CSC 433/533
Computer Graphics

Alon Efrat
alon@cs.arizona.edu

Lecture 01
Introduction
Course Material Source
Credits

- Joshua Levine
- Cem Yuksel (@cem_yuksel)
- Vladlen Koltun
- Steve Marschner
- Justin Solomon
- Wojciech Matusik
- Adam Bargteil
- Many others...

Today’s Agenda

- Registration Issues? See me after class.
- Course webpage: here
- Go over syllabus and introduce the course
Required Course Materials


• ISBN 978-1482229394

• Book is available electronically through the library

• A number of other materials distributed through the webpage

Required Course Materials

• If you do not know Javascript, you will need a reference!

• Two electronic references:
  • https://eloquentjavascript.net/
  • http://speakingjs.com/
Course Grading

1. Assignments (6 total): 46%. Lowest one is dropped.
   - Includes both a written and programming component
2. Midterm Exam: 21%
3. Final Exam: 21%
4. Max(Midterm, Final): 12%

Prerequisites

- Programming skills are necessary! We will develop assignments in Javascript and WebGL (but will be willing to accept other submissions (processing.org, C++/OpenGL, Matlab, Java/python)
- Linear algebra is essential, but we will review many of the concepts throughout (so the course is self contained)
  - Mostly we will use very basics concepts from Linear Algebra: Vectors, Vectors additions, Dot product, Matrix Multiplication
- Previous experience in HCI, graphics, or visualization useful, but not required.
Course Policies

• Do:
  • Do your best to class prepared and ready to participate
  • Write clean, correct, documented, and tested code
  • Contact me if you have any special needs
  • Be considerate of others and respectful of them and their time.
  • Discuss problems with classmates before coding.

• Don’t:
  • Steal code, share answers (from both others in the class and outside of it)
  • Steal copyrighted materials
  • Be dishonest
  • Criticize people. Instead, critique ideas
  • Violate department/university policies (academic integrity, title IX, etc.)

Administrative Updates

• CAPS: Counseling & Psych Services: https://health.arizona.edu/

• Reminder from Academic Services: Be respectful of your TA!
Coding Policies

• Unless otherwise noted, you **must write your own code**. Collaborating with or sharing code with other people in this course is sometimes permitted, but sometimes plagiarism. Feel free to share code that does not address the algorithmic challenges. For example, GUI functionalities, sliders, reading/writing files - all legit to share.

  • You are allowed to **discuss** assignments with other students at the **conceptual** level.

  • You must **cite sources** for code snippets or ideas taken from **external sources**.

• OK to: get ideas from the book, from the web

• NOT OK to: share or distribute code, use ideas without attribution.

Computer Graphics
What is it? Why do we study it? Who does it?
CG Areas

- Imaging: Representing and manipulating 2D images
- Modeling: Specifying and Storing 3D objects
- Rendering: Creating 2D images from 3D
- Animation: Creating the illusion of motion through sequences of images.

Many Applications of CG!

- Entertainment: Video Games, Cartoons, Films
- CAD/CAM/CAE: Computer-aided design/modeling/engineering
- Visualization
- Medical Imaging
- Education / Training / Simulation
- Art
Course Expectations

What will you accomplish?

Course Objectives

- Describe and apply the foundations of computer graphics, including hardware systems, mathematics for computer graphics, light, and color;

- Understand and write programs that implement key concepts from two-dimensional graphics, including raster image formats, image and signal processing, image deformation, and rasterization;

- Understand and write programs that implement key concepts for rendering complex three-dimensional scenes, including projections and view transformations, visibility algorithms, ray tracing, scene descriptions, and spatial data structures;

- Become familiar with and implement partial prototypes for a variety of advanced topics in computer graphics; including texturing, texture mapping animation, physically-based modeling, procedural modeling, curves and surfaces, global illumination, implicit modeling, and/or interaction; and

- Understand Hardware-Accelerated techniques (Z-buffer, Shaders, GPU)
Assignments

- Six assignments expected
- We will use Javascript for at least the first few assignments (will discuss in class options for the rest)
- Submissions will be made using git and GitHub classroom (more details in the first assignment). At least for the first two assignments
- While the majority of grade-points (45%) will be devoted to the programming component of the assignment, but most assignments will also include a few written exercises.
Exams

• Will cover all material taught prior to the exam, but more specs to come.

• Written portions of assignments will be good practices for the exams

• Refer to the course calendar

Class Participation

• Attendance is not mandatory, and will not be collected but...

• I expect you to make your best effort to attend, on time, and a participating member of the audience.

• Discussions during lecture will be frequent.

• Have your webcam on will make the discussions more effective and enjoyable.

• Discussions on piazza will are encouraged.
Lec01 Reading

- The course syllabus
- FOCG, Ch. 1

Lec02 Recommended Reading

- See https://alonefrat.github.io/csc433-533/index.html#L02 for links on learning Javascript
Sample of course’s topics

- RGB Additive
- CMYK Subtractive

Aliasing and anti-aliasing
Temporal Aliasing: The Wagon Wheel Effect

http://youtu.be/0k2lhYk6Lfs
Shading