

# Shadow & Normal Mapping

---

CS 248 W'12  
Alexander Chia  
(ac@alexchia.com)

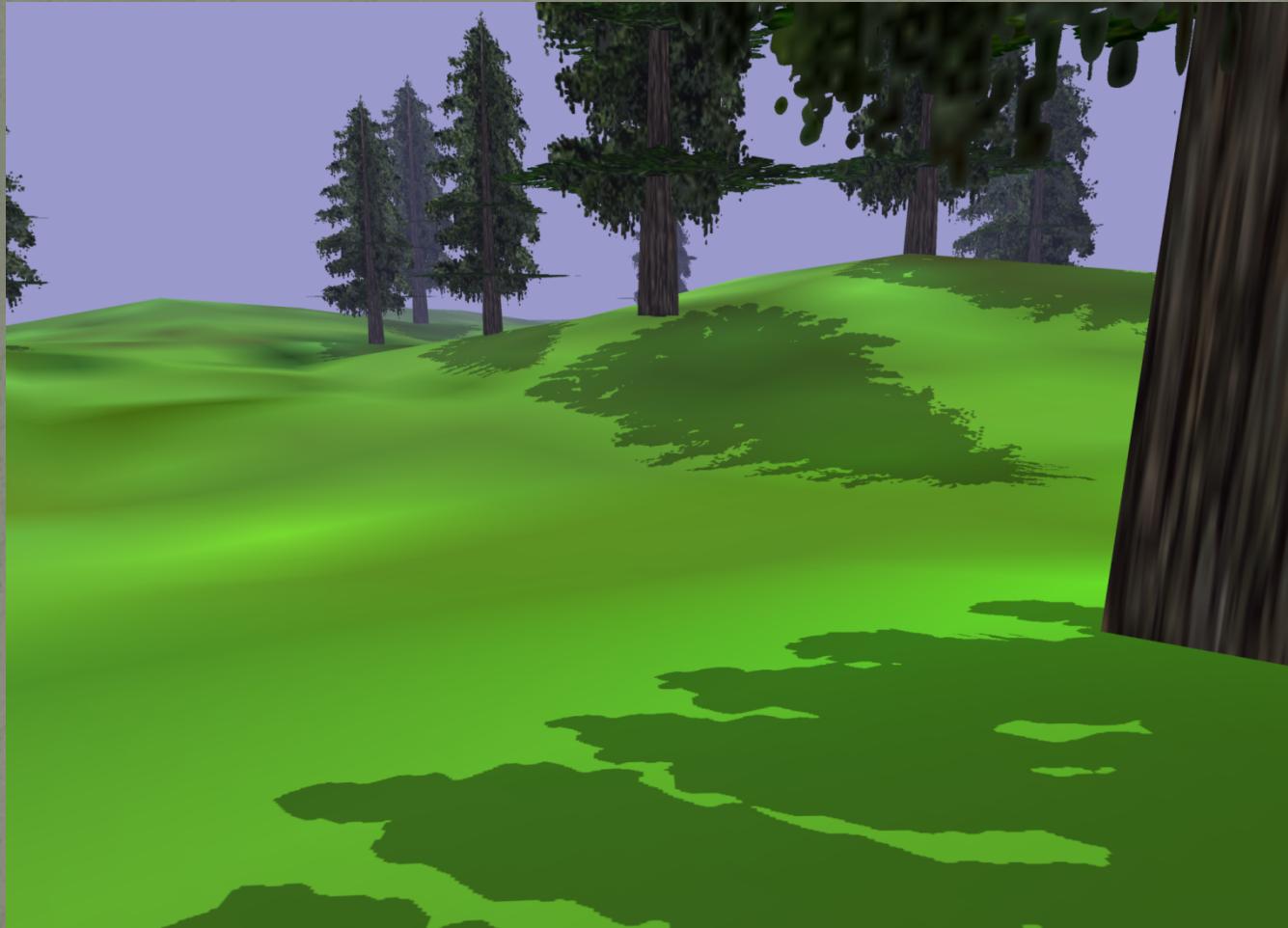
# Outline

- Using custom matrices in shaders
- Shadow Mapping
- Normal Mapping
- Questions

# Using matrices in shaders

- (Main.cpp)
  - glMatrixMode(GL\_TEXTURE)
  - glActiveTextureARB(GL\_TEXTURE5\_ARB)
  - glLoadIdentity(); glMultMatrixf(..)
  - ...
- (SomeVertexShader.gsl)
  - ... = gl\_TextureMatrix[5] \* ...

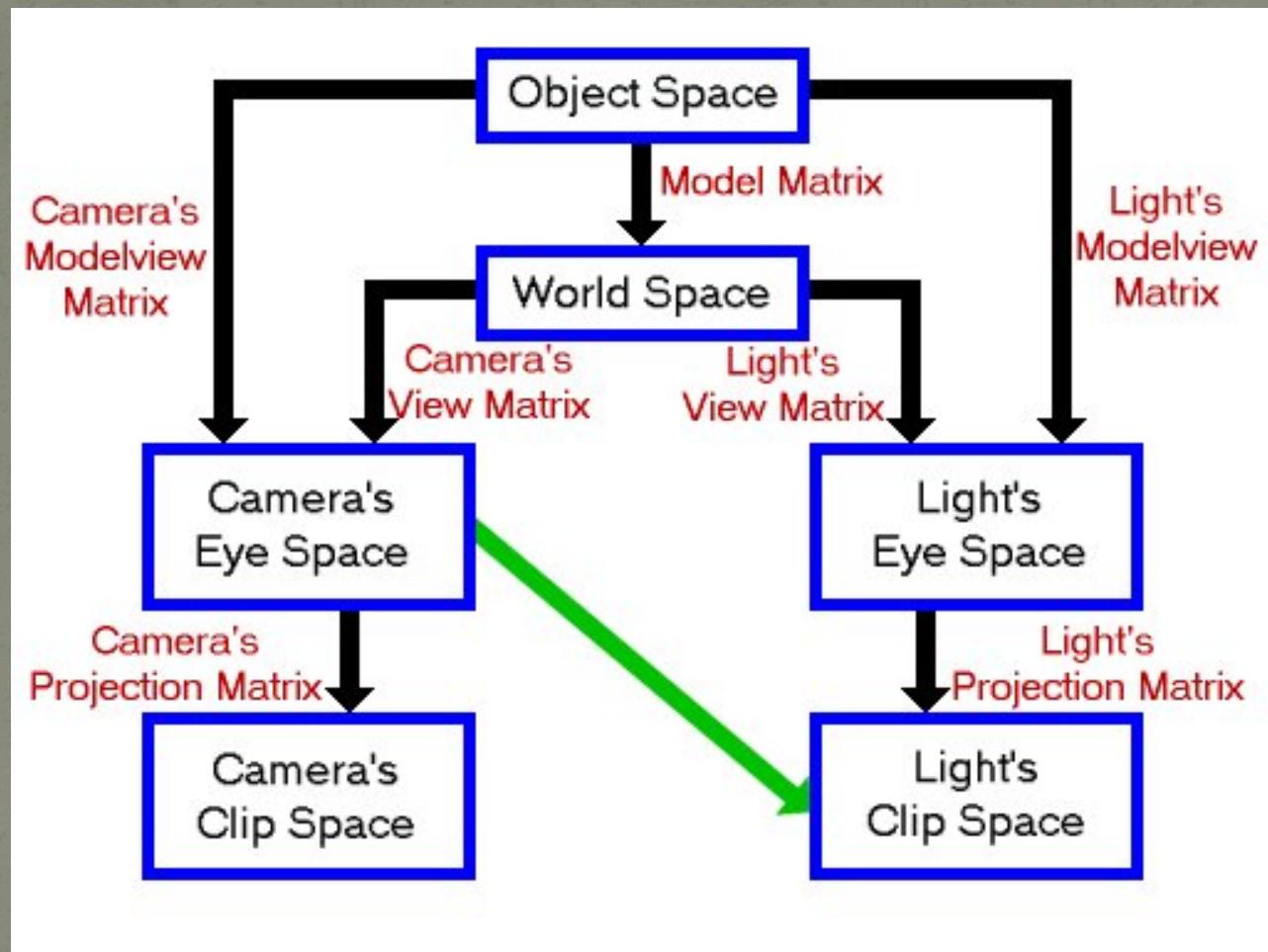
# Shadow Mapping



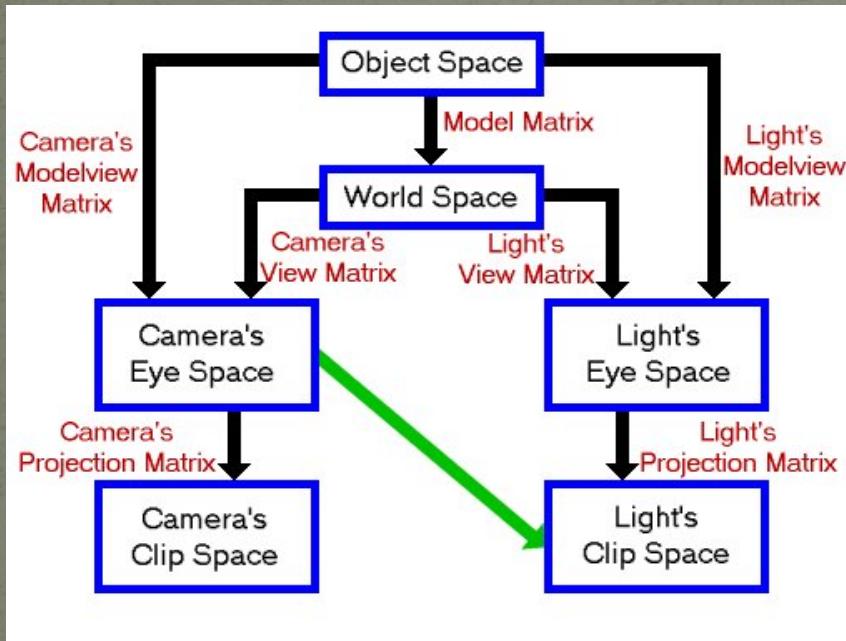
# Shadow Mapping (Concept)

- Render image from POV of light to create shadow map
- (ie. what would the scene look like if rendered from the POV of the light?) Light's view matrix = gluLookAt? glOrtho?
- When rendering a pixel, deciding if it is in shadow?
  - Project into light clip space
  - Compare z values (ie. distance from light source)
    - Distance from light > Z value of rendered texel in shadow map => occluding object => shadow!

# Shadow Mapping



# Shadow Mapping



- Many ways to project to Light clip space
- Raw vertex -> Model matrix->Light view matrix -> Light projection matrix?
- Transformed vertex -> Inverse camera view matrix -> Light view projection matrix?
- You choose!

# Shadow Mapping

- Light clip space  $\sim [-1, 1] \rightarrow$  Light texture space  $\sim [0, 1]$
- Multiply by bias matrix

- $$\begin{bmatrix} .5 & 0 & 0 & 0 \\ 0 & .5 & 0 & 0 \\ 0 & 0 & .5 & 0 \\ .5 & .5 & .5 & 1 \end{bmatrix}$$

# Shadow Mapping

- `shadow2DProj(shadowMap, shadowCoord)` in fragment shader
  - Compares z value of `shadowCoord` with corresponding texel in `shadowMap`
  - Returns a vector. First value in vector (`.r/.x`) = 0.0 if in shadow, 1.0 if not in shadow
- Self-shadowing
  - Add a bias to z..
- Precision of depth buffer

# Normal Mapping



# Normal Mapping

- The normal map is simply a texture that contains the normal vector in *tangent space*
- use TBN matrix – tangent bitangent normal to transform the normal vector sampled from the texture to the current *tangent space*
- Normal vector's .xyz is the .rgb values of texel sampled from texture

# Questions