Shaders: Review

CSCI 4830/7000
Advanced Computer Graphics
Spring 2009

What is it?

- A program used to replace parts of the (OpenGL) graphics pipeline
- Vertex Shader sets Vertex Attributes
- Fragment Shader sets Pixel Attributes
- Geometry Shader turns one Vertex into More Vertexes

Why is this better?

- Instructions runs on GPU
  - fast
  - parallel
  - support for "typical"operations
- Ultimate flexibility over behavior
  - C-like instructions
  - control flow (if/for/while) for complex decisions
  - convenient extensions (vec4, mat4)

GL Shader Language (GLSL)

- Often call "GLSLang"
- Added to OpenGL 2.0
  - First appeared as extension in OpenGL 1.5
  - Can be accessed in older versions using extentions
  - GL Extension Wrangler (GLEW) often used
- Geared to real time graphics
  - Inserted into OpenGL pipeline
  - Vertex Shader to manipulate vertexes
  - Fragment Shader to manipulate pixels
  - Geometry Shader as extension

GLSL Variable Qualifiers

- uniform (e.g. gl_ModelViewMatrix)
  - input to vertex and fragment shader from OpenGL or application [read-only]
- attribute (e.g. gl_Vertex)
  - input per-vertex to vertex shader from OpenGL or application [read-only]
- varying (e.g. gl_FrontColor)
  - output from vertex shader [read-write], interpolated, then input to fragment shader [read-only]
- const (e.g. gl_MaxLights)
  - compile-time constant [read-only]

Topic Covered

- Procedural Textures
- Lighting
- Image Processing
- Fun with Textures
  - Blue Marble
  - Ichor
  - Noise
  - Shadows
- Particle Systems

Procedural Textures

- What goes in which shader
  - Lighting in vertex shader
  - Texture in fragment shader
- Passing information between shaders
- Control flow in shaders
- Debugging shaders

Lighting

- Review of lighting
  - Phong vs. Blinn-Phong
  - Vertex vs Pixel
- Components of lighting
  - Ambient, Diffuse, Specular
- Using functions and textures in GLSL

Image Processing

- Draw scene as before
- Take screen shot of window
- Redraw window applying image processing
- Image processing
  - Blur, sharpen, erosion, dilation
- Edge detection
  - Laplacian, Prewitt, Sobel
- Multiple passes
- Functions in GLSL

Fun with Textures 1

- Ichor
  - Textures used to store density and velocity fields
- Blue Marble
  - Daytime textures for each month
  - Nighttime textures
  - Gloss and clouds
-

Fun with Textures 2: Noise

- Texture used to store complex function
- Noise used to make natural looking graphics
  - Rendering natural phenomena like clouds
  - Rendering materials like wood or concrete
  - Adding imperfections to surfaces like bumps
  - Adding imperfections to motion like jitters
-

Fun with Textures 3: Shadows

- Shadow mapping requires a second Z-buffer
- Draw image from light point of view
- Copy Z buffer to texture
- Draw image from eye point of view
  - Generate light position texture coordinates
  - Draw nearest to light as lit, others shadowed
- Use frame buffer object to keep data near GPU

Particle systems

- Use many points to represent system
  - Confetti cannon
  - Fire
- DrawPoints
  - Pass addition parameters to points
  - Vertex shader
    - Position
    - Color
    - Lifetime