Computer Graphics

Adam Finkelstein
Princeton University
COS 426, Spring 2016
Overview

• Administrative stuff
  ◦ People, times, places, etc.

• Syllabus
  ◦ What will I learn in this course?

• Imaging
  ◦ Getting started …
Administrative Stuff

• Instructors
  ◦ Adam Finkelstein
  ◦ Huiwen Chang, Nora Willett, & Linguang Zhang

• Book
  ◦ *Computer Graphics with OpenGL, 4th Ed*,
    Hearn, Baker, and Carithers,

• Registration
  ◦ If you are not yet signed up, see me after class

• Web page
  ◦ www.cs.princeton.edu/courses/archive/spr16/cos426/
COS 426: Computer Graphics
Spring 2016

General Information

Description: This course will study topics in computer graphics, covering methods in image processing, modeling, rendering, and animation.

Prerequisites: The course is appropriate for students who have taken COS217 and COS226 (or the equivalent). Javascript will be used as the main programming language.


Instructors: Professor Adam Finkelstein with TAs: Huiwen Chang, Nora Willett, and Linguang Zhang.

Students: requires PU login

Time/place: Lecture:
  • Tue & Thu 3-4:20pm, Friend 006

Precepts:
  • Wed 7:30-8:30pm, Friend 004 and/or 008.

Office Hours:
  • Mon 2-3pm (TA, location TBA)
  • Wed 11am-noon (Prof, CS Bldg 424)
  • Fri 11am-noon (TA, location TBA)

Questions: We will use Piazza to handle Q&A this semester. Please post your questions there instead of mailing the staff, if possible.
Coursework

• Exams (25%)
  ◦ In class (3/10 and 4/28)

• Programming Assignments (50%)
  ◦ Assignment #1: Image Processing (due 2/21)
  ◦ Assignment #2: Modeling (due 3/6)
  ◦ Assignment #3: Rendering (due 4/10)
  ◦ Assignment #4: Animation (due 4/24)

• Final Project (25%)
  ◦ Your choice! (due Dean’s Date)
Programming Assignments

• When?
  ◦ Roughly every 2-3 weeks

• Where?
  ◦ Anywhere you want, e.g. home or clusters

• How?
  ◦ Javascript
  ◦ Some OpenGL (WebGL, GLSL)

• What?
  ◦ Basic feature lists
  ◦ Extra credit lists
  ◦ Art contest
Art Contest

- Everybody should submit entries!
  - 1 point for submitting
  - 2 points for winning

Cool Images/Videos
(Jimmy Zuber, CS 426, Spring 2014)

Bloopers
(Reed Tantiviramanond, CS 426, Spr15)

Characters for web banner
Collaboration Policy

• Overview:
  ○ You must write your own code
  ○ You must reference your resources
  ○ See policy on course web, and ask when in doubt

• It’s OK to …
  ○ Talk with other students about ideas, approaches, etc.
  ○ Get ideas from information in books, wikipedia, etc.
  ○ Use “support” code provided with our assignments

• It’s NOT OK to …
  ○ Show your code to another student (e.g. post on web)
  ○ Look at code written by another student
  ○ Leverage code acquired from other sources
Questions / Discussion

• Piazza (www.piazza.com)
  ◦ View announcements
  ◦ Post questions to the class
  ◦ Answer other students questions
  ◦ Set up for everyone enrolled as of today
  ◦ Use this instead of email to instructors/TAs (can send private messages)
Precepts

• When and Where
  ◦ Wed 7:30-8:30
  ◦ Friend 004 – just down the hall (with 008 as overflow)

• Attendance
  ◦ Topics vary, so attend the ones that help you
  ◦ Tomorrow: getting up to speed in Javascript
Overview

• Administrative stuff
  ○ People, times, places, etc.

➢ Syllabus
  ○ What will I learn in this course?

• Imaging
  ○ Getting started …
Introduction

• What is computer graphics?
  ○ Imaging = representing 2D images
  ○ Modeling = representing 3D objects
  ○ Rendering = constructing 2D images from 3D models
  ○ Animation = simulating changes over time
Syllabus

I. Imaging
II. Modeling
III. Rendering
IV. Animation

Image Processing
(Rusty Coleman, CS426, Fall99)

Modeling
(Andrew Werner, Spring 2014)

Rendering
(David Paulk, CS426, Spr2015)

Animation
(Riley Thomasson, Spring 2014)
Part I: Imaging

• Image Basics
  ◦ Definition
  ◦ Color models

• Image Representation
  ◦ Sampling
  ◦ Reconstruction
  ◦ Quantization & Aliasing

• Image Processing
  ◦ Filtering
  ◦ Warping
  ◦ Composition
  ◦ Morphing

Image Composition
  (Michael Bostock, CS426, Fall99)

Image Morphing
  (Ianf, Wikipedia)
Part II: Modeling

• Representations of geometry
  ◦ Curves: splines
  ◦ Surfaces: meshes, splines, subdivision
  ◦ Solids: voxels, CSG, BSP

• Procedural modeling
  ◦ Sweeps
  ◦ Fractals
  ◦ Grammars

(Brendan Chou, Spring 2014)
Part III: Rendering

• 3D Rendering Pipeline
  ◦ Modeling transformations
  ◦ Viewing transformations
  ◦ Hidden surface removal
  ◦ Illumination, shading, and textures
  ◦ Scan conversion, clipping
  ◦ Hierarchical scene graphics
  ◦ OpenGL

• Global illumination
  ◦ Ray tracing
  ◦ Radiosity

Global Illumination
(Diana Liao, CS 426, Spr15)
Part IV: Animation

• Keyframing
  ◦ Kinematics
  ◦ Articulated figures

• Motion capture
  ◦ Capture
  ◦ Warping

• Dynamics
  ◦ Physically-based simulations
  ◦ Particle systems

• Behaviors
  ◦ Planning, learning, etc.
Applications

- Entertainment
  - Computer-aided design
  - Scientific visualization
- Training
- Education
- E-commerce
- Computer art

Geri’s Game (Pixar Animation Studios)
Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- Computer art

Gear Shaft Design
(Intergraph Corporation)

Los Angeles Airport
(Bill Jepson, UCLA)

Boeing 777 Airplane
(Boeing Corporation)
Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- Computer art

Apo A-1
(Theoretical Biophysics Group, University of Illinois at Urbana-Champaign)

Visible Human
(National Library of Medicine)

Airflow Inside a Thunderstorm
(Bob Wilhelmson, University of Illinois at Urbana-Champaign)
Applications

• Entertainment
• Computer-aided design
• Scientific visualization

Training
• Education
• E-commerce
• Computer art

Desk Assembly
(Silicon Graphics, Inc.)

Driving Simulation
(Evans & Sutherland)

Flight Simulation
(NASA)
Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training

Education
- E-commerce
- Computer art

Forum of Trajan
(Bill Jepson, UCLA)

Human Skeleton
(SGI)
Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- Computer art

Interactive Kitchen Planner
(Matsushita)

Virtual Phone Store
(Lucent Technologies)
Applications

• Entertainment
• Computer-aided design
• Scientific visualization
• Training
• Education

→ Computer art

Blair Arch
(Marissa Range '98)
Overview

• Administrative stuff
  ○ People, times, places, etc.

• Syllabus
  ○ What will I learn in this course?

➤ Imaging
  ○ Let’s get started …
What is an Image?
What is an Image?

An image is a 2D rectilinear array of pixels

Digital image
What is a Pixel?
What is a Pixel?

Sample of a function at a position

$I(x,y)$

Digital image
What Function?
What Function?

Could be any function …

Art of Science
(Friend Center hallway)
What Function?

What about photographic images?

$I(x,y)$?

Digital photograph
Plenoptic Function

Each pixel of a photographic image is a function of radiance arriving at a sensor.
The 7D plenoptic function $L(x,y,z,\theta,\phi,t,\lambda)$ describes the radiance arriving ... 

- at any position $(x,y,z)$,
- in any direction $(\theta,\phi)$,
- at any time $(t)$,
- at any frequency $(\lambda)$
An idealized photographic image contains a 2D array of samples of the 7D plenoptic function:

- at a particular camera viewpoint,
- for 2D array of directions,
- at a certain time,
- at certain frequencies
Photographic Images

In practice, can’t measure plenoptic function directly

- Photoreceptors in eye
- Film in a traditional camera
- CCD cells in digital camera
Photographic images

Photographic pixels as finite samples of the plenoptic function

\[ f(x, y, z, \theta, \phi, t, \lambda) \]

Digital photograph
What Frequencies?

\[ f(x, y, z, \theta, \phi, t, \lambda) \]
Electromagnetic Spectrum

• Visible light frequencies range between...
  ○ Red = $4.3 \times 10^{14}$ hertz (700nm)
  ○ Violet = $7.5 \times 10^{14}$ hertz (400nm)

Figures 15.1 from H&B
Color

• The color of light is characterized by its spectrum
  ◦ Magnitude of energy at every visible frequency

Figures 15.3-4 from H&B
How do we represent a color in a computer?

Must store a finite amount of data to represent magnitudes for infinite number of frequencies

White Light

Orange Light

Figures 15.3-4 from H&B
Color Frame Buffer

<table>
<thead>
<tr>
<th></th>
<th>255</th>
<th>150</th>
<th>75</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Blue channel
Green channel
Red channel
Frame Buffer Display

- Video display devices
  - Liquid Crystal Display (LCD)
  - Plasma panels
  - Thin-film electroluminescent displays
  - Light-emitting diodes (LED)

- Hard-copy devices
  - Ink-jet printer
  - Laser printer
  - Film recorder
  - Electrostatic printer
  - Pen plotter
Frame Buffer Display

Example: liquid crystal display (LCD)

Figure 2.16 from H&B
Frame Buffer Display

LCD up close

Colors are interleaved
Color

Why red, green, and blue (RGB)?
Human Color Perception

Spectral-response functions of each of the three types of cones on the human retina.

Tristimulus theory of color

Figure 13.18 from FvDFH
### RGB Color Model

Colors are additive

<table>
<thead>
<tr>
<th>R</th>
<th>G</th>
<th>B</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>Black</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>Red</td>
</tr>
<tr>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>Green</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>Blue</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>Yellow</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0</td>
<td>1.0</td>
<td>Magenta</td>
</tr>
<tr>
<td>0.0</td>
<td>1.0</td>
<td>1.0</td>
<td>Cyan</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>White</td>
</tr>
<tr>
<td>0.5</td>
<td>0.0</td>
<td>0.0</td>
<td>?</td>
</tr>
<tr>
<td>1.0</td>
<td>0.5</td>
<td>0.5</td>
<td>?</td>
</tr>
<tr>
<td>1.0</td>
<td>0.5</td>
<td>0.0</td>
<td>?</td>
</tr>
<tr>
<td>0.5</td>
<td>0.3</td>
<td>0.1</td>
<td>?</td>
</tr>
</tbody>
</table>
RGB Color Cube

Figures 15.11&15.12 from H&B
Other Color Models

- CMY
- HSV
- XYZ
- La*b*
- Others

Different color models are useful for different purposes
CMY Color Model

Useful for printers because colors are subtractive

<table>
<thead>
<tr>
<th>C</th>
<th>M</th>
<th>Y</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>White</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0</td>
<td>0.0</td>
<td>Cyan</td>
</tr>
<tr>
<td>0.0</td>
<td>1.0</td>
<td>0.0</td>
<td>Magenta</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>1.0</td>
<td>Yellow</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>0.0</td>
<td>Blue</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0</td>
<td>1.0</td>
<td>Green</td>
</tr>
<tr>
<td>0.0</td>
<td>1.0</td>
<td>1.0</td>
<td>Red</td>
</tr>
<tr>
<td>1.0</td>
<td>1.0</td>
<td>1.0</td>
<td>Black</td>
</tr>
<tr>
<td>0.5</td>
<td>0.0</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.5</td>
<td>0.5</td>
<td></td>
</tr>
<tr>
<td>1.0</td>
<td>0.5</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>

Plate II.7 from FvDFH
CMY Color Model

Figure 15.14 from H&B
HSV Color Model

Useful for user interfaces because dimensions are intuitive

<table>
<thead>
<tr>
<th>H</th>
<th>S</th>
<th>V</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.0</td>
<td>1.0</td>
<td>Red</td>
</tr>
<tr>
<td>120</td>
<td>1.0</td>
<td>1.0</td>
<td>Green</td>
</tr>
<tr>
<td>240</td>
<td>1.0</td>
<td>1.0</td>
<td>Blue</td>
</tr>
<tr>
<td>*</td>
<td>0.0</td>
<td>1.0</td>
<td>White</td>
</tr>
<tr>
<td>*</td>
<td>0.0</td>
<td>0.5</td>
<td>Gray</td>
</tr>
<tr>
<td>*</td>
<td>*</td>
<td>0.0</td>
<td>Black</td>
</tr>
<tr>
<td>60</td>
<td>1.0</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>270</td>
<td>0.5</td>
<td>1.0</td>
<td></td>
</tr>
<tr>
<td>270</td>
<td>0.0</td>
<td>0.7</td>
<td></td>
</tr>
</tbody>
</table>
HSV Color Model

- HSV interpretation in terms of color spectrum:
  - Hue = dominant frequency (highest peak)
  - Saturation = excitation purity (ratio of highest to rest)
  - Value = luminance (area under curve)

![Energy vs. Frequency Graphs]

White Light

Orange Light

Figures 15.3-4 from H&B
XYZ Color Model (CIE)

Derived from perceptual experiments
All spectra that map to same XYZ give same visual sensation
XYZ Color Model (CIE)

Normalized amounts of X and Y for colors in visible spectrum

Figure 15.7 from H&B
XYZ Color Model (CIE)

Useful for reasoning about coverage of color gamuts
XYZ Color Model (CIE)

Compare Color Gamuts

Identify Complementary Colors

Determine Dominant Wavelength and Purity

Useful for characterizing perceptual qualities of colors
La*b* Color Model

Non-linear compression of XYZ color space based on perception

Useful for measuring perceptual differences between colors
Summary

• Images
  ◦ Pixels are samples
  ◦ Photographs sample plenoptic function

• Colors
  ◦ Tristimulus theory of color
  ◦ Different color models for different devices, uses, etc.
  ◦ RGB model is common due to human perception
  ◦ CIE Chromaticity Diagram

• Next time
  ◦ Image processing!