Computer Graphics
Fall 2022
Overview

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

• Syllabus
  ○ What will I learn in this course?

• Imaging
  ○ Getting started …
Administrative Stuff

• Instructors
  ◦ Prof: Szymon Rusinkiewicz
  ◦ TAs: Yuting Yang, Yuanqiao Lin, Guðni Nathan Gunnarsson

• Book

• Web page
  ◦ www.cs.princeton.edu/~cos426

• Discussion
  ◦ Ed (linked from web page, or from Canvas)
COS 426: Computer Graphics
Fall 2022

Syllabus

Description
Computer graphics lies at the intersection of computer science, geometry, physics, and art. This course provides an introduction to the field, with an emphasis on practical methods and applications in image processing, modeling, rendering, and computer animation. The goal of this course is to equip students with the tools and techniques they need to build projects with significant graphical components; this includes applications for realizing artistic visions (art and architecture), user interaction (UI/UX development), entertainment products (video games, CGI, animations, and augmented reality), visualizations and academic
Coursework

• Exam (15%)  
  ◦ During class hours (in person or online, TBA): Oct 13

• Programming Assignments (60%)  
  ◦ Assignment #0: JS Paint (warmup)  
  ◦ Assignment #1: Image Processing  
  ◦ Assignment #2: Modeling  
  ◦ Assignment #3: Ray Tracer  
  ◦ Assignment #4: Rasterizer  
  ◦ Assignment #5: Animation

• Final Project (20%)  
  ◦ Interactive game (completed in groups of 3-4): due Dean’s Date

• Participation (5%)
## Programming Assignments

- **When?**  
  - Roughly every 2-3 weeks

- **How?**  
  - Javascript (precept this week)  
  - Some OpenGL (WebGL, GLSL)

- **What?**  
  - Basic feature lists  
  - Extra credit lists  
  - Art contest
Art and Simulation Contest

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

Cool Images/Videos
(Jad Bechara, CS 426, Spring 2018)

Bloopers
(Reed Tantiviramanond, CS 426, Spr15)

Characters for web banner
Collaboration Policy – See Website

• **Overview:**
  ◦ You must type your own code, but may work in pairs
  ◦ 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 …**
  ◦ Share your code digitally (e.g. post on web, email)
  ◦ Copy code directly from another student
  ◦ Leverage code acquired from other sources
Questions / Discussion

• Ed (edstem.org, log in through Canvas)
  ◦ View announcements
  ◦ Post questions to the class
  ◦ Answer other students’ questions
  ◦ Is set up for everyone enrolled as of today
  ◦ Use this instead of email to instructors
    (can send private messages)
Precepts

• When and Where
  ○ Thu 7:30 - 8:20 pm, Friend 111
  ○ Fri 10:00 - 10:50 am, Friend 111
  ○ Fri 11:00 - 11:50 am, Friend 111
  ○ Attend your own precept if you can, but if unable then attend another precept

• Topics
  ○ Usually centered around assignments
  ○ This week: 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, Fall 99)
- Rendering (David Paulk, CS426, Spr 2015)
- Modeling (Andrew Werner, Spring 2014)
- 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
Part II: Modeling

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

- Procedural modeling
  - Sweeps
  - Fractals
  - Grammars
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
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)

Minecraft
(Xbox One Edition)
Applications

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

Gear Shaft Design
(Intergraph Corporation)

Boeing 777 Airplane
(Boeing Corporation)

Los Angeles Airport
(Bill Jepson, UCLA)
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)

Visible Human
(National Library of Medicine)
Applications

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

Early 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

Virtual Stores
(Matterport)
Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- 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 … (Yes, this WILL be on the exam!)
What is an Image?
What is an Image?

• An image is a 2D rectilinear array of pixels
What is a Pixel?

Pixel

Digital image
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?

$l(x,y)$?
Plenoptic Function

• Each pixel of a photographic image is a function of \textit{radiance} arriving at a sensor
Plenoptic Function

- 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 \((\lambda)\)
Photographic Images

* An idealized photographic image contains a 2D array of samples of the 7D plenoptic function:
  - at a particular camera viewpoint,
  - for a 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
How Are Digital Images Stored?

- Frame Buffer
- User
- RAM
- Disk
- Network

Based on Figure 1.2 from FvDFH
Frame Buffer Limits: Resolution

- **Spatial resolution**
  - Image has only “Width” x “Height” pixels

- **Intensity resolution**
  - Each pixel has only “Depth” bits for colors / intensities

- **Temporal resolution**
  - Screen refreshes images at only “Rate” Hz

<table>
<thead>
<tr>
<th>Typical Resolutions</th>
<th>Width x Height</th>
<th>Depth</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cheap laptop</td>
<td>1366 x 768</td>
<td>24</td>
<td>60 Hz</td>
</tr>
<tr>
<td>High-end laptop</td>
<td>2560 x 1600</td>
<td>24</td>
<td>60 Hz</td>
</tr>
<tr>
<td>TV</td>
<td>1920 x 1080</td>
<td>16-ish</td>
<td>60 (interleaved)</td>
</tr>
<tr>
<td>Film</td>
<td>3000 x 2000</td>
<td>36</td>
<td>24 Hz</td>
</tr>
<tr>
<td>Printer</td>
<td>5100 x 6600</td>
<td>1-4</td>
<td>-</td>
</tr>
</tbody>
</table>
What Frequencies / Wavelengths?

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

Corresponding frequency is: \( \frac{c}{\lambda} \)
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
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
Color

• How do we represent a color in a computer?
  ◦ Must store a finite amount of data to represent magnitudes for infinite number of frequencies

Figures 15.3-4 from H&B
# Color Frame Buffer

## Table

<table>
<thead>
<tr>
<th>Blue</th>
<th>Green</th>
<th>Red</th>
</tr>
</thead>
<tbody>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
</tr>
<tr>
<td>255</td>
<td>150</td>
<td>75</td>
</tr>
</tbody>
</table>

- **Blue channel**
- **Green channel**
- **Red channel**
Why red, green, and blue (RGB)?
Modern Understanding of Color

- Two types of receptors: rods and **cones**

  - Rods and cones
  - Cones in **fovea** (central part of retina)
Human Color Perception

- Spectral-response functions of each of the three types of cones on the human retina
- Tristimulus theory of color
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>

Plate II.3 from FvDFH
RGB Color Cube

- Grayscale
  - Cyan (0, 1, 1)
  - Black (0, 0, 0)
  - Blue (0, 0, 1)
  - Magenta (1, 0, 1)

- Green (0, 1, 0)
- Yellow (1, 1, 0)
- Red (1, 0, 0)

H&B
RGB Spectral Colors

- Amounts of RGB primaries needed to display spectral colors

Figure 15.5 from H&B
XYZ Color Model (CIE)

- Linear transform of RGB
  - All colors can be composed of non-negative amounts of XYZ

\[
\begin{pmatrix}
X \\
Y \\
Z
\end{pmatrix} =
\begin{pmatrix}
0.412452 & 0.357580 & 0.180423 \\
0.212671 & 0.715160 & 0.072169 \\
0.019334 & 0.119193 & 0.950227
\end{pmatrix}
\begin{pmatrix}
R \\
G \\
B
\end{pmatrix}
\]

Figure 15.6 from H&B
CIE Chromaticity Diagram

- Normalized amounts of X and Y for colors in visible spectrum

chromaticity diagram. Monochromatic colors are located on the perimeter. Color saturation decreases towards the center of the diagram. White light is located in the center. Also shown are the regions of distinct colors. The equal-energy point is located at the center and has the coordinates \((x, y) = (1/3, 1/3)\).
RGB Color Gamut

- Color gamut (range of colors) of a typical RGB computer display

Figure 15.13 from H&B
## Other Color Models

- CMY
- HSV
- CIELAB
- Others

Different color models are useful for different purposes
CMY Color Model

- Useful for printers because colors are *subtractive*
- Add black ink – CMYK

<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
HSV Color Model

• Intended for ease of color picking
CIELAB Color Model

- Non-linear transform of XYZ based on human perception

\[
L^* = 116f \left( \frac{Y}{Y_n} \right) - 16
\]
\[
a^* = 500 \left( f \left( \frac{X}{X_n} \right) - f \left( \frac{Y}{Y_n} \right) \right)
\]
\[
b^* = 200 \left( f \left( \frac{Y}{Y_n} \right) - f \left( \frac{Z}{Z_n} \right) \right)
\]

\[
f(t) = \begin{cases} \frac{3\sqrt{t}}{3\delta^2} + \frac{4}{29} & \text{if } t > \delta^3 \\ \frac{6}{29} & \text{otherwise} \end{cases}
\]

- \(X_n = 95.047\),
- \(Y_n = 100.000\),
- \(Z_n = 108.883\)

Useful for measuring perceptual differences between colors
Frame Buffer Display

• Video display devices
  ◦ Liquid Crystal Display (LCD)
  ◦ 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
- Pixels with finite area (rectangles)
- Colors are interleaved
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!