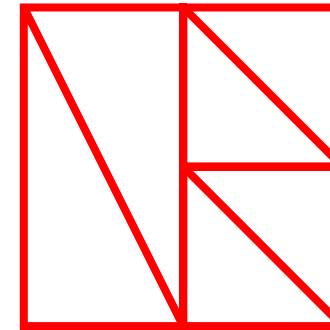
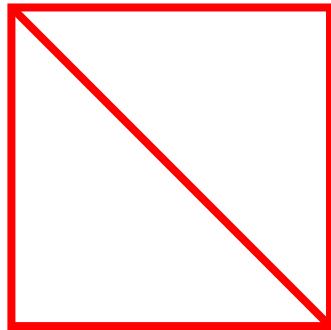
Notice that uniform steps on the image plane do not correspond to uniform steps along the edge.

How do we fix it?

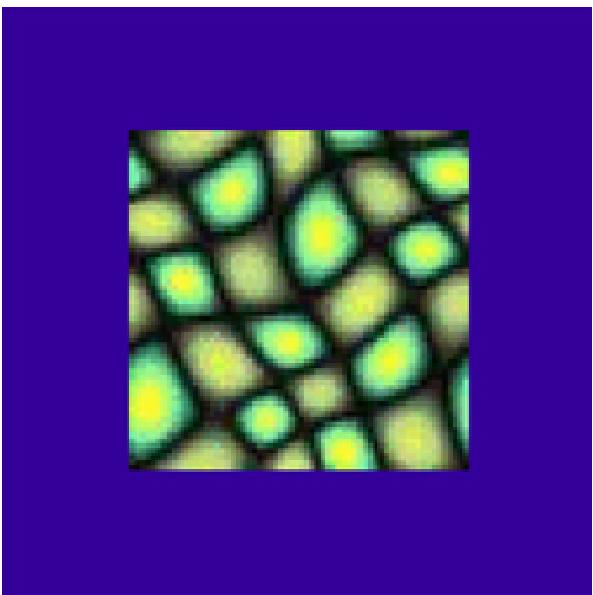
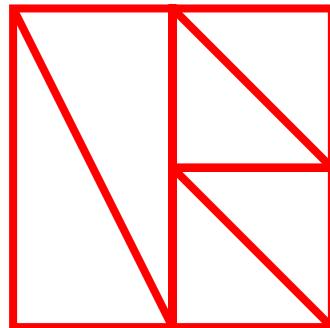
- We can reduce the perceived artifacts of non-perspective correct interpolation by subdividing the model into smaller triangles. Why?
- However, sometimes the errors become obvious
 - At "T" joints
 - When switching between levels-of-detail representations (mipmapping... next time)



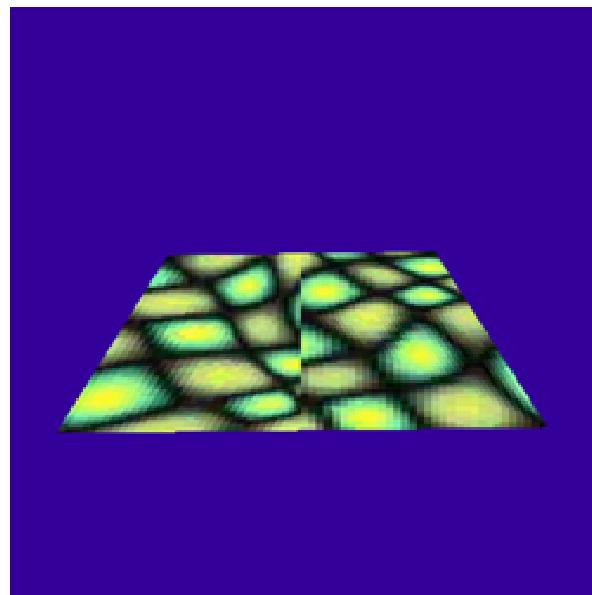
Subdivision



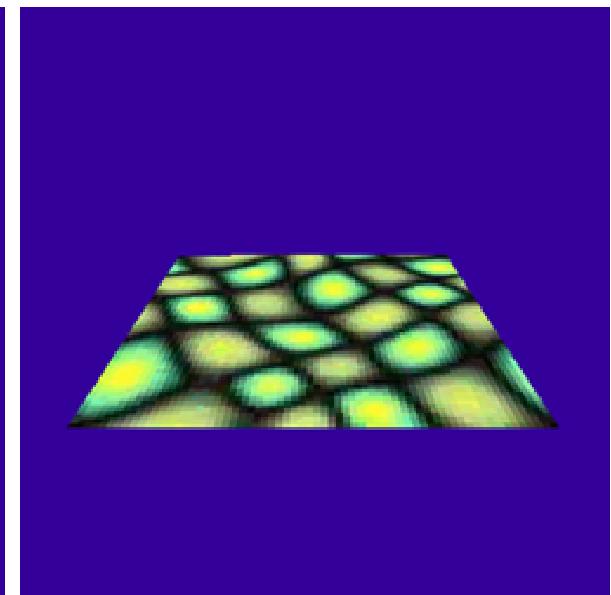
Subdivision



texture source

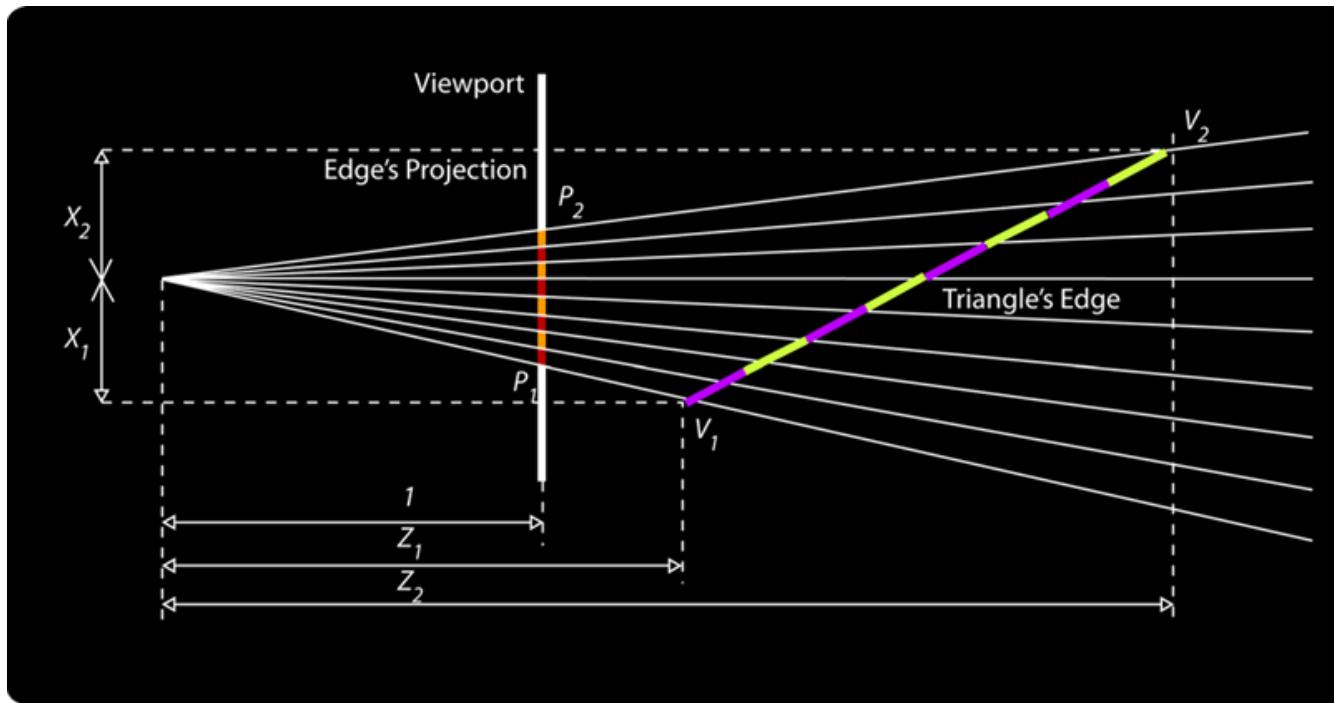


what we get



what we want

Linear Interpolation in Screen Space



Compare linear interpolation in screen space

$$p(t) = p_1 + t(p_2 - p_1) = \frac{x_1}{z_1} + t\left(\frac{x_2}{z_2} - \frac{x_1}{z_1}\right)$$

to interpolation in 3-space

$$\begin{bmatrix} x \\ z \end{bmatrix} = \begin{bmatrix} x_1 \\ z_1 \end{bmatrix} + s \left(\begin{bmatrix} x_2 \\ z_2 \end{bmatrix} - \begin{bmatrix} x_1 \\ z_1 \end{bmatrix} \right) \quad P \begin{bmatrix} x \\ z \end{bmatrix} = \frac{x_1 + s(x_2 - x_1)}{z_1 + s(z_2 - z_1)}$$

Perspective Correct Interpolation

We need a mapping from t values to s values:

$$\frac{x_1}{z_1} + t \left(\frac{x_2}{z_2} - \frac{x_1}{z_1} \right) = \frac{x_1 + s(x_2 - x_1)}{z_1 + s(z_2 - z_1)}$$

Solve for s in terms of t :

$$s = \frac{t z_1}{z_2 + t (z_1 - z_2)}$$

Unfortunately, at this point in the pipeline (after projection) we no longer have z . However, we do have $w_1 = 1/z_1$ and $w_2 = 1/z_2$, so:

$$s = \frac{t \frac{1}{w_1}}{\frac{1}{w_2} + t \left(\frac{1}{w_1} - \frac{1}{w_2} \right)} = \frac{t w_2}{w_1 + t (w_2 - w_1)}$$

Questions?

Today

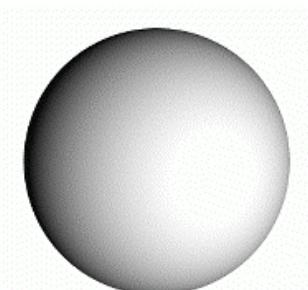
- 2D Texture Mapping
 - Perspective Correct Interpolation
 - Illumination
 - Texture Mapping Difficulties
 - Projective Texturing
- Other Mapping Techniques

Texture Mapping & Illumination

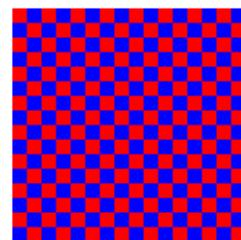
- Texture mapping can be used to alter some or all of the constants in the illumination equation: as the final color for the pixel, or as the diffuse color, or to alter the normal, ... the possibilities are endless! (e.g. GL_DECAL, GL_MODULATE, GL_BLEND, ...)

$$I_{total} = k_a I_{ambient} + \sum_{i=1}^{lights} I_i \left(k_d (\hat{N} \cdot \hat{L}) + k_s (\hat{V} \cdot \hat{R})^{n_{shiny}} \right)$$

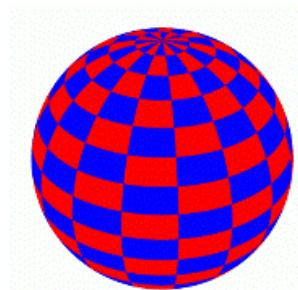
Phong's Illumination Model



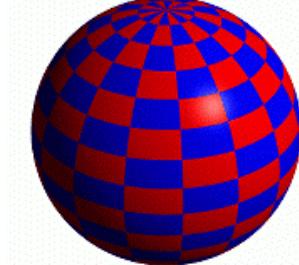
Constant Diffuse Color



Diffuse Texture Color



Texture used as Label

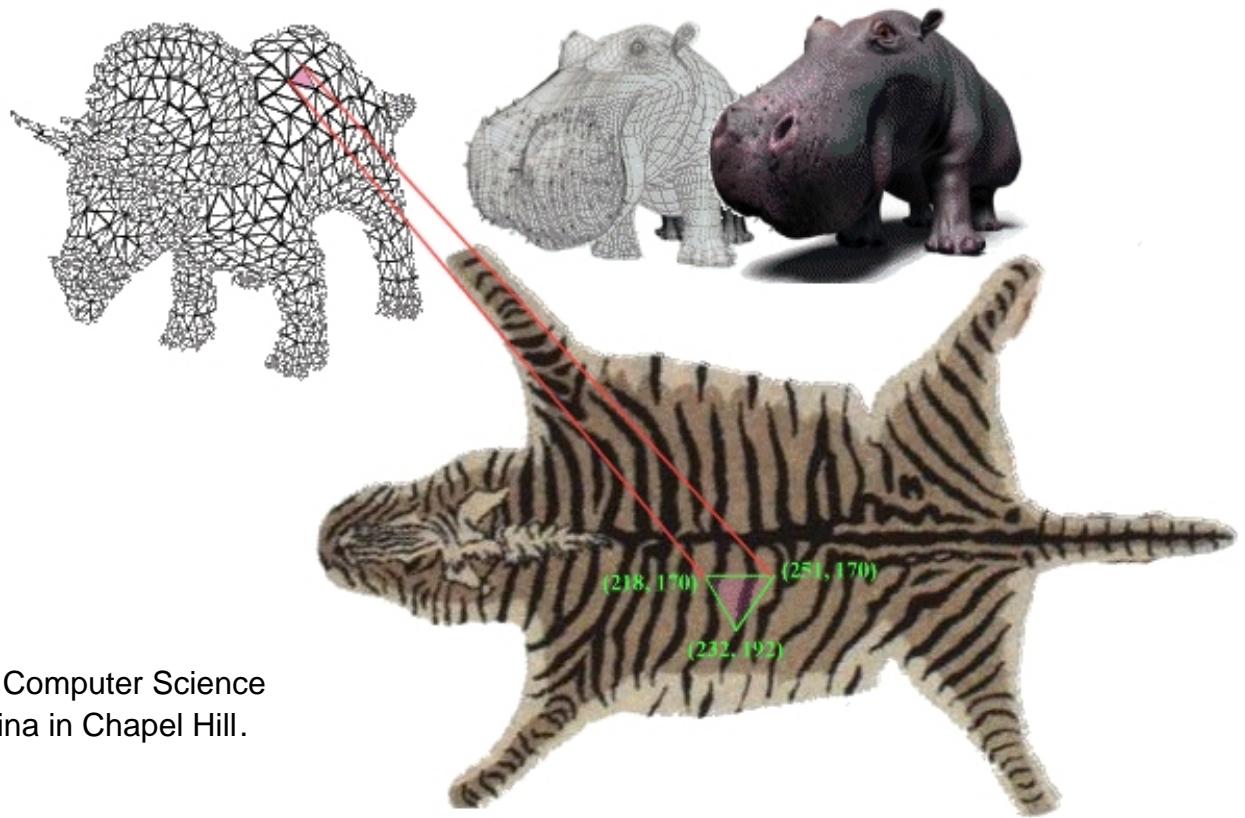


Texture used as Diffuse Color

Courtesy of Leonard McMillan, Computer Science at the University of North Carolina in Chapel Hill. Used with permission.

2D Texture Mapping

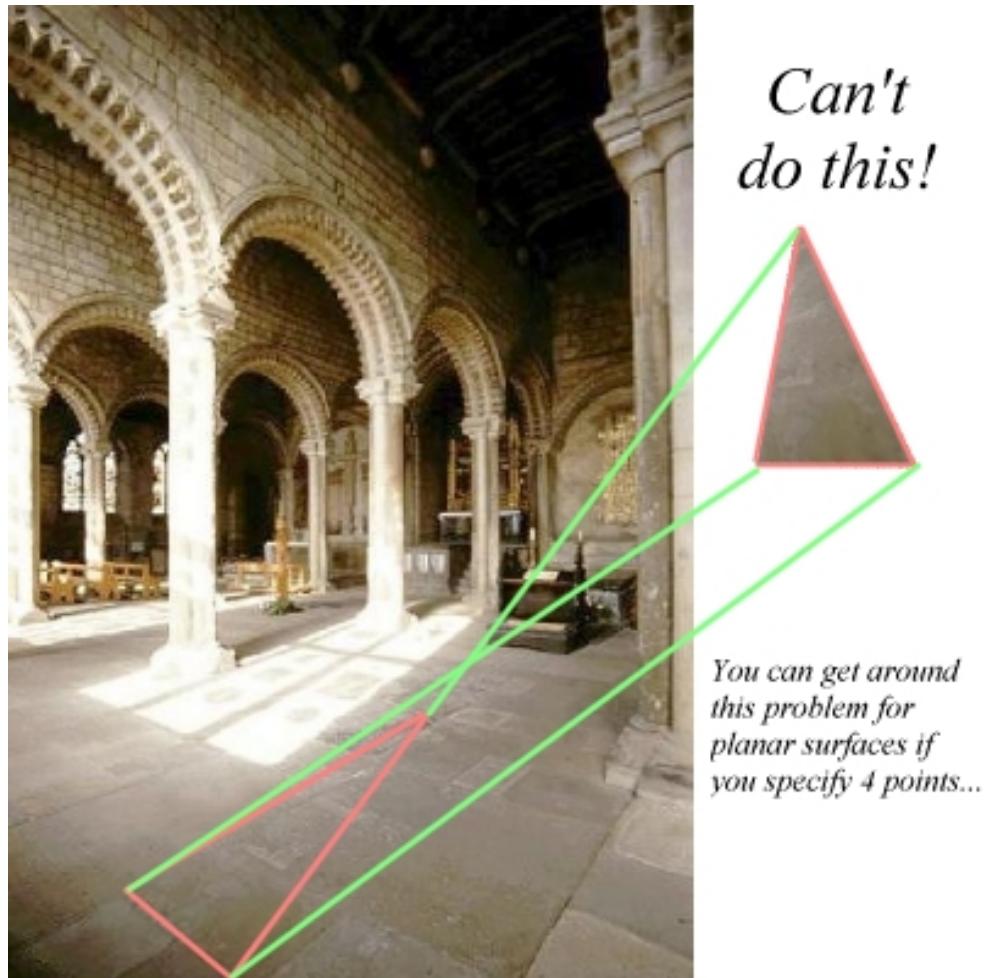
- Increases the apparent complexity of simple geometry
- Requires perspective projection correction
- Specifies variations in shading within a primitive:
 - Illumination
 - Surface Reflectance



Courtesy of Leonard McMillan, Computer Science
at the University of North Carolina in Chapel Hill.
Used with permission.

Texture Mapping Difficulties

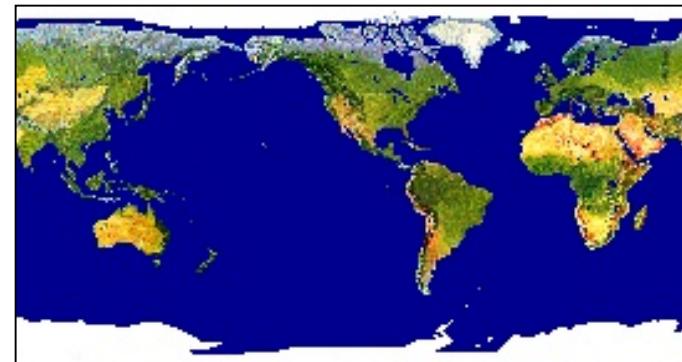
- Tedious to specify texture coordinates for every triangle
- Easier to model variations in reflectance than illumination
- Acquiring textures is surprisingly difficult
 - Texture image can't have projective distortions
 - Seamless tiling
 - Non-repeating textures



Courtesy of Leonard McMillan, Computer Science at the University of North Carolina in Chapel Hill. Used with permission.

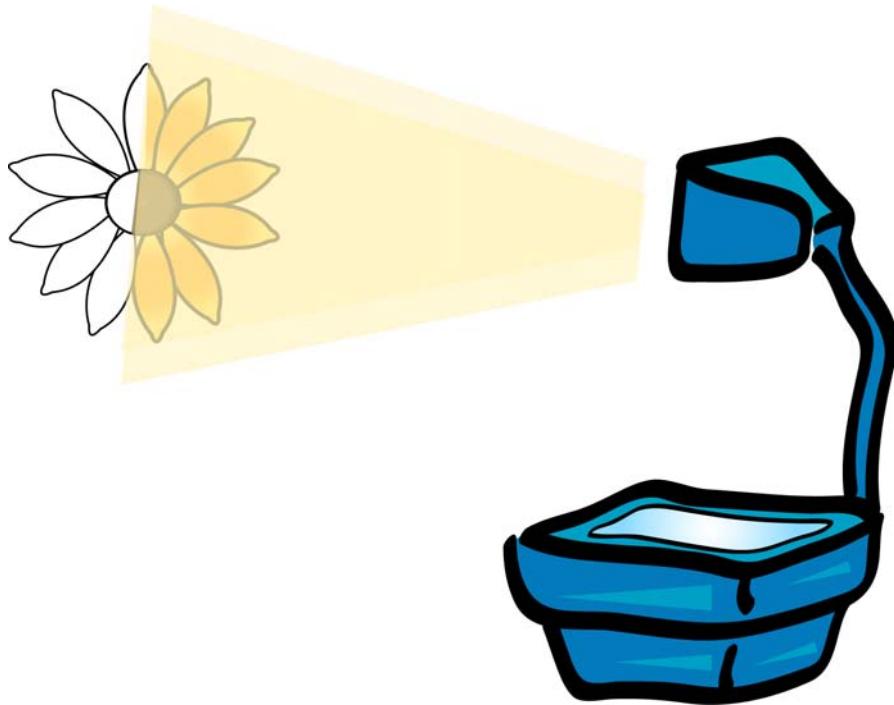
Common Texture Coordinate Mappings

- Orthogonal
- Cylindrical
- Spherical
- Perspective Projection
- Texture Chart



Projective Textures

- Treat the texture as a light source (a slide projector)
- No need to specify texture coordinates explicitly
- A good model for shading variations due to illumination
- A fair model for reflectance (can use pictures)



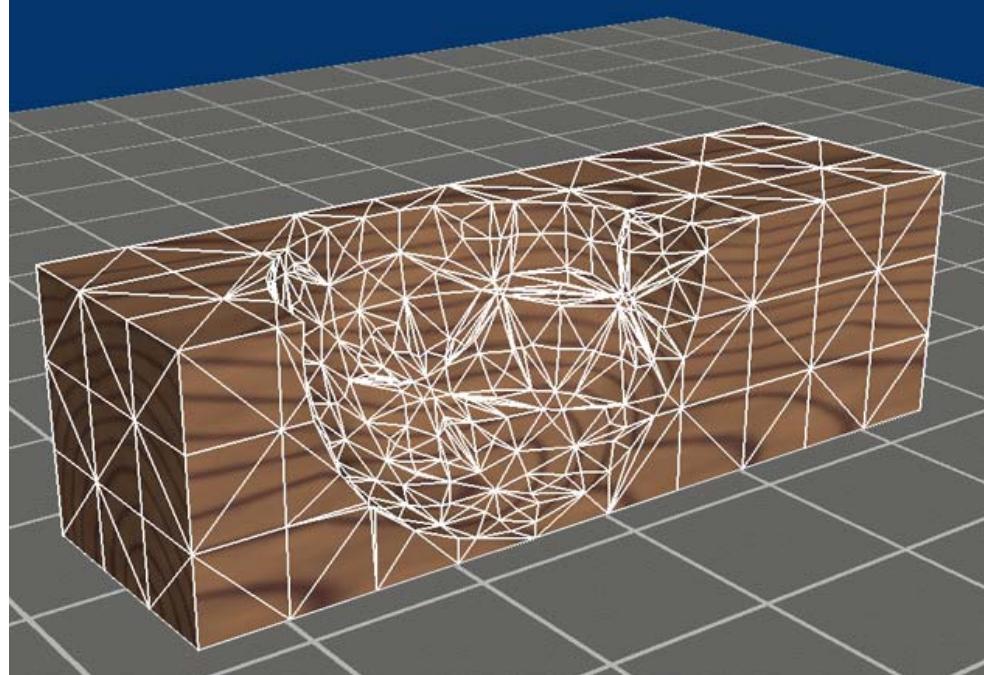
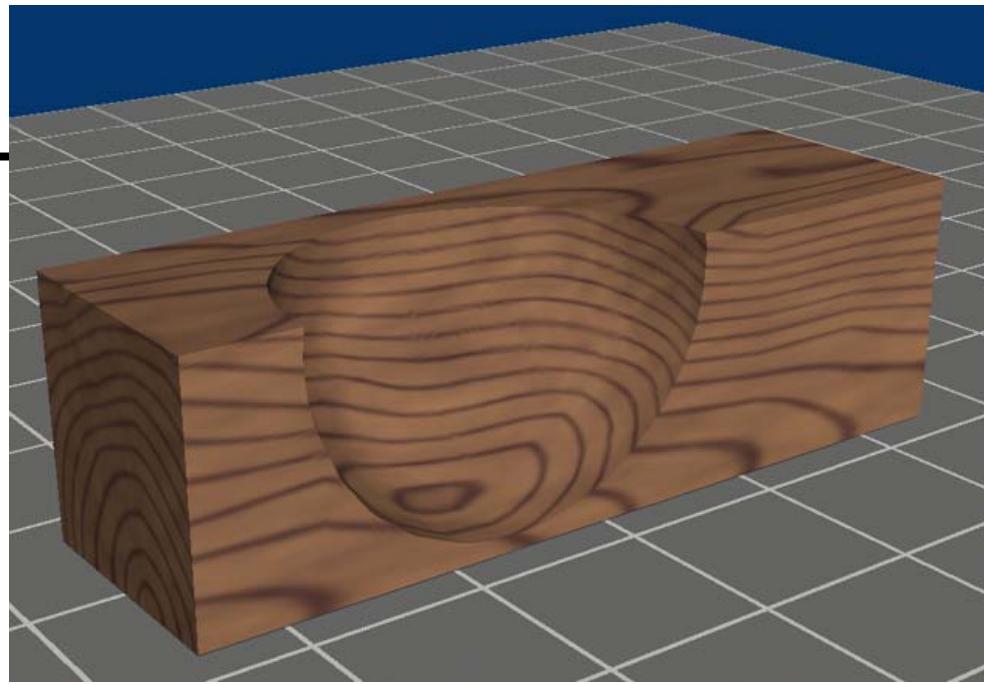
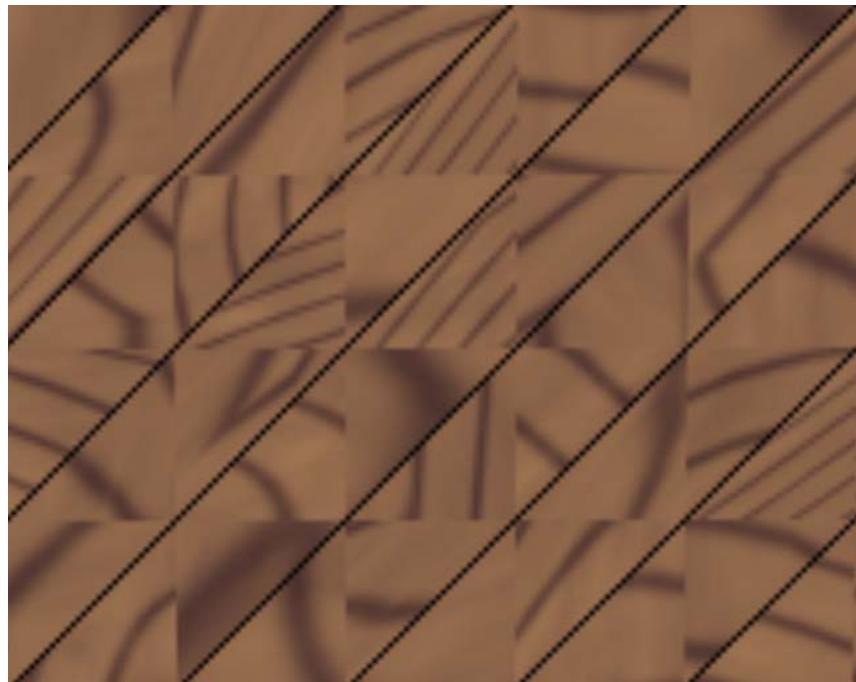
Projective Texture Example

- Modeling from photograph
- Using input photos as textures

Images removed due to copyright considerations

Texture Chart

- Pack triangles into a single image



Questions?

Today

- 2D Texture Mapping
- Other Mapping Techniques:
 - Projective Shadows and Shadow Maps
 - Bump Mapping
 - Displacement Mapping
 - Environment Mapping (for Reflections)

Projective Shadows

Images removed due to copyright considerations.

Figure from Moller & Haines “Real Time Rendering”

Shadow Maps

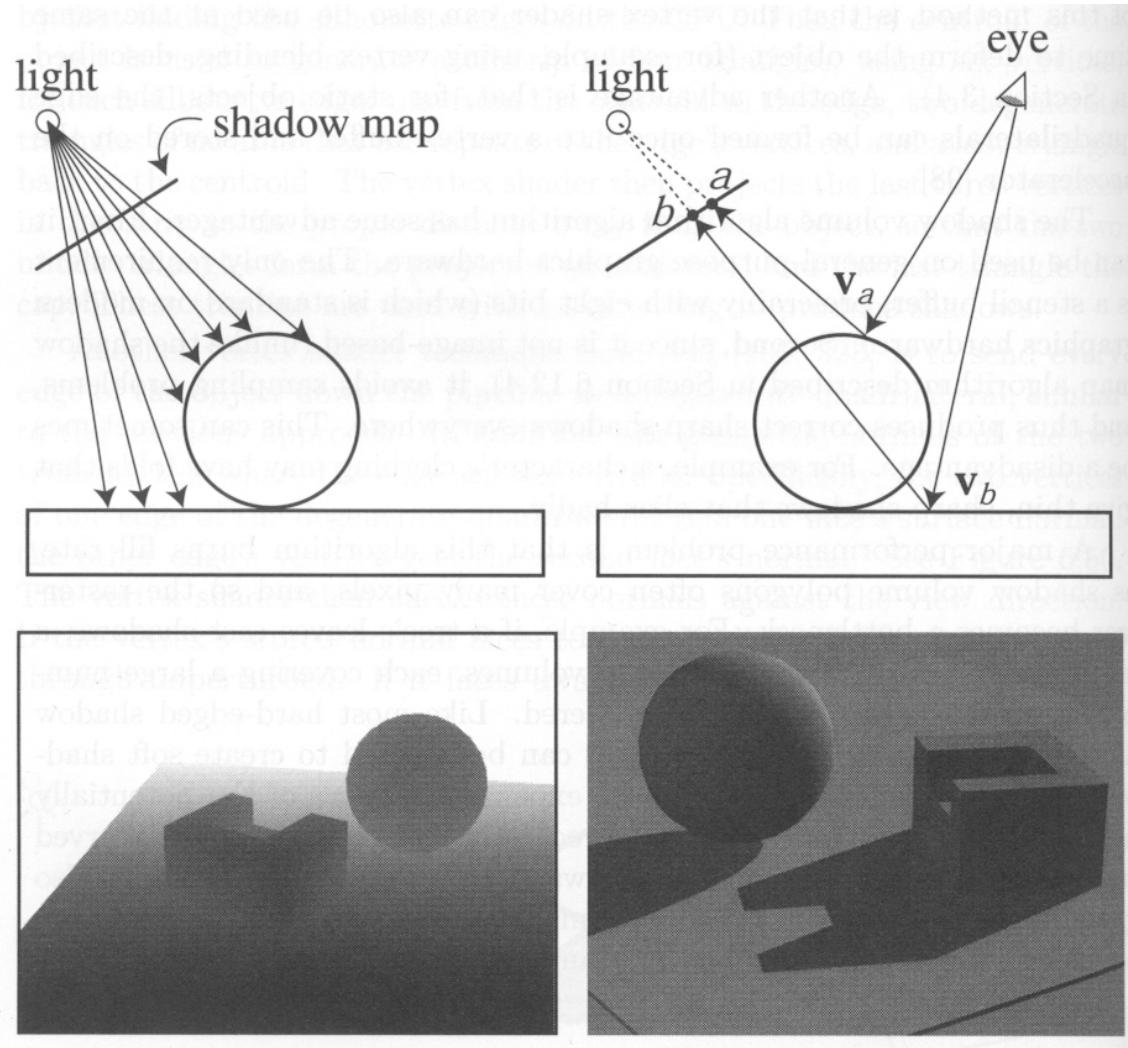


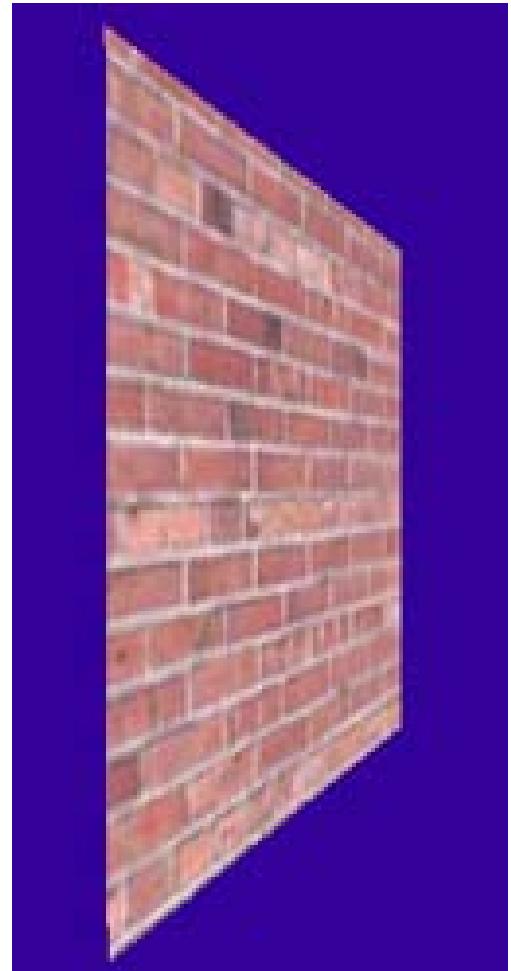
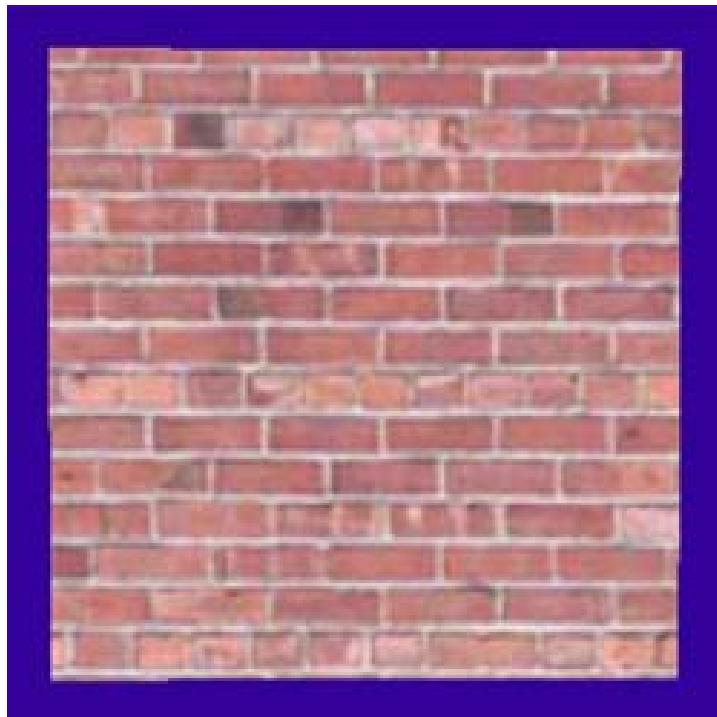
Figure from Foley et al. "Computer Graphics Principles and Practice"

Today

- 2D Texture Mapping
- Other Mapping Techniques:
 - Projective Shadows and Shadow Maps
 - **Bump Mapping**
 - Displacement Mapping
 - Environment Mapping (for Reflections)

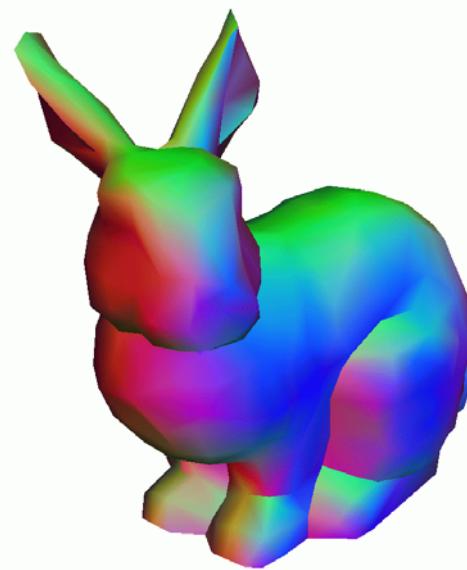
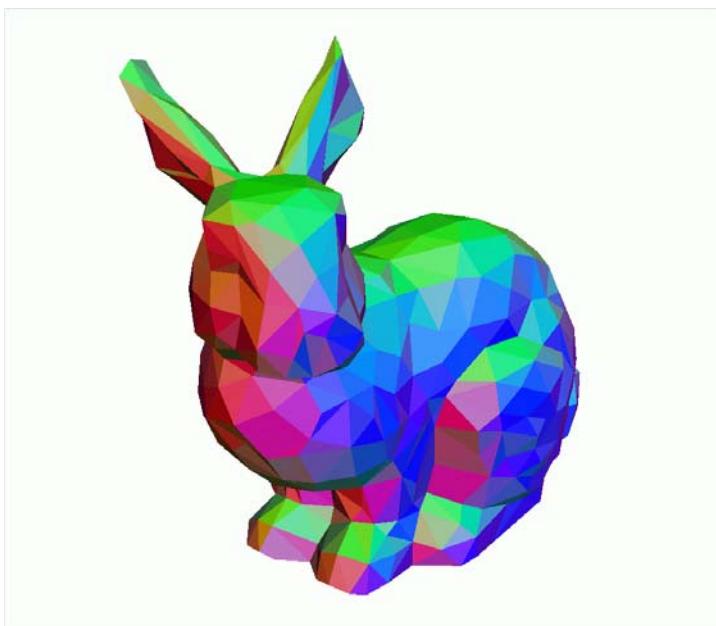
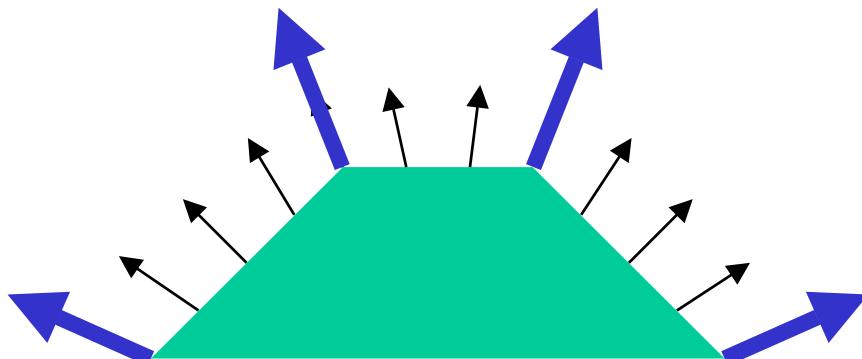
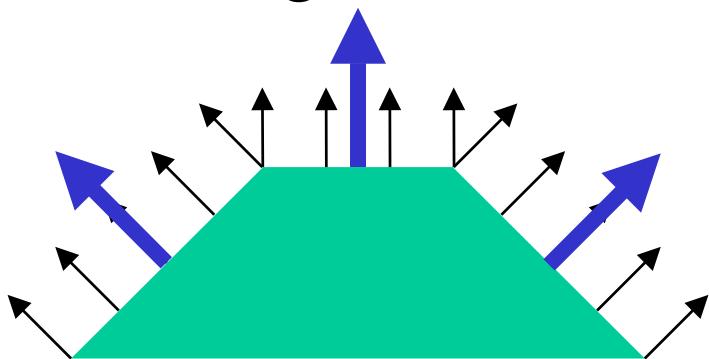
What's Missing?

- What's the difference between a real brick wall and a photograph of the wall texture-mapped onto a plane?
- What happens if we change the lighting or the camera position?



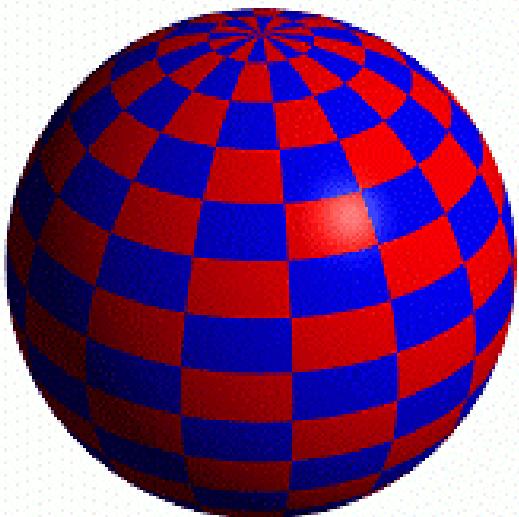
Remember Phong Normal Interpolation?

- Instead of using the normal of the triangle, interpolate an averaged normal at each vertex across the face

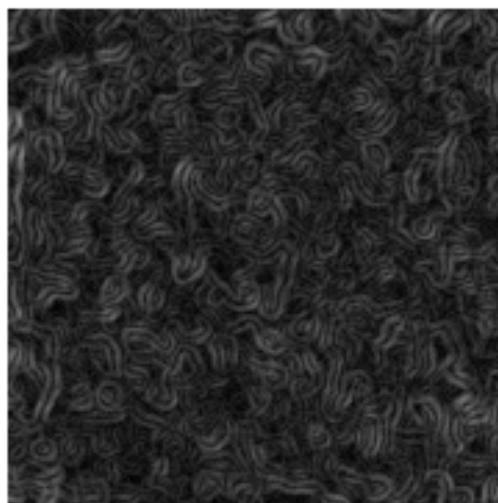


Bump Mapping

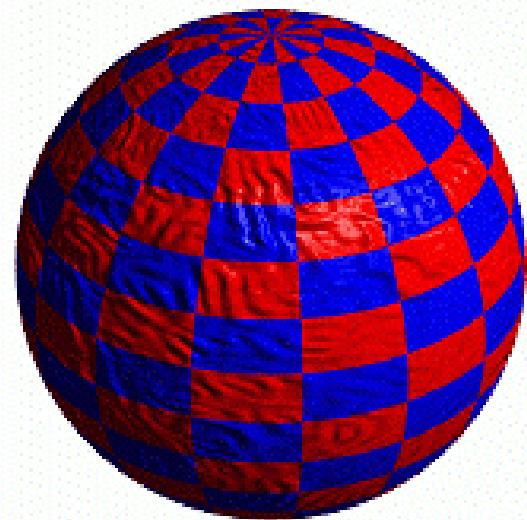
- Textures can be used to alter the surface normal of an object.
- This does not change the actual shape of the surface -- we are only shading it as if it were a different shape!



Sphere w/Diffuse Texture



Swirly Bump Map

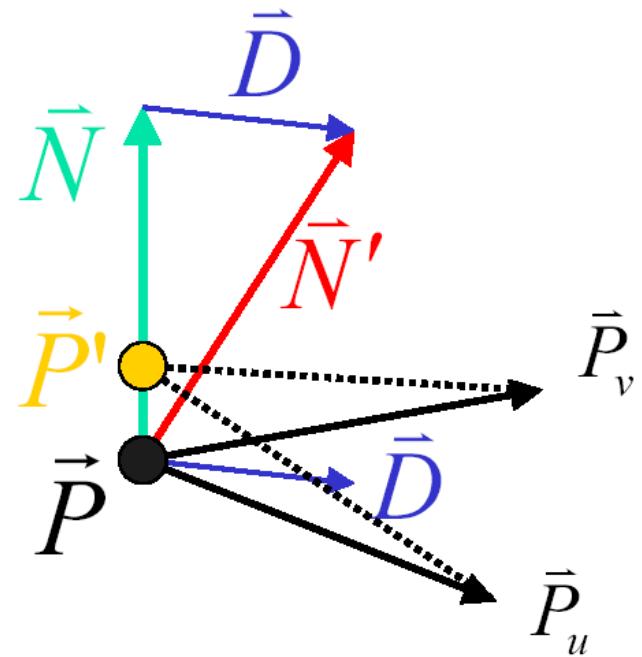
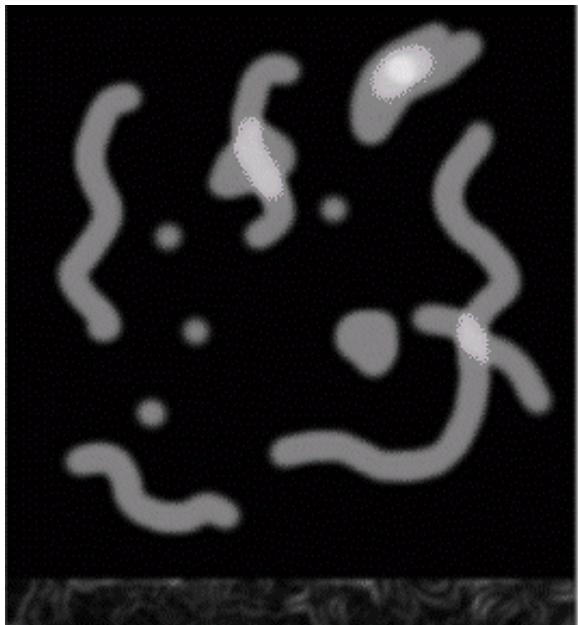


Sphere w/Diffuse Texture & Bump Map

Courtesy of Leonard McMillan. Used with permission.

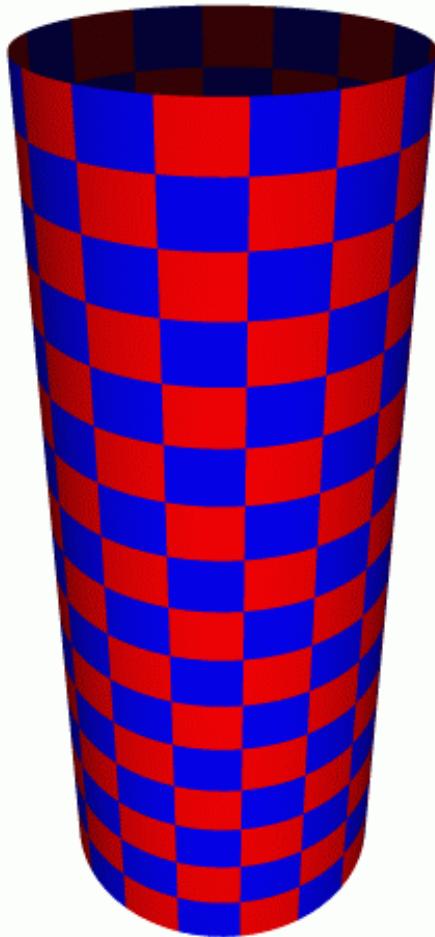
Bump Mapping

- The texture map is treated as a single-valued height function.
- The partial derivatives of the texture tell us how to alter the true surface normal at each point to make the object appear as if it were deformed by the height function.

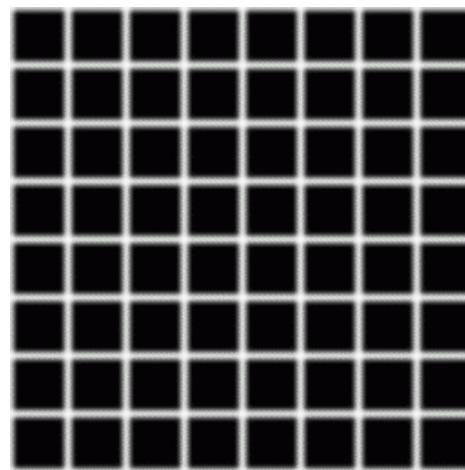


Courtesy of Leonard McMillan. Used with permission.

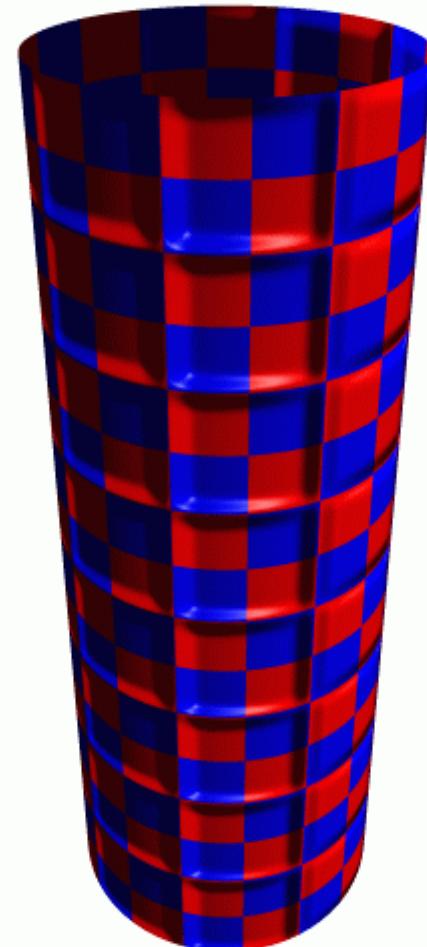
Another Bump Map Example



Cylinder w/Diffuse Texture Map

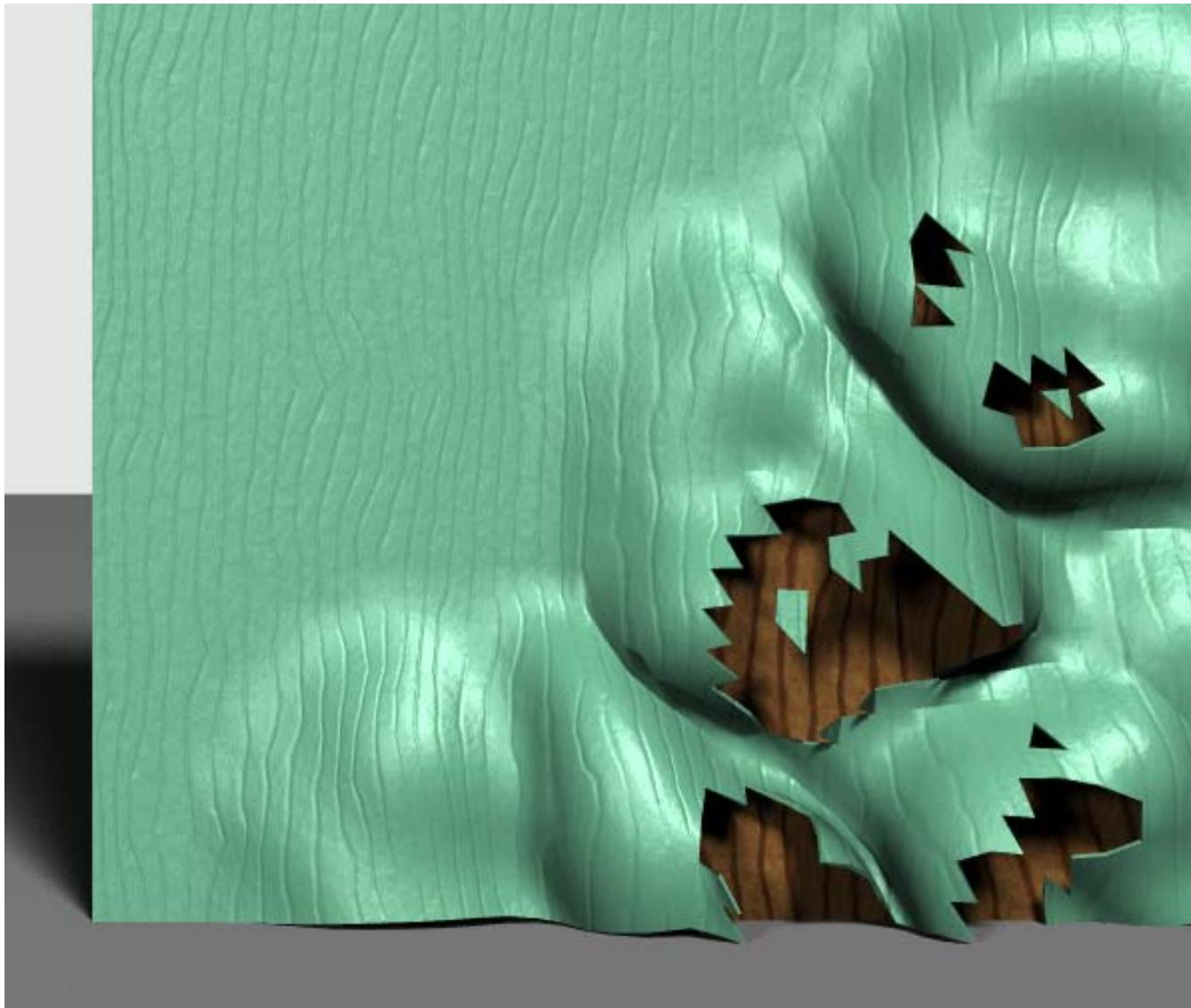


Bump Map



Cylinder w/Texture Map & Bump Map

Another Bump Map Example



MIT EECS 6.837, Durand and Cutler

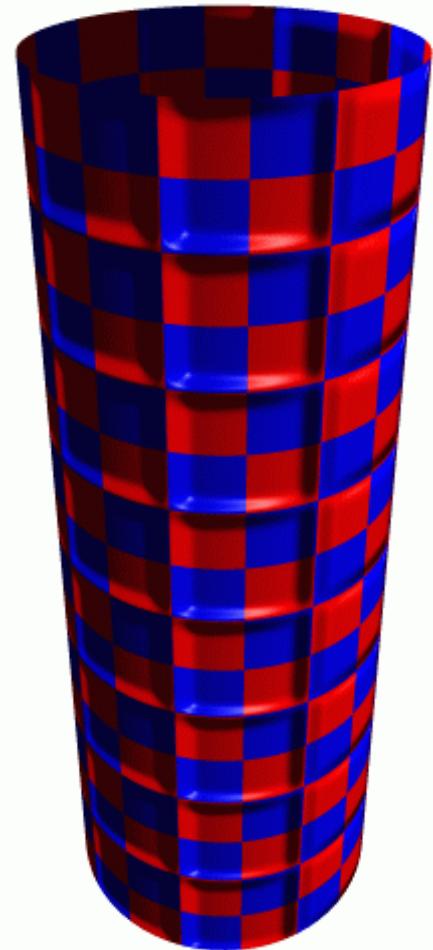
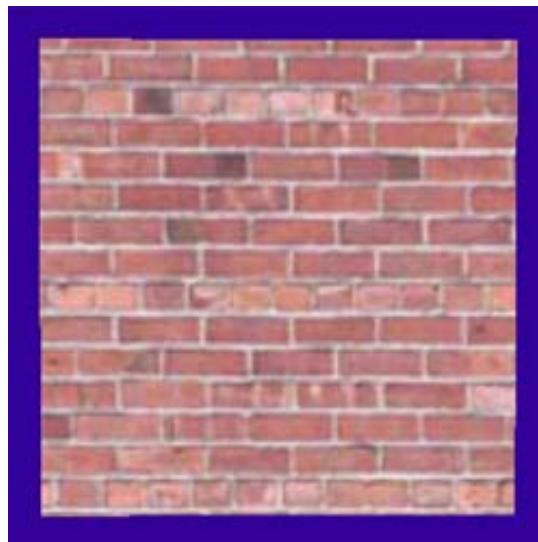
Questions?

Today

- 2D Texture Mapping
- Other Mapping Techniques:
 - Projective Shadows and Shadow Maps
 - Bump Mapping
 - **Displacement Mapping**
 - Environment Mapping (for Reflections)

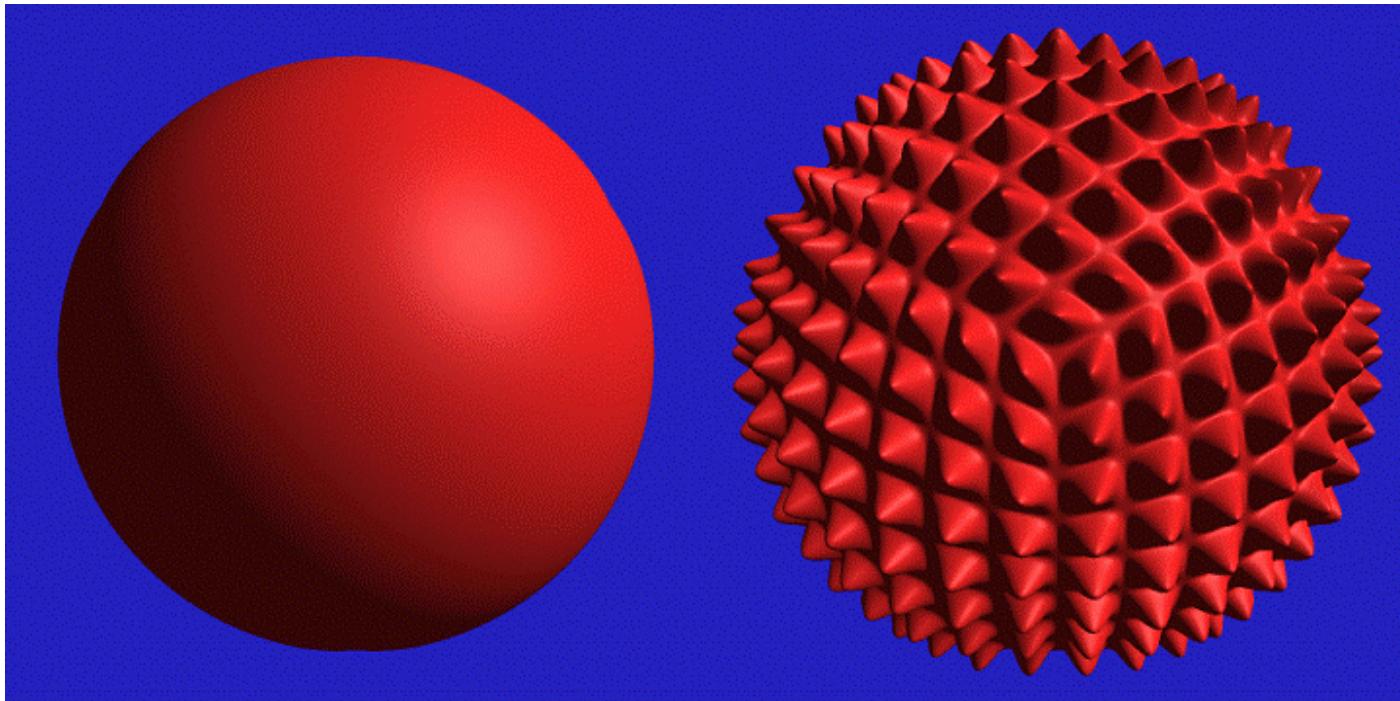
What's Missing?

- What does a texture- & bump-mapped brick wall look like as you move the viewpoint?
- What does the silhouette of a bump-mapped sphere look like?



Displacement Mapping

- Use the texture map to actually move the surface point.
How is this different than bump mapping?
- The geometry must be displaced before visibility is determined. Is this easily done in the graphics pipeline? In a ray-tracer?



Displacement Mapping



Image from:

*Geometry Caching for
Ray-Tracing Displacement Maps*

by Matt Pharr and Pat Hanrahan.

Courtesy of Matt Pharr. Used with permission.

*note the detailed shadows
cast by the stones*

Displacement Mapping



Courtesy of Ken Musgrave.
Used with permission.

Ken Musgrave

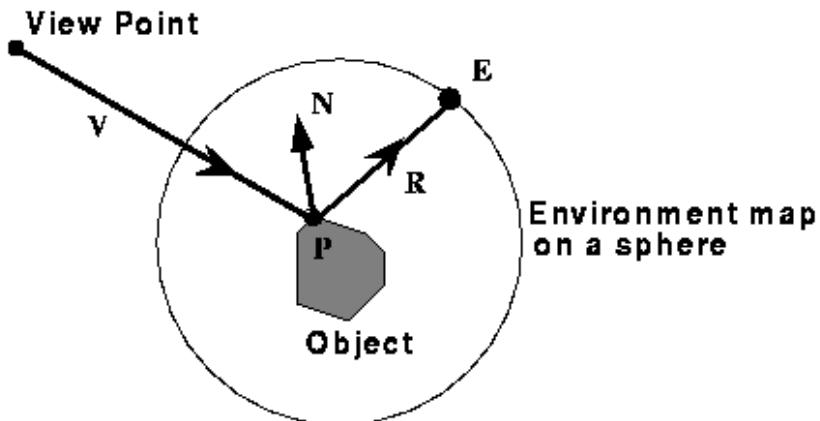
Questions?

Today

- 2D Texture Mapping
- Other Mapping Techniques:
 - Projective Shadows and Shadow Maps
 - Bump Mapping
 - Displacement Mapping
 - Environment Mapping (for Reflections)

Environment Maps

- We can simulate reflections by using the direction of the reflected ray to index a spherical texture map at "infinity".
- Assumes that all reflected rays begin from the same point.



What's the Best Chart?

Image removed due to copyright considerations.

Questions?

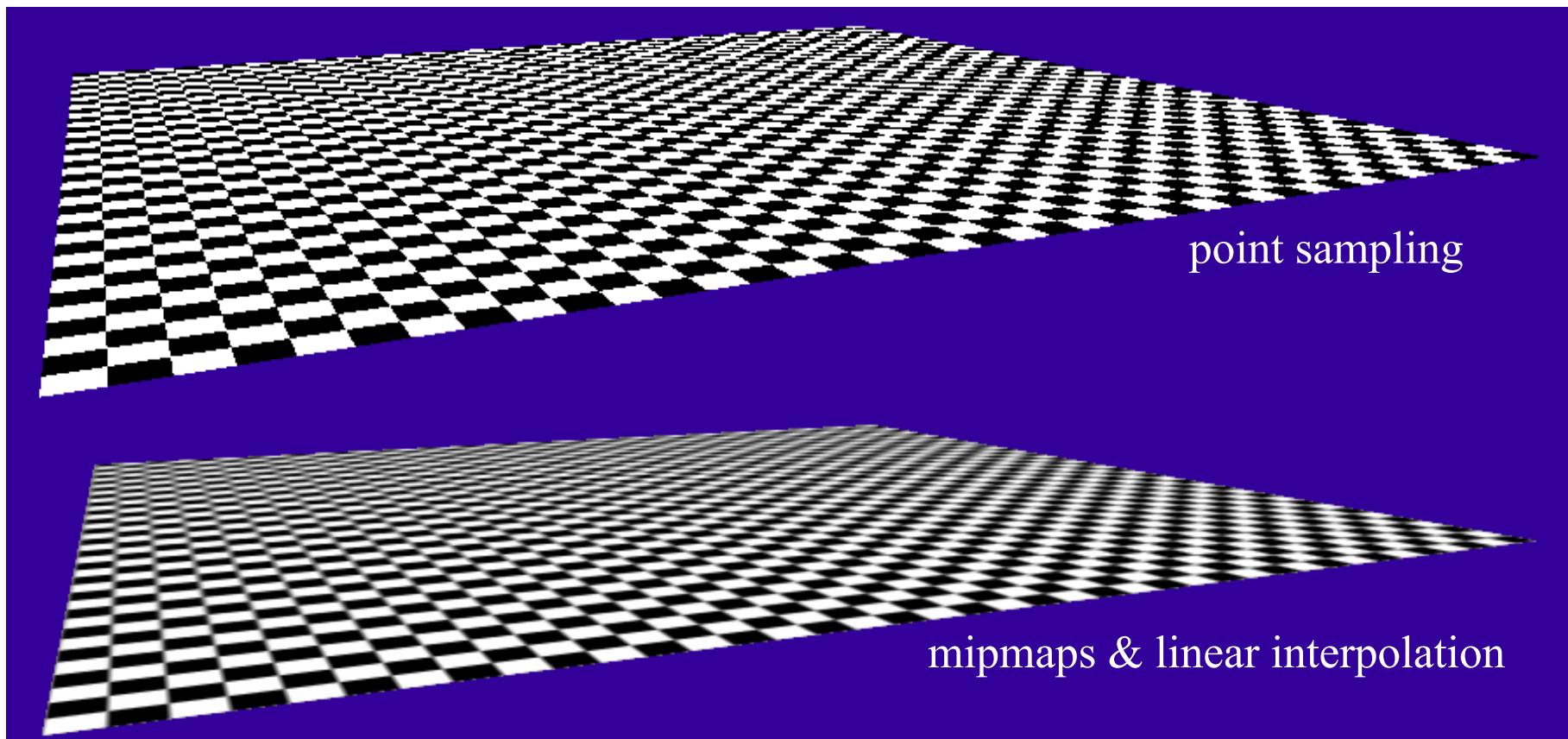


HENRIK WANN JENSEN - 2001

Image by Henrik Wann Jensen
Environment map by Paul Debevec

Textures can Alias

- *Aliasing* is the under-sampling of a signal, and it's especially noticeable during animation



Next Time:

Aliasing, Anti-Aliasing & Mipmaps for Texturing