Course Overview

Administrative Issues
Modeling
Animation
Rendering
OpenGL Programming

January 15, 2002
Frank Pfenning
Carnegie Mellon University
http://www.cs.cmu.edu/~fp/courses/graphics/

Course Information On-Line

• http://www.cs.cmu.edu/~fp/courses/graphics/
  – Schedule (slides, readings)
  – Assignments (details, due dates)
  – Software (libraries, hints)
  – Resources (books, tutorials, links)
• news:cmu.cs.class.cs462

About Me

• Research: Programming Languages & Logic
• Teaching: Anything
• http://www.cs.cmu.edu/~fp/
• Office Hours
  – Wed 2:30-3:30, WeH 8117
  – Right after class
  – By appointment

Teaching Assistants

• Michael Henson (Wed 6:00-8:00)
• John Ketchpaw (Mon 6:00-8:00)
• Shayan Sarkar (TBA)
• Available in new graphics lab, WeH 5336
• Card reader for access (email me if denied)
• Instructions for account setup on web page soon

Prerequisites

• 15-213 Intro to Computer Systems
• 21-241 Matrix Algebra
• 21-259 Calculus in 3D
• See me if you are missing any and we haven’t discussed it

Postrequisites

• 05-831 Building Virtual Worlds (Pausch, F’02)
• 15-4xx Game Programming (Kuffner, F’02)
• 15-497 Computer Animation (Hodgins, S’03)
• 15-463 Computer Graphics II (Heckbert, S’03)
Textbook

- **Interactive Computer Graphics**  
  A top-down approach with OpenGL  
  Edward Angel

- **OpenGL: A Primer**  
  Edward Angel

- Available bundled in CMU Bookstore

- Supplementary text:  
  **Computer Graphics: Principles and Practice**  
  Foley, van Dam, Feiner, Hughes  
  On reserve in library (soon)

Grading

- 45% Programming Assignments (4)
- 20% Written Assignments (4)
- 10% Midterm (one sheet of notes only, in class)
- 25% Final (open book)

- Alternating assignments  
  - Programming (2 weeks)  
  - Written (1 week)

- No collaboration!

Course Overview

- The computer graphics trinity  
  - Modeling: how to represent objects  
  - Animation: how to control and represent motion  
  - Rendering: how to create images  

- OpenGL graphics library

- **Not** in this course:  
  - Human-computer interaction  
  - Graphic design  
  - Graphics hardware  
  - DirectX API

Computer Graphics Goals I

- Synthetic images indistinguishable from reality  
- Practical, scientifically sound, in real time

Example: Ray Tracing

- 2001 Internet ray tracing competition, N. Kern

Example: Radiosity

- Lightscape by Autodesk
Computer Graphics Goals II

- Creating a new reality
- Practical, aesthetically pleasing, in real time

Example: Illustrating Smooth Surfaces

- SIGGRAPH 2000 Conference, A. Hertzmann, D. Zorin

Example: Image Analogies


1. Course Overview

- Administrative Issues
- Topics Outline (next)

2. OpenGL Basics

- Primitives and attributes
- Color
- Viewing
- Control functions
- [Angel, Ch. 2]

3. Input and Interaction

- Clients and servers
- Event driven programming
- Text and fonts
- [Angel, Ch. 3]
<table>
<thead>
<tr>
<th>4. Objects &amp; Transformations</th>
<th>5. Viewing and Projection</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Linear algebra review</td>
<td>• Orthographic projection</td>
</tr>
<tr>
<td>• Coordinate systems and frames</td>
<td>• Perspective projection</td>
</tr>
<tr>
<td>• Rotation, translation, scaling</td>
<td></td>
</tr>
<tr>
<td>• Homogeneous coordinates</td>
<td></td>
</tr>
<tr>
<td>• OpenGL transformation matrices</td>
<td></td>
</tr>
<tr>
<td>• [Angel, Ch. 4]</td>
<td>• Camera positioning</td>
</tr>
<tr>
<td></td>
<td>• Projections in OpenGL</td>
</tr>
<tr>
<td></td>
<td>• Hidden surface removal</td>
</tr>
<tr>
<td></td>
<td>• [Angel, Ch. 5]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>6. Hierarchical Models</th>
<th>7. Light and Shading</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Graphical objects</td>
<td>• Light sources</td>
</tr>
<tr>
<td>• Animations</td>
<td>• Ambient, diffuse, and specular reflection</td>
</tr>
<tr>
<td>• OpenGL routines</td>
<td>• Normal vectors</td>
</tr>
<tr>
<td>• Parameters and transformations</td>
<td></td>
</tr>
<tr>
<td>• [Angel, Ch. 8]</td>
<td>• Material properties in OpenGL</td>
</tr>
<tr>
<td></td>
<td>• Radiosity</td>
</tr>
<tr>
<td></td>
<td>• [Angel, Ch. 6]</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>8. Curves and Surfaces</th>
<th>9. Rendering</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Review of 3D-calculus</td>
<td>• Clipping</td>
</tr>
<tr>
<td>• Explicit representations</td>
<td>• Bounding boxes</td>
</tr>
<tr>
<td>• Implicit representations</td>
<td>• Hidden-surface removal</td>
</tr>
<tr>
<td>• Parametric curves and surfaces</td>
<td>• Line drawing</td>
</tr>
<tr>
<td>• Hermite curves and surfaces</td>
<td>• Scan conversion</td>
</tr>
<tr>
<td>• Bezier curves and surfaces</td>
<td>• Antialiasing</td>
</tr>
<tr>
<td>• Splines</td>
<td>• [Angel, Ch. 7]</td>
</tr>
<tr>
<td>• Curves and surfaces in OpenGL</td>
<td></td>
</tr>
<tr>
<td>• [Angel, Ch. 10]</td>
<td></td>
</tr>
</tbody>
</table>
10. Textures and Pixels
- Texture mapping
- OpenGL texture primitives
- Bump maps
- Environment maps
- Opacity and blending
- Image filtering
- [Angel, Ch. 9]

11. Ray Tracing
- Basic ray tracing [Angel, Ch. 6.10]
- Spatial data structures [Angel, Ch. 8.9]
- Motion Blur
- Soft Shadows

12. Physically Based Models
- Particle systems
- Spring forces
- Cloth
- Collisions
- Constraints
- Fractals
- [Angel, Ch. 11]

13. Scientific Visualization
- Height fields and contours
- Isosurfaces
- Volume rendering
- Texture mapping of volumes

Wildcards & Possible Guest Lectures
- Graphics hardware
- More on animation
- Motion capture
- Virtual reality and interaction
- Video game programming
- Non-photo-realistic rendering

Hot Application Areas
- Special effects
- Feature animation
- PC graphics boards
- Video games
- Visualization (science, architecture, space)
- The web
Hot Research Topics

• Modeling
  – getting models from the real world
  – multi-resolution

• Animation
  – physically based simulation
  – motion capture

• Rendering:
  – more realistic: image-based modeling
  – less realistic: impressionist, pen & ink

Acknowledgments

• Jessica Hodgins
• Paul Heckbert
• Joel Welling