12/4

Administrivia
- Exam time 12:25 in social sciences
- Quiz scores n+1 /46 = %
  80% = A (≈ 37)
  60% = B (≈ 27.5)
- Demo times
- LAST LECTURES
  Today: Real Time / Hack Rendering
  Thursday: Image-Based Modeling + Rendering (EVALS?)
  Tuesday: Animation
  Thursday: HOT TOPICS (wrap up and review)

TODAY: REAL-TIME RENDERING

LAST TIME: Making nice pictures is tough
This time - how do we do it fast

Short answer: Cheat!

Better answer:
understand what you can do fast
(hardware capabilities)
figure out how to use this in weird ways

WARNING:
HARDWARE IS MOVING TARGET

SINCE HARDWARE IS MOVING TARGET:
- look for generally timeless principles
- understand today so can learn about tommorow's when it comes

What can we do?
Triangles - shaded (GOURAUD → per pixel)
z-buffered
textured (increasingly fancy)
Other buffers
DOUBLE, Z,
Accumulation + Stencil
Programmable Hardware (2001)

What strategies?
Use geometry for visual effects / lights
Multiple passes
⇒ learn about stencil
Use precomputation / texture
⇒ multi-texturing
⇒ shadow maps
Use texture in new ways
⇒ environment mapping
⇒ bump mapping
⇒ displacement mapping

Draw Geometry to fake effects
- we can draw lots of polygons!

Light Cones (from P3 demo)

Shadows -
multipass - each object drawn multiple times
Reflections -
identically (just need to set up matrix)

Problem - how to keep the shadow on the ground?

Problem - how to dim ground w/o totally blackening it

alpha ⇒ multiple regions get darker

Answer:
① color hacks
② limit where we draw

12/4 - 4

Stencil Buffer

General Tool in GL

Used for lots of things -

people are very creative!

Idea: Integer Valued Buffer -

write anything you want

Do a per pixel test (like z-buffer)

Example:

Clear Stencil when Clear Screen

Set Stencil to 1 when colored

⇒ Draw Ground

Stencil Test - only draw if not zero

⇒ Draw Shadows

Example 2:

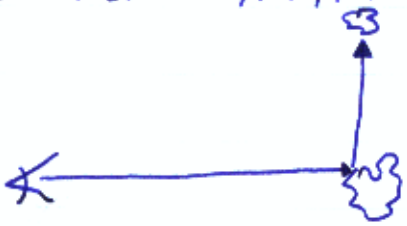
Use "increment stencil" mode

Only draw shadows if stencil =

Stencil Buffers are used in all kinds of ways - very creatively!

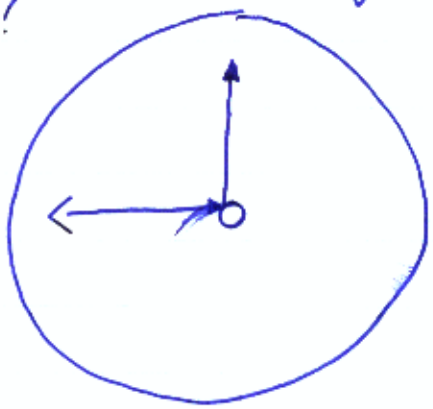
12/4-5

Environment Mapping - Hack Reflections



Color depends on -
- normal
- object position

If object is far away - who cares about position?



paint "world" onto Environment MAP
where to look depends only on normal and eye-vector
assume ray @ center of sphere

Approximation: reflected objects are far away

QUESTION ≡ What shape to make Environment?
- Sphere? (how to map?)
- Cube?
- Cylinder?

Use for specular highlights

12/4 - 6

Bump Mapping

How to make the surface <u>seem</u> rough

Need to vary lighting / Shading

Lighting depends on normal

Bump Mapping - Texture Map for Normals

Problems?

Silhouettes

Self-Occlusions / Self-Shadowing

Still is flat

lighting calculations per pixel

Displacement Mapping

geometric displacements per pixel

really does change geometry

lose simplicity of scanning triangles

12-4/7

Accumulation Buffer

Sum Up Multiple Images
Filtering

Filter over time (motion blur)
Filter in space (anti-aliasing)
Depth of Field
Move Eye-Point
Keep "in focus plane"