Introduction to Computer Graphics

Prof. George Wolberg
Dept. of Computer Science
City College of New York
Course Description

• Intense introduction to computer graphics.
• Intended for advanced undergraduate and graduate students.
• Topics include:
  - OpenGL API, GLSL shading language
  - Geometric transformations
  - 3D viewing
  - Geometric modeling, curves and surfaces
  - Shading, texture mapping, compositing
<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction, history, vector/raster graphics</td>
</tr>
<tr>
<td>2-4</td>
<td>OpenGL, GLSL, Qt</td>
</tr>
<tr>
<td>5-6</td>
<td>Geometry, 2D/3D transformations</td>
</tr>
<tr>
<td>7</td>
<td>Texture mapping</td>
</tr>
<tr>
<td>8</td>
<td>Projections, perspective</td>
</tr>
<tr>
<td>9</td>
<td>3D viewing</td>
</tr>
<tr>
<td>10</td>
<td>Midterm</td>
</tr>
<tr>
<td>11-12</td>
<td>Shading</td>
</tr>
<tr>
<td>13-14</td>
<td>Curves and surfaces</td>
</tr>
</tbody>
</table>
Required Text

Supplementary Texts

• Supplementary Texts:
    • The definitive OpenGL programming reference
Grading

• The final grade is computed as follows:
  - Midterm exam: 25%
  - Final exam: 25%
  - Homework programming assignments: 50%

• Substantial programming assignments are due every three weeks.

• Proficiency in C/C++ is expected.

• Prereqs: CSc 221
Contact Information

• Prof. Wolberg
  - Office hours:
    • Wednesdays 1:00-2:00pm, NAC 8/202N
  - Email: wolberg@cs.ccny.cuny.edu

• Teaching Assistant (TA): Siavash Zokai
  - Email: ccny.cs472@gmail.com

• See class web page for all class info such as homework and sample source code:
  www-cs.ccny.cuny.edu/~wolberg/cs472
Objectives

• Broad introduction to Computer Graphics
  - Software
  - Hardware
  - Applications

• Top-down approach

• Shader-Based

• Programs in C/C++ will be assigned to reinforce understanding of the material
Prerequisites

- Good programming skills in C (or C++)
- Basic data structures
- Geometry
- Simple linear algebra
OpenGL Resources

• Can run OpenGL on any system
  - Desktop OpenGL on Windows, Mac, Linux
  - OpenGL ES on mobile platforms: iOS, Android

• Get Qt from www.qt.io/download-open-source
  - Graphical user interface toolkit for all platforms
  - Adds sliders, pushbuttons, advanced widgets to GUI

• www.opengl.org
  - Standards documents and sample code

• www.opengl-tutorial.org
  - Informative tutorials on basic and intermediate topics

• www.khronos.org
Outline: Part 1

• Part 1: Introduction
  - What is Computer Graphics?
  - Applications Areas
  - History
  - Image formation
  - Basic Architecture
Outline: Part 2

• Part 2: Modern OpenGL (shader-based)
  - Architecture
  - Qt for advanced GUIs
  - Simple programs in two and three dimensions
  - Basic shaders and GLSL
  - Interaction
Outline: Part 3

• Part 3: Texture Mapping
  - Buffers
  - Shader Applications
  - Compositing and Transparency
Outline: Part 4

• Part 4: Three-Dimensional Graphics
  - Geometry
  - Transformations
  - Homogeneous Coordinates
  - Viewing
  - Lighting and Shading
Outline: Part 5

• Part 5: Curves and Surfaces
  - Bezier Curves
  - Hermite Curves
  - B-Splines
  - Cubic Splines
  - Coons Patches
What is Computer Graphics?

Prof. George Wolberg
Dept. of Computer Science
City College or New York
Objectives

• In this lecture, we explore what computer graphics is about and survey some application areas
• But we start with a historical introduction
Computer Graphics

- \textit{Computer graphics} deals with all aspects of creating images with a computer
  - Hardware
  - Software
  - Applications
Related Fields
Example

• Where did this image come from?

• What hardware/software did we need to produce it?
Preliminary Answer

• **Application**: The object is an artist’s rendition of the sun for an animation to be shown in a domed environment (planetarium)

• **Software**: Maya for modeling and rendering but Maya is built on top of OpenGL

• **Hardware**: PC with discrete graphics (GPU) for modeling and rendering
Basic Graphics System

Input devices

Image formed in frame buffer

Output device
Can be used as a line-drawing device (vector graphics) or to display contents of frame buffer (raster graphics)

• Computer graphics goes back to the earliest days of computing
  - Strip charts
  - Pen plotters
  - Simple displays using A/D converters to go from computer to calligraphic CRT

• Cost of refresh for CRT too high
  - Computers slow, expensive, unreliable

- **Wireframe graphics**
  - Draw only lines
- Sketchpad
- Display Processors
- Storage tube

wireframe representation of sun object
Project Sketchpad

- Ivan Sutherland’s PhD thesis at MIT
  - Recognized the potential of man-machine interaction
  - Loop
    - Display something
    - User moves light pen
    - Computer generates new display
  - Sutherland also created many of the now common algorithms for computer graphics
Display Processor

- Rather than have host computer try to refresh display use a special purpose computer called a *display processor* (DPU)

  ![Diagram of display processor system]

- Graphics stored in display list (display file) on display processor
- Host *compiles* display list and sends to DPU

Angel/Shreiner: Interactive Computer Graphics 6E © Addison-Wesley 2012
Direct View Storage Tube

• Created by Tektronix
  - Did not require constant refresh
  - Standard interface to computers
    • Allowed for standard software
    • Plot3D in Fortran
  - Relatively inexpensive
    • Opened door to use of computer graphics for CAD community

• Raster Graphics
• Beginning of graphics standards
  - IFIPS
    • GKS: European effort
      – Becomes ISO 2D standard
    • Core: North American effort
      – 3D but fails to become ISO standard

• Workstations and PCs
Raster Graphics

- Image produced as an array (the \textit{raster}) of picture elements (\textit{pixels}) in the \textit{frame buffer}
Raster Graphics

• Allow us to go from lines and wireframes to filled polygons
PCs and Workstations

• Although we no longer make the distinction between workstations and PCs, historically they evolved from different roots
  - Early workstations characterized by
    • Networked connection: client-server model
    • High-level of interactivity
  - Early PCs included frame buffer as part of user memory
    • Easy to change contents and create images

Realism comes to computer graphics

smooth shading  environment mapping  bump mapping

• Special purpose hardware
  - Silicon Graphics geometry engine
    • VLSI implementation of graphics pipeline

• Industry-based standards
  - PHIGS
  - RenderMan

• Networked graphics: X Window System

• Human-Computer Interface (HCI)

- OpenGL API
- Completely computer-generated feature-length movies (Toy Story) are successful
- New hardware capabilities
  - Texture mapping
  - Blending
  - Accumulation, stencil buffers

• Photorealism
• Graphics cards for PCs dominate market
  - Nvidia, ATI
• Game boxes and game players determine direction of market
• Computer graphics routine in movie industry: Maya, Lightwave
• Programmable pipelines
Computer Graphics: 2010-

- Xbox, Playstation
  - Realistic rendering, animation
- Kinect sensor
  - Gesture recognition
- Touchscreen interfaces
  - Phones, tablets, Windows 10
- 3D scanning and printing
  - Editing tools for rapid prototyping 3D models
- Virtual reality
  - Oculus Rift, Samsung Gear VR, Google Cardboard