What is Computer Graphics?

Study of: digital synthesis and manipulation of visual content, and interaction with it
(Video: Zhang and James, Harmonic Fluids)
(Video: The AlloSphere)
(Video: The Hubble Ultra Deep Field in 3D)
Graphics is...

Rendering

Rendering in POV-Ray by Gilles Tran
Graphics is...

Modeling

Human head modeled in ZBrush (Shon Mitchell)
Procedurally generated model of Zurich (Parish and Müller, 2004)
Trees generated with L-systems (Talton et al., 2010)
Engine CAD drawing (SolidWorks Corp.)
Graphics is...

Animation

Safonova and Hodgins, 2007
Graphics is...

*Physical Simulation*

Losasso et al., 2008
Graphics is...

*Digital Capture*

Light Field Microscopy (Levoy et al., 2006, 2007)

Digital Michelangelo Project (Levoy et al., 2000)
Graphics is...

*Image Processing*

- Image Analogies (Hertzmann et al., 2001)
- Scene Completion with Millions of Photographs (Hays and Efros, 2007)
Graphics is...
Geometric Computing

Pose Oblivious Shape Signature (Gal et al., 2007)
Shadow Art (Mitra and Pauly, 2009)

Discovering Structural Regularity in 3D Geometry (Pauly et al., 2008)
Graphics is...

Entertainment and Interaction

Jurassic Park (Universal Pictures, 1993)

Toy Story, the first full length film produced entirely on computers (Pixar/Disney, 1995)

World of Warcraft (Blizzard, 2004)

Crysis (Crytek/EA, 2007)
Graphics is...

*Virtual Environments*

Second Life (Linden Research, Inc.)
Graphics is...

User Interfaces

Xerox Star (Xerox, introduced 1981)
Windows 7 (Microsoft, 2009)
iPhone (Apple, introduced 2007)

Wall-mounted multitouch display (Perceptive Pixel, 2007)
Virtual car design in the CAVE (U. Mich./Prince, 1996)
Graphics is...
Visualization

Wordle (Feinberg, 2009)

Streamgraphs of listening history & movie sales (Byron/Wattenberg (t); New York Times, 2008 (b))

Britain from Above (BBC, 2009)

Visualization of traffic conditions (Google, 2009)
Graphics is...

Design

\[ F = \frac{\text{Gm}_1 \text{m}_2}{r^2} \]
Graphics is...

Hardware

(GPU Images from anandtech.com)

Folding@Home (Pande et al.)
Graphics is...

Art

The Mandelbulb (Daniel White, 2009)
Course Outline

- Introduction
- Light and Color
- Digital Cameras and Displays
- Image Processing and Compression
- Drawing and 2D Geometry
- 3D Geometry
- Rendering
Course Outline

- Curves and Surfaces
- Animation and Simulation
- Interaction
- Visualization
- Programmable Graphics Hardware
- Typography and Layout
- Sampling, Aliasing and Special Topics
You should know...

- Basic linear algebra
  - Vectors, matrices, simple coordinate systems (cartesian, polar, spherical, cylindrical)
- Basic calculus
  - Limits, derivatives, integrals, notation
You should know...

- **C/C++**
  - Either is ok, but we provide more utility code for C++
  - Please write **standards-compliant, platform-neutral** code
    - Your code should compile with a C99/C++11 compiler, without changes, on **all** desktop platforms (Windows, Linux, OS X)
    - We will test your code on a Linux (or occasionally OS X) system...
      - ... but do **NOT** use platform-specific features
Warning!

- I am a huge stickler for **well-written code**
- This includes proper...
  - Names, capitalization
  - Indentation
  - Documentation (functions, classes, code blocks...)
  - Modularization and re-use
  - Overall code structure and organization
- Yes, I will reserve 5-10% of each assignment for code quality
Coding Tips

• Don’t reinvent the wheel
• Look at the supplied code frameworks for models to follow
• Pick a coding standard (e.g. https://google.github.io/styleguide/cppguide.html) and stick with it
  • It’s ok to disagree with a standard, especially on minor formatting conventions such as where to put an opening brace, but you should have a good reason for doing so, and be consistent in your own use
  • http://astyle.sourceforge.net can help with formatting, though it’s best to train yourself so it’s virtually impossible for you to write badly constructed/formatted code
• Coding is a design process
How to **pass** this course

- Attend the lectures
  - Yes, attendance is mandatory. I understand if you skip a lecture here or a lecture there, but wholesale bunking is a recipe for grade disaster.

- **Understand the concepts**
  - Do the assignments
    - Try to get the bonus credits
  - Do the exams
How to **fail** this course

- Skip the lectures
- Ignore the concepts and only memorize formulæ
- Slack on the assignments
- Miss the exams
- Violate academic integrity

[http://www1.iitb.ac.in/newacadhome/rules.jsp](http://www1.iitb.ac.in/newacadhome/rules.jsp)

- We will run your code through automatic code checkers. They are **remarkably good** (better than you are at covering up copying).
Resources

- Course website: http://www.cse.iitb.ac.in/~cs475
- Lecture slides, assignments, schedule, FAQ etc
- Textbooks
  - Shirley and Marschner, *Fundamentals of Computer Graphics*, 3rd ed. (in a pinch 2nd is ok)
    - Pretty much any decent OpenGL guide that covers programmable shaders (GLSL) is also ok
Grading Policy

- Assignments: 55%
  1. Mandelbrot set (due Mon, Aug 1, 11:59pm)
  2. Image filtering
  3. Raytracing
  4. Interactive game (Quake)
  5. Visualization
- Midterm: 15%
- Final: 30%
AS1: Mandelbrot Set (due Aug 1, 11:59pm)

- Fractal set embedded in 2D complex plane
- Studied by Brooks, Matelski, Mandelbrot etc.
- Set of complex $c$ s.t. $z \leftarrow z^2 + c$ does not diverge

Brooks & Matelski, 1978

Beyer, 2006
AS1: Mandelbrot Set (due Aug 1, 11:59pm)

Wolfgang Beyer, 2006