The Focus of this Course

• Architecture of interactive graphics systems.
• Representations for modeling 2D and 3D objects and scenes.
• Algorithms for generating images of 2D and 3D objects and scenes.
Vermeer Replica
(Foley, et al.)
Not the Focus of this Course

• The use of drawing, modeling or animation programs.

• The use of GUI toolkits.
Contacting Professor Ellman

- Office: 117 OLB
- Phone: 437-5991
- Email: thellman@vassar.edu
Office Hours

• M, W: 12pm-1pm
• Tu, Th: 1:30pm-2:30pm
• Or send email to make an appointment.
• Or just stop by my office.
Textbooks


• Rost, Randi J., *OpenGL Shading Language*, Third Edition. (Recommended)
Grading

• Assignment 1 is 15% of the grade.
• Assignment 2 is 25% of the grade.
• Assignment 3 is 25% of the grade.
• Assignment 4 is 25% of the grade.
• Class participation is 10% of the grade.
Submission of Homework

• Assignments are due:
  – At the midnight on the date specified. (If you attend our class that day.)
  – At the start of class on the date specified. (If you do not attend class that day.)
• Prepare a Zip file including the contents of the MS Visual Studio solution/project folder in which you have prepared your work.
• Go to the Vassar Moodle site for our Computer Science III class.
• Upload your Zip file to the drop box for the current assignment.
Lateness Policy

- Late homework will be accepted with a 10% grade penalty.
- But only up until the start of first class after the due date.
- Homework turned any later will receive no credit at all.
Class Wiki

https://www.cs.vassar.edu/courses/cs378-201303/top

- Overview of the course.
- Schedule of topics and readings.
- Programming assignments.
- Summary of grading policy.
- Professor Ellman’s lecture notes.
Communication

• I sometimes send email messages to the entire class.
• My messages typically contain hints or clarifications of homework assignments.
• I keep an archive of all such messages on the class web page.
Getting Help

• Send me an email describing your problem.

• Include a Zip file of your project.

• I will try to reply promptly by email.
Academic Integrity

• You may discuss general ideas with classmates.
• You must do each programming assignment entirely by yourself.
• You may not discuss or share programs with other students.
• Vassar regulations require the professor to report suspected violations of academic integrity to the Dean of Studies.
• Read the “Originality and Attribution” pamphlet.
Facilities, Languages, Libraries

• Facilities:
  – Asprey Lab (Windows).
  – Your own PC (Windows, Mac?).
• Languages:
  – C++
  – Microsoft Visual C++ IDE.
• Libraries:
  – C++ Standard Template Library (STL).
  – GLUT: Library combining OpenGL with window systems.
Vector Graphics Architecture

Host Computer

1. MOVE 10,10
2. LINE 10, 100
3. LINE 100, 10
4. LINE 10, 10
5. GOTO 1

Display Controller
Vector Graphics Architecture

- Refresh buffer holds display program.
- Display controller executes program many times per second.
- Display controller does random scanning.
- Can generate true analog images.
- Computational cost of generating image grows with image complexity.
Raster Graphics Architecture

Host Computer

Video Controller

100000000000000
110000000000000
101000000000000
100100000000000
111110000000000
Raster Graphics Architecture

- Refresh buffer holds array of pixels.
- Buffer may be separate or part of main memory.
- Image complexity does not impact refresh process.
- Can display filled solid regions.
- Graphics system must perform scan conversion, e.g., converting continuous lines into pixel array.
- Has problems with aliasing or jaggedness.
- Animation is computationally expensive.
Raster Graphics Terminology

- Pixel, Pixmap, Bitmap, Bitplane.
- Bilevel system, color system.
- Refresh buffer, frame buffer.
- Aliasing, Anti-aliasing.
Bit Planes (24 Bit Graphics)

- Each column represents one pixel.
- Eight bits for R, G and B.
- Intensity values: 0 … 255.
- Total of 24 bits for colors.
- In 32 bit word, remaining 8 bits are the alpha channel. (Used for mixing colors.)
Conceptual Framework of Interactive Graphics System

- Application Program
  - Model
- Graphics System
  - Model
- Other components
Graphics v. Image Processing

• Graphics does synthesis: Model → Image.

• Image processing does analysis:
  – Pattern recognition: Image → Model.

• Sometimes the distinction gets blurred.
  – E.g. Performance animation.
  – E.g. Texture mapped photographs.
  – Image → Model → Image.
Computer Graphics Applications

• Icon & window interfaces. (Mac OS, Windows)
• Desktop publishing. (Adobe Illustrator)
• Data visualization. (Tufte, Banchoff)
• Computer-aided design. (Home Depot)
• Real time process control. (Air Traffic Control)
• Computer-aided instruction. (Training Pilots)
• Feature Films. (Toy Story)
• Computer Games. (GTA, Unreal, etc.)
ShutterBug Credits

Produced by Tom Williams and H. B. Siegel, with the assistance of M. W. Mantle

All images rendered with PhotoRealistic RenderMan 3.2

Copyright Pixar, 1990


Copyright Addison-Wesley, 1990
Two Parts of Computer Graphics

• Modeling:
  – Data structures for representing 2D/3D objects and scenes.
  – Algorithms for creating and manipulating these data structures.

• Rendering:
  – Given a model of the world and a description of an observer.
  – Generate an image representing the scene as seen by the observer.
Issues in Modeling & Rendering

- Representing the size, shape, & texture of 3D objects.
- Representing light sources and their properties.
- Simulating the geometric optics of vision.
- Simulating the interaction of light and surfaces.
- Capturing an analog world in digital computing.
- Speed, speed, and more speed!
Overview of Course

• Introduction to OpenGL and GLUT software.
• Geometric transformations.
• Viewing objects in OpenGL.
• Modeling shapes and surfaces.
• Color, light, shading and texturing.
• Rendering algorithms.
• Animation.
Jorge Angles (BMRT)
Renzo Del Fabro (BMRT)
Jonathan Merritt (BMRT)
Kevin Odhner (POV)
N.B. and G.M. (POV)
Vermeer Replica
(Foley, et al.)