

# Texture Mapping

Brian Curless  
CSE 557  
Fall 2009

## Reading

### Required

- ♦ Shirley, 11.1-11.2, 11.4-11.6

### Recommended

- ♦ Paul S. Heckbert. Survey of texture mapping. **IEEE Computer Graphics and Applications** 6(11): 56--67, November 1986.

### Optional

- ♦ Woo, Neider, & Davis, Chapter 9
- ♦ James F. Blinn and Martin E. Newell. Texture and reflection in computer generated images. **Communications of the ACM** 19(10): 542--547, October 1976.

## Texture mapping



Texture mapping (Woo et al., fig. 9-1)

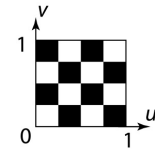
Texture mapping allows you to take a simple polygon and give it the appearance of something much more complex.

- ♦ Due to Ed Catmull, PhD thesis, 1974
- ♦ Refined by Blinn & Newell, 1976

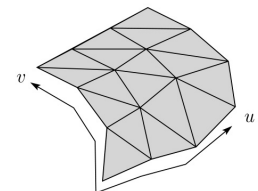
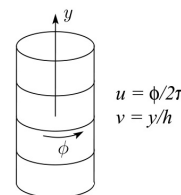
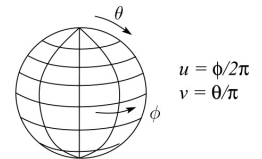
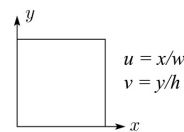
A texture can modulate just about any parameter – diffuse color, specular color, specular exponent, ...

## Implementing texture mapping

A texture lives in its own abstract image coordinates parameterized by  $(u, v)$  in the range  $([0..1], [0..1])$ :



It can be wrapped around many different surfaces:



Computing  $(u, v)$  texture coordinates in a ray tracer is fairly straightforward.

Note: if the surface moves/deforms, the texture goes with it.

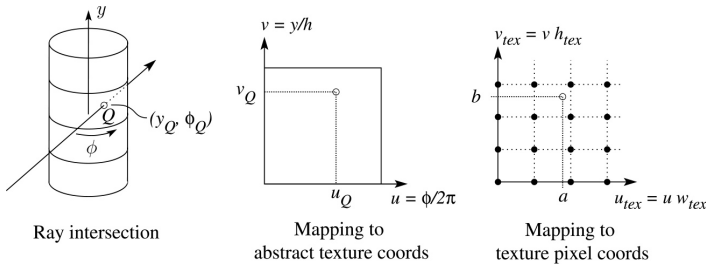
## Mapping to texture image coords

The texture is usually stored as an image. Thus, we need to convert from abstract texture coordinate:

$(u, v)$  in the range  $[[0..1], [0..1]]$

to texture image coordinates:

$(u_{tex}, v_{tex})$  in the range  $[[0.. w_{tex}], [0.. h_{tex}]]$

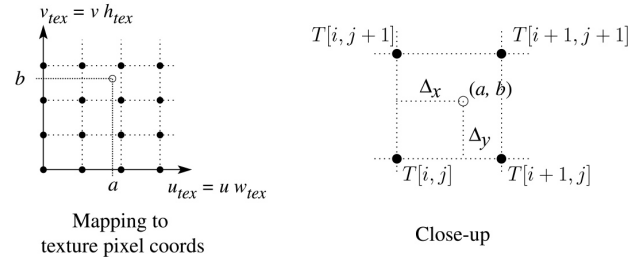


**Q:** What do you do when the texture sample you need lands between texture pixels?

5

## Texture resampling

We need to resample the texture:



A common choice is **bilinear interpolation**:

$$T(a, b) = T(i + \Delta_x, j + \Delta_y)$$

$$= \text{_____} T(i + \Delta_x, j) + \text{_____} T(i + \Delta_x, j + 1)$$

$$T(i + \Delta_x, j) = \text{_____} T[i, j] + \text{_____} T[i + 1, j]$$

$$T(i + \Delta_x, j + 1) = \text{_____} T[i, j + 1] + \text{_____} T[i + 1, j + 1]$$

$$T(a, b) = \text{_____} T[i, j] + \text{_____} T[i + 1, j] +$$

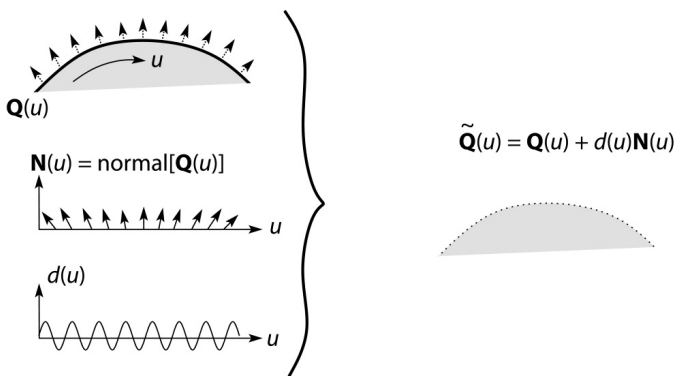
$$\text{_____} T[i, j + 1] + \text{_____} T[i + 1, j + 1]$$

6

## Displacement mapping

Textures can be used for more than just color.

In **displacement mapping**, a texture is used to perturb the surface geometry itself:



- These displacements “animate” with the surface

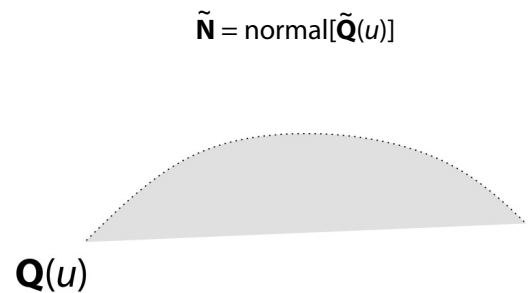
**Q:** Do you have to do hidden surface calculations on  $\tilde{Q}$ ?

7

## Bump mapping

In **bump mapping**, a texture is used to perturb the normal:

- Use the original, simpler geometry,  $Q(u)$ , for hidden surfaces
- Use the normal from the displacement map for shading:

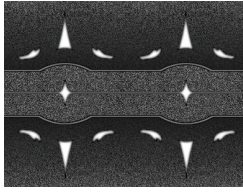


**Q:** What artifacts in the images would reveal that bump mapping is a fake?

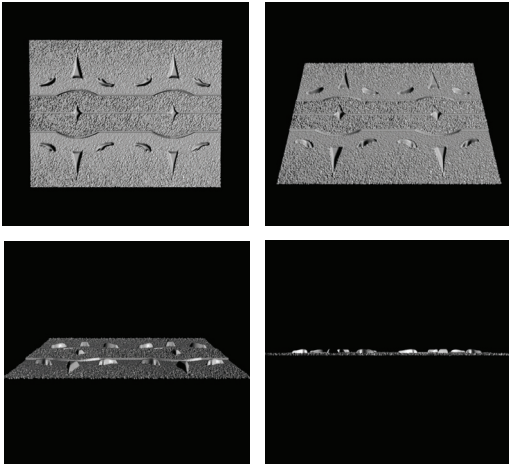
8

## Displacement vs. bump mapping

Input texture

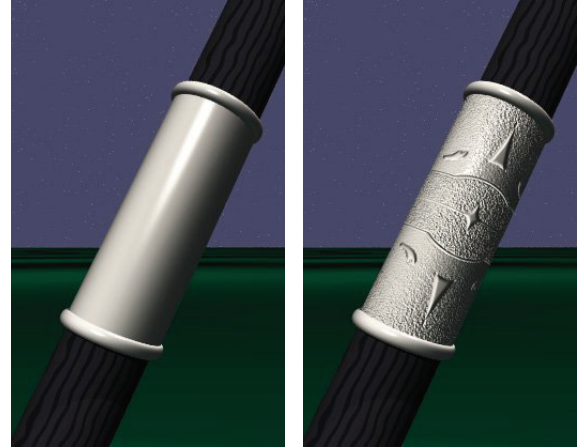


Rendered as displacement map over a rectangular surface



9

## Displacement vs. bump mapping (cont'd)



Original rendering

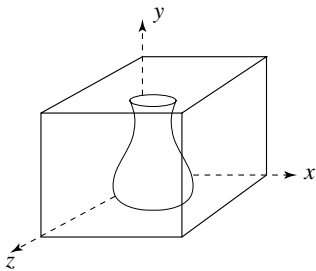
Rendering with bump map wrapped around a cylinder

*Bump map and rendering by Wyvern Aldinger*

10

## Solid textures

**Q:** What kinds of artifacts might you see from using a marble veneer instead of real marble?



One solution is to use **solid textures**:

- ◆ Use model-space coordinates to index into a 3D texture
- ◆ Like “carving” the object from the material

One difficulty of solid texturing is coming up with the textures.

11

## Solid textures (cont'd)

Here's an example for a vase cut from a solid marble texture:



*Solid marble texture by Ken Perlin, (Foley, IV-21)*

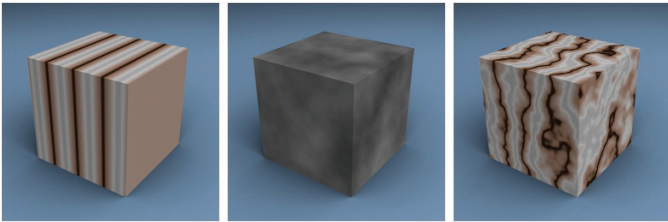
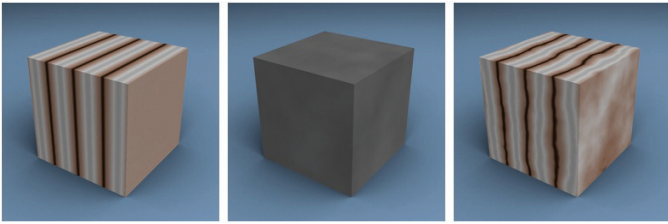
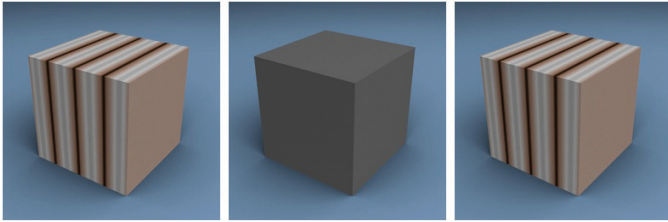
12

## Solid textures (cont'd)

$in(x,y,z) =$   
 $stripes(x)$

$shift(x,y,z) =$   
 $K \cdot noise(x,y,z)$

$out(x,y,z) =$   
 $stripes(x + shift(x,y,z))$

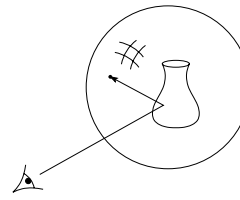


Increasing K

↓

13

## Environment mapping



In **environment mapping** (also known as **reflection mapping**), a texture is used to model an object's environment:

- ◆ Rays are bounced off objects into environment
- ◆ Color of the environment used to determine color of the illumination
- ◆ Really, a simplified form of ray tracing
- ◆ Environment mapping works well when there is just a single object – or in conjunction with ray tracing

Under simplifying assumptions, environment mapping can be implemented in hardware.

With a ray tracer, the concept is easily extended to handle refraction as well as reflection.

14