Color

Color Images

- Goal of OpenGL is to draw color pictures on the computer screen
- Window is a rectangular array of pixels
- How to determine the final color of every pixel

Color Perception

- Our eyes see a mixture of photons of different wavelengths as a color

- Visible spectrum:
  - Violet (390 nm)   to   Red (720 nm)

- Cone cells in the retina are excited by photons
  - Three types of cone cells respond best to three different wavelengths

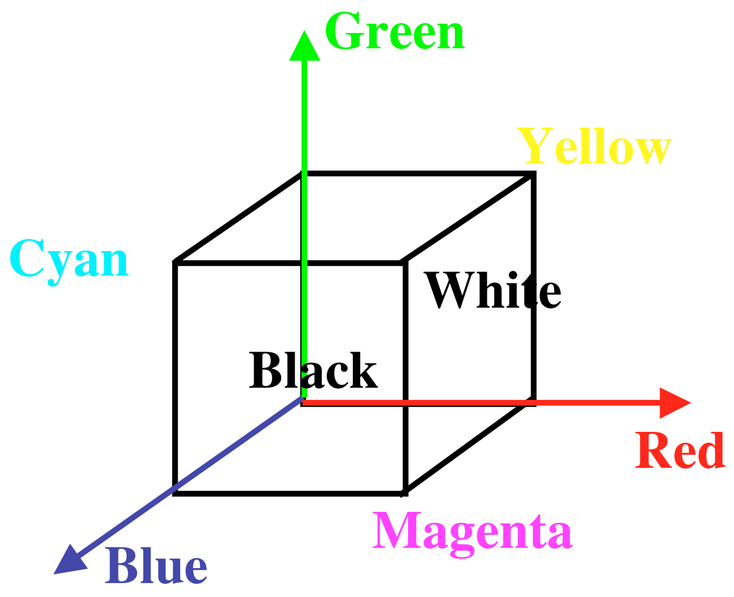
    Red   Green   Blue
  - Other representations: HLS, HSV, CMYK

Computer Color

- Follows RGB analogy
  - ➢ Each pixel on the screen emits right amounts of the R, G and B light to appropriately stimulate different types of cones in the eye to display a particular color
- Color cube
  - ➢ Combining the R, G and B light results in different colors

    Red and Blue make megenta

    Red and Green make yellow



- Color buffer
  - ➢ Memory for the color information for pixels
  - ➢ Size of buffer is expressed in bits; an n bit buffer could 2^n possible colors for each pixel

Color Display Mode

- RGBA mode
  - ➢ Red, green, blue and alpha commponets
  - ➢ The R, G and B values can range from 0.0 (none) to 1.0 (full intensity)
  - ➢ A 24-bitplane system provides 8 bits each to R, G and B

    The values are clamped to (0.0,1.0)

    Each color component range:

    $0/2^n = 0.0,\ 1/2^n,\ 2/2^n,\ \ldots\ldots\ldots,\ 2^n/2^n = 1.0$

    thus displaying up to 256x256x256 ~ 16.77 million distinct colors

- Color-Index mode
  - ➢ Use color map or table
  - ➢ Stores a single number (index) for each pixel to indicate an entry in a lookup table or color map

Specifying Color

- RGBA mode is preferable over color-index mode

- Each object is drawn using the current color
  - Lighting can change the actual color that will ultimately be shown

- void **glColor4**{b s i f d ub us ui}(*TYPE r*, *TYPE g*, *TYPE b*, *TYPE a*);

  void **glColor4**{b s i f d ub us ui}**v**(const, *TYPE *v*);
  - Sets the current red, green, blue, and alpha values
  - Default value of alpha value (a) is 1.0
  - Several acceptable data types for parameters

glColor3f(1.0,0.0,0.0)	RED
glColor3f(1.0,1.0,0.0)	YELLOW
glColor3f(1.0,1.0,1.0)	WHITE
glColor3f(0.0,0.0,0.0)	BLACK

Shading Model

- void **glShadeModel**(GLenum *mode*)
  - ➢ Sets the shading model with argument mode taking GL_FLAT or GL_SMOOTH
- Flat shading
  - ➢ The color of one particular vertex defines the color of entire primitive
- Smooth (Gouraud) shading
  - ➢ The color at each vertex is treated individually, and the colors for the interior of the polygon are interpolated between the vertex colors
  - ➢ Neighboring pixels have slightly different color