CS 5600

Introduction to Computer Graphics
http://www.eng.utah.edu/~cs5600/
Prof: Chuck Hansen

Goal: have fun and learn graphics!

CS 5600

• Class mailing list:
cs5600@list.eng.utah.edu
• Sign up:
https://sympa.eng.utah.edu/sympa/info/cs5600
• Text: OpenGL Programming Guide
  "Red Book"
supplemental reading material
  » DDA – Line Drawing
  » Ray Tracing

CS 5600

• Grading:
  70% homework
  25% exams
  5% class participation (ask questions, respond to questions)
• Cheating: DON'T share code! DON'T grab code off the web!
• Late penalty: -20%/day but 4 one-day grace periods.

Prereqs:

• Normalized Vector?
• Matrix multiply?
• Vector multiply?
• Dot-product? (what is it?)

Prereqs:

• Normalized Vector?
• Matrix multiply?
• Vector multiply?
• Dot-product? (what is it?)

\[ \mathbf{a} \cdot \mathbf{b} = |\mathbf{a}| |\mathbf{b}| \cos \theta \]
**Prereqs:**

- Cross Product?
  - Properties?

**Cross Product**

\[ V_1 = [3, -1, 0] \quad V_2 = [4, -3, 1] \]

\[
\begin{array}{ccc}
  x & y & z \\
  3 & -1 & 0 \\
  4 & -3 & 1 \\
\end{array}
\]

\[
\text{Det: } -1, 0, 3 \\
\]

**Cross Product**

\[ V_1 = [3, -1, 0] \quad V_2 = [4, -3, 1] \]

\[
\begin{array}{ccc}
  x & y & z \\
  3 & -1 & 0 \\
  4 & -3 & 1 \\
\end{array}
\]

\[
\text{Det: } -1, 0, 3 \\
\]

**Cross Product**

\[ V_1 = [3, -1, 0] \quad V_2 = [4, -3, 1] \]

\[
\begin{array}{ccc}
  x & y & z \\
  3 & -1 & 0 \\
  4 & -3 & 1 \\
\end{array}
\]

\[
\text{Det: } -1, 0, 3 \\
\]

**Cross Product**

\[ V_1 = [3, -1, 0] \quad V_2 = [4, -3, 1] \]

\[
\begin{array}{ccc}
  x & y & z \\
  3 & -1 & 0 \\
  4 & -3 & 1 \\
\end{array}
\]

\[
\text{Det: } -1, 0, 3 \\
\]
Cross Product

\[ \text{V1} = [3, -1, 0] \quad \text{V2} = [4, -3, 1] \]

\[
\begin{array}{ccc}
3 & -1 & 0 \\
4 & -3 & 1 \\
\end{array}
\]

\[ \text{Det: } -1, -3, -9 \cdot (-4) \]

Cross Product \textbf{Sarrus' scheme}

\[ \text{V1} = [3, -1, 0] \quad \text{V2} = [4, -3, 1] \]

\[
\begin{array}{ccc}
3 & -1 & 0 \\
4 & -3 & 1 \\
\end{array}
\]

\[ \text{Det: } -1, -3, -9 \cdot (-4) \]
Cross Product **Sarrus' scheme**

\[
\begin{array}{ccc}
V1 = [3, -1, 0] & V2 = [4, -3, 1] \\
\end{array}
\]

\[
\begin{array}{ccc}
x & y & z \\
3 & -1 & 0 \\
4 & -3 & 1 \\
\end{array}
\]

Det: -1, -3, -5

---

**Line Equations**

- Explicit form: \( y = mx + b \)
- Implicit form: \( f(x,y) = Ax + By + C = 0 \)
- Parametric form: \( P(x,y) = P_0 + t \vec{D} \)

---

**Line Equations**

Show: \( y = mx + b \) equivalent to \( P(x,y) = P_0 + t \vec{D} \)

For \( P_1P_2 \) (assume \( b=0 \)):

\[
\begin{align*}
y - y_1 &= ((y_2 - y_1)/(x_2 - x_1)) (x - x_1) \\
x &= ((x_2 - x_1)/(y_2 - y_1)) (y - y_1) + x_1
\end{align*}
\]

---

**Color**

- Color is complicated!
  - Highly nonlinear
  - No single model to explain all
- Many simplistic models, explanations
- Many myths
- Much new knowledge

---

CS5600 Computer Graphics
Modified from Rich Riesenfeld

Spring 2013
Wavelength Spectrum

- Seen in physics, physical phenomena (rainbows, prisms, etc)
- 1 Dimensional color space

Color Space

- “Navigating,” moving around in a color space, is tricky
- Many color representations (spaces)
- Can you get to a nearby color?
- Can you predictably adjust a color?

R-G-B Color Space

- Convenient colors (screen phosphors)
- Decent coverage of the human color
- Not a particularly good basis for human interaction
  - Non-intuitive
  - Non-orthogonal (perceptually)
Color Cube: \((r,g,b)\) is RHS

Complementary Colors Add to Gray

Complementary Colors

Additive Primary Colors

Additive Primary Colors
Additive Primary Colors

- Red: (1,0,0)
- Magenta: (1,0,1)
- Blue: (0,0,1)

Subtractive Primary Colors

- Yellow: (1,1,0)
- Magenta: (1,0,1)
- Cyan: (0,1,1)
- Black: (0,0,0)
- Green: (0,1,0)
- Blue: (0,0,1)

Computer Graphics CS5600
**Color Space**

\((H, S, V)\)  

- Introduced by Albet Munsell, late 1800s  
  - He was an artist and scientist  
- Hue: Color  
- Saturation/Chroma: Strength of a color  
  - Neutral gray has 0 saturation  
- Brightness/Value: Intensity of light emanating from image

**HSV/HSL**

**HSI/HSV**

- Value/Luminance – total amount of energy  
- Saturation – degree to which color is one wavelength  
- Hue – dominant wavelength

**HSV**

- \(\text{Max} = \max(R, G, B)\)  
- \(\text{Min} = \min(R, G, B)\)  
- \(S = (\text{Max} – \text{Min})/\text{Max}\)  
- If \(R==\text{Max} \rightarrow h = (G-B)/(\text{Max-Min})\)  
- If \(G==\text{Max} \rightarrow h = 2+(B-R)/(\text{Max-Min})\)  
- If \(B==\text{Max} \rightarrow h = 4 + (R-G)/(\text{Max-Min})\)  
- If \(h<0 \rightarrow H = h/6 + 1\)  
- If \(h>0 \rightarrow H = h/6\)

**HSV User Interaction**
HLS

- $S = \sqrt{((R-G)^2 + (R-B)^2 + (G-B)^2)/2}$
- $I = (R + G + B)/3$
- $H = (a - \arctan((R-I)b/(G-B)))/(2\pi)$ --- angle
  - $a = -\pi/2$ if $G>B$
  - $-3\pi/2$ if $G<B$
  - $H = 1$ if $G=B$
- $a = \sqrt{3}$

(Hue, Saturation, Value/Intensity)

$(H, S, V)$ Color Space

The hue of an object may be blue, but the terms light and dark distinguish the brightness of one object from another.

HSV Color Space (Cone)

OpenGL Color

```c
void glColor3f (GLfloat red, GLfloat green, GLfloat blue);
void glColor4f (GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);
```

Color is OpenGL State (once set, it doesn't change)

Typical usage:

```c
gBegin(GL_POINTS)
gColor3f ( 1.0, 0.0, 0.0);
gVertex2i ( 64, 64);
gVertex2i (128, 128);
gColor3f ( 0.0, 1.0, 0.0);
gVertex2i ( 256, 256);
```
**Tristimulus Color Theory**

- Any color can be matched by a mixture of three fixed base colors (primaries)
- Eye has three kinds of color receptors called cones
- Eye also has rods (low light receptors)

**Color Receptors in Eye**

- Red, Green, Blue
- Long, Medium, Short

**Fraction of light absorbed by each type of cone**

<table>
<thead>
<tr>
<th>Wavelength (nm)</th>
<th>Relative sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>400</td>
<td>0.00</td>
</tr>
<tr>
<td>400</td>
<td>0.02</td>
</tr>
<tr>
<td>400</td>
<td>0.04</td>
</tr>
<tr>
<td>400</td>
<td>0.06</td>
</tr>
<tr>
<td>400</td>
<td>0.08</td>
</tr>
<tr>
<td>400</td>
<td>0.10</td>
</tr>
<tr>
<td>400</td>
<td>0.12</td>
</tr>
<tr>
<td>400</td>
<td>0.14</td>
</tr>
<tr>
<td>400</td>
<td>0.16</td>
</tr>
<tr>
<td>400</td>
<td>0.18</td>
</tr>
<tr>
<td>400</td>
<td>0.20</td>
</tr>
</tbody>
</table>

**CIE* Color Space**

\((X, Y, Z)\) represents an imaginary basis that does not correspond to what we see.

Define the normalized coordinates:

\[
x = \frac{X}{X + Y + Z} \\
y = \frac{Y}{X + Y + Z} \\
z = \frac{Z}{X + Y + Z}
\]

* Commission Internationale de l’Éclairage

**CIE Color Space of Visible Colors**

\[
x = \frac{X}{X + Y + Z} \\
y = \frac{Y}{X + Y + Z} \\
z = \frac{Z}{X + Y + Z}
\]

\(x + y + z = 1\)

The projection of the plane of the triangle onto the \((X,Y)\) plane forms the chromaticity diagram that follows.

**Color Gamuts: CIE Color Chart**
The additive colors $C_1$ and $C_2$ combine to form $C_3$ on the line connecting $C_1$ and $C_2$.

The Color Gamuts of different displays and printers are not likely to match. Printers usually have smaller gamuts.
**CIE L*a*b* Color Space**

Equal distances represent approximately equal color difference.

**Trichromatic Theory**

**Shortcomings**

- Color blindness
  - R-G, B-Y, All
- Yellow seems primary
- Color constancy

**Color Blindness**

- Normal
- Protan (L-cone)
- Deutan (M-cone)
- Tritan (S-cone)

**Mondrian Color Patches**

- Colors look different depending on their neighbors
- Adjacency/black lines
- Color edges are critical to color perception
- Can determine color in non-white lighting conditions
Opponent Color Theory

- Humans encode colors by differences
- E.g. R-G, and B-Y Differences
  - Color blindness

\[ \begin{align*}
\text{Long (R)} & \quad \rightarrow \quad \text{Short (B)} \\
\text{Medium (G)} & \quad \leftrightarrow \quad \text{Achromatic}
\end{align*} \]

Perceptual Distortions

- Color-deficiency
- Interactions between color components
  - brightness - hue (Bezold-Brucke Phenomenon)
  - saturation - brightness (Helmholtz-Kohlrausch effect)
- Simultaneous contrast
  - brightness
  - hue
- Small field achrominance
- Effects of color on perceived size

Bezold-Brucke Phenomenon

- Hurvich ‘81, pg. 73.

Helmholtz-Kohlrausch effect

Simultaneous Contrast
Color Applets

www.cs.brown.edu/exploratories/freeSoftware/repository/
edu/brown/cs/exploratories/applets/combinedColorMixing/
/combined_color_mixing_java_browser.html

End Color