CSE 5542: Real-Time Rendering

Instructor: Huamin Wang (whmin@cse.ohio-state.edu)
Credit: 3
Class: MWF 11:30 AM – 12:25 PM, Bolz Hall 0314
Office hours: MF 12:30PM – 1:30PM, DL 583
Webpage: http://www.cse.ohio-state.edu/~whmin/courses/cse5542-2013-spring/main.html
Prerequisite: CSE 3901/3902/3903 (project design)
MATH 2568 (linear algebra)
or permission from the instructor
Instructor Permission?

- Enthusiastic about Computer Graphics
- Fluent in C/C++ programming
- Comfortable with linear algebra (vector and matrix calculations)
- Willing to learn fast-changing technology (OpenGL and graphics hardware) by yourself
Reference books


- Very easy to read!
- Help you to understand the lectures and prepare for exams
- Many OpenGL/GLSL examples in C
Reference Books

Latest
OpenGL programmer’s Guide

Latest
OpenGL reference manual
Reference Books


OpenGL shading Language 3rd edition by Randi Rost
Grading

- Four Labs: 40%
  2D and 3D drawing – transformation, lighting, texture mapping, etc.
- Final Lab: 15%
- Exam: 30%
- Written assignments: 15%

<table>
<thead>
<tr>
<th>• Grader:</th>
<th>TBA</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Office Hours:</td>
<td>TBA</td>
</tr>
<tr>
<td>• Grader Office:</td>
<td>TBA</td>
</tr>
</tbody>
</table>
Graphics Topics

Geometry
(How to represent the shape)

Rendering
(How to generate an image)

Animation
(How to deform the shape)
Graphics Applications

Electronic games

Movies/TV/Commercials

Computer-Aided Design (CAD)

Visual arts

Scientific visualization
Real-Time Rendering

- Rendering: using computers to generate 2D images from 3D scenes.
  - Is Photoshop doing rendering?
- Real-Time: images are generated fast (≥30FPS).
  - The movie industry and the game industry, which one prefers real-time rendering?
What will I learn from this course?

• A basic understanding of graphics hardware/software
• Learn how to use OpenGL/GLSL to control graphics hardware for 2D/3D drawing
• Advanced real-time rendering algorithms
Specific Topics

• Overview of Graphics Hardware and Software
• Coordinate systems
• OpenGL geometry drawing
  – OpenGL vertex buffer objects
• OpenGL Shading Language (GLSL)
  – Vertex and fragment shaders
• 3D transformation and Viewing
• Quaternion and virtual trackball
Specific Topics

• Illumination
  – Flat, Gourald, Phong shading models
  – Fixed function pipeline and shaders

• Visibility and Z-buffering

• Texture Mapping
  – Image and procedural textures

• Bump, environment, and projective texture mapping
Specific Topics

• Real-time shadows

• Advanced topics in shaders
  – Geometry shader
  – Tessellation shader

• Advanced topics in rendering and graphics applications
Where do I do my labs?

• Your own machine … or
• Graphics PC Lab – CL 112D
• Graphics cards that supports at least OpenGL 2.1
• Platforms: PC (visual studio), Mac OS X or Linux