

# Hardware Shadow Mapping

## *Putting It All Together*

Andrew Van Pernis

`arakel@vr.clemson.edu`

Clemson University



# Outline

# Outline

- Motivation
- More GL Extensions
- Shadow Maps



# Motivation

# Motivation

**Shadows add a sense of depth and realism.**



# More GL Extensions

# More GL Extensions

- `GL_ARB_texture_border_clamp`

# More GL Extensions

- `GL_ARB_texture_border_clamp`
- `GL_SGIS_generate_mipmap`



# More GL Extensions

- `GL_ARB_texture_border_clamp`
- `GL_SGIS_generate_mipmap`
- `GL_SGIX_depth_texture`

# More GL Extensions

- `GL_ARB_texture_border_clamp`
- `GL_SGIS_generate_mipmap`
- `GL_SGIX_depth_texture`
- `GL_SGIX_shadow`

# More GL Extensions

- `GL_ARB_texture_border_clamp`
- `GL_SGIS_generate_mipmap`
- `GL_SGIX_depth_texture`
- `GL_SGIX_shadow`
- `GL_NV_register_combiners`

# GL\_ARB\_texture\_border\_clamp

- Clamps texture to border color
- Border color is defined through `glTexParameter...`

# GL\_SGIS\_generate\_mipmap

- Fast mipmap generation
  - Supported by NVIDIA
  - `glTexParameteri (GL_TEXTURE_2D ,  
GL_GENERATE_MIPMAP_SGIS ,  
GL_TRUE ) ;`
- Requires main texture by power of 2

# GL\_SGIX\_depth\_texture

- Supports textures with an internal format of `GL_DEPTH_COMPONENT`
- Allows direct copying of depth information
- Supports 16, 24, and 32 bits of depth information

# GL\_SGIX\_SHADOW

- Defines a comparison operation with the R texture coordinate
- Allows for less than or equal to or greater than or equal to comparisons
- Returns a texture value of 0 or 1 based on the comparison

# GL\_NV\_register\_combiners

- Fancier version of the ARB combiners
- Maybe worth another talk



# Shadow Maps

# Basic Algorithm

- 1). Update matrices for eye and light
- 2). if (need\_depth\_map\_update)  
    Update the depth map
- 3). Activate automatic texture coordinates
- 4). Configure register combiners
- 5). Draw scene
- 6). Disable register combiners
- 7). Deactivate automatic texture coordinates
- 8). Repeat

# Matrix update

- Faster if done in software
- Full 4x4 matrix is needed for automatic texture coordinate generation

# Updating the depth map

- Only update the depth component of the framebuffer by using `glColorMask...`
- Draw from the point of view of the light
- Copy the depth component to a texture

# Automatic texture coordinates

- Map the space of the light into the eye space
- Use `glTexGen...`
- Set up so that the R coordinate is equivalent to depth

# Register combiners

- `GL_SGIX_shadow` comparison is a boolean operation
- Use combiners to blend the shadow into the scene for a more natural look

# Results

