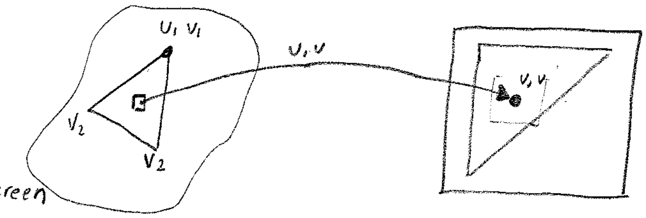
TEXTURE MAPPING 2

RECAP:

Object / Triangle

Texture Coordinate



Interpolation Linear in Space / Perspective on Screen

For each pixel do lookup for color

Bilinear interpolation since u,v is continuous

TRI-LINEAR interpolation in MIPMAP to get areas

Small GOTCHA

Lighting computed at VERTEX

Color at PIXEL

① do GOURAUD Shaing

Texture Modulates ← multiplies color

$$\text{color} = C_D (N\cdot L) + C_S (H\cdot N)^p + C_A L_a$$

$$\text{color} = C_T \left(\text{———} \right)$$

→ Make objects white / mult over color

→ Doesn't allow control of specular color sepeate

GL has a workaround

→ Modulate color

other choices (replace, subtract)

Opportunity to <u>combine</u> multiple color sources

Combine Multiple Textres = MULTI-TEXTURE

(more on that Later)

Additional Texture MAP TRICKS

Objects Look too flat

BUMP MAPPING ← use texture to change normal

need to compute lighting per-pixel (since normal changes)

Surface is flat, but reflects as if bumpy.

AKA Normal MAPS

NOTE: offsets to "real" normal are whats stored

for triangle have "u" and "v" vector

$$n = \underbrace{n + \alpha u + \beta v}_{\text{vectors from triangle}}$$

Only changes lighting

- no self-shadowing (hacks to fix)
- doesn't change sihouette
- per-pixel lighting required

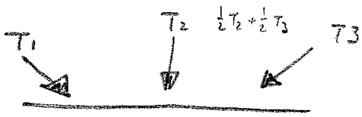
DISPLACEMENT MAPPING

- ACTUALLY MOVE POINT (which changes normal)
- Very hard to do (pixel might move to other pixel)
- Gets realistic effects

How to fake this without per-pixel computation

View Dependent Texture Mapping

- determine what object looks like under different view directions
- Blend between different textures



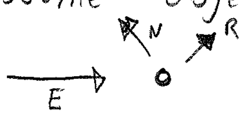
Use of MULTI-TEXTURE
1. Multipass w/ multiply or blend
2. Texture Combining

- Use texture over whole triangle
  per-pixel effects pre-computed (need some way to compute them)
- can vary based on light direction as well

MORE MAPPING:

Environment mapping

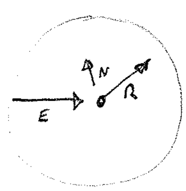
- make mirror reflections
- assume object is infinitessimal sphere



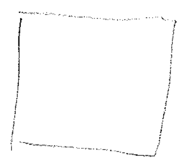
R depends on E, N

assume E is constant

Use R to Look up into Map of "Environment"



← spherical environment map



cubic environment map

Cylindrical environment map

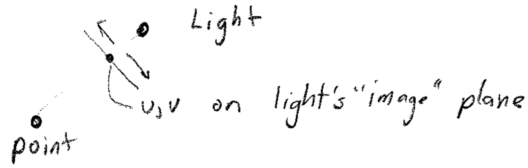
Lighting w/ Texture

Environment Map is mirror reflection (specularity)
put lights in it to get realistic lights
very bright spots in textre (use of HDR)

"Paint" Lighting Details on to objects

→ MULTI - TEXTURE TO ADD LAYERS OF LIGHTS

→ Slide Projector Mapping



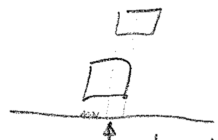
Shadow MAPS - something different
1. render scene from lights point of view
2. visible objects are lit, occluded are in shadow
   render and keep the z-buffer (the shadow map)
3. draw from the camera's viewpoint
   for any pixel, see its distance from light
   check in map to see it occluded

Hack Shadows / Hack Lights

draw black or light splotches

How to control where they go?

How to avoid overdraw



shouldn't be twice as dark, but if using blend w/ black it will be

Stencil Buffer

A buffer you can do anything with

Write values w/ drawing

Test values w/ drawing

Example

1. clear to zero
2. draw ground set stencil bit
3. draw shadows
   - only draw when stencil bit set
   - reset stencil bit when drawing