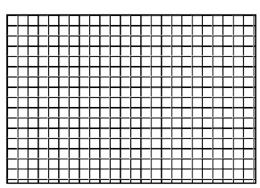
Introduction to Graphics

Basic Definitions

- Raster: A rectangular array of points or dots.
- Pixel (Pel): One dot or picture element of the raster



- Raster-based display hardware common
- Imaging model (API for display) may be different

Image Models: Stroke Model

- Early displays were stroke based
- Everything defined using lines

Image Models: Pixel Model

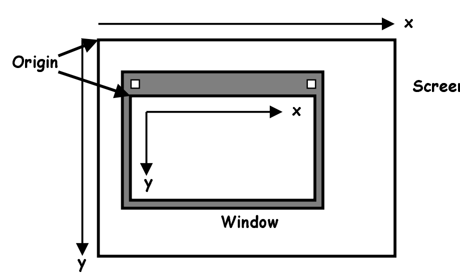
- Close match to hardware
- 4 kinds: bitmap, grey, full clr, index clr
- *Discrete* rep. Causes aliasing

Image Models: Outline Model

- Use strokes to outline regions
  - Compact, device ind. representation

Coordinate Systems: Device Coordinates

- Screen/Window coords start in upper left corner



Coordinate Systems: Physical Coordinates

- Pixels are different size on every display
  - Expressed as dots per inch (dpi)
- Fonts described using “points”
  - 72 points per inch
- Performance issues

Coordinate Systems: Model Coordinates

- Often coordinates are expressed in units independent of display
- Adds another level of scaling

MP * DrawScale * PhysToPixel + WO = OP

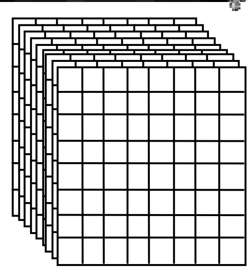
Coordinate Systems: Interactive Coordinates

- Inverse: map from window to model

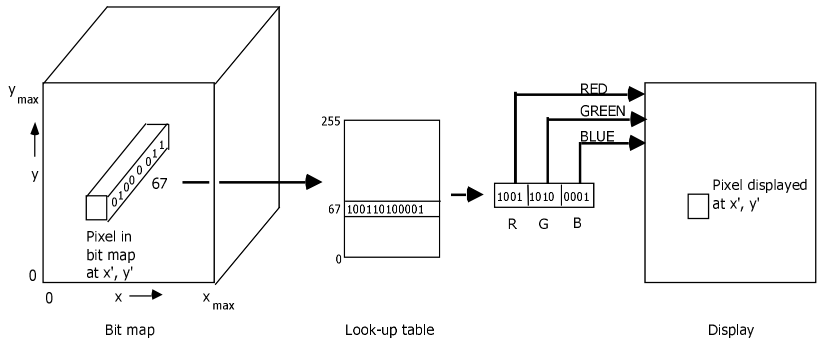
(IP - WO)/(DrawScale * PhysToPixel) = MP

Hardware: Frame Buffers

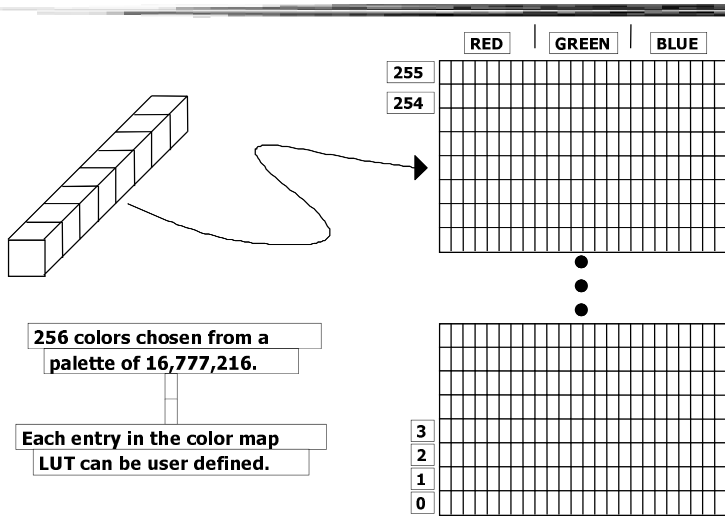
- 2D array
  - each (x,y) location = a pixel
- *Bit Planes, Bit Depth*
  - number of bits in a pixel



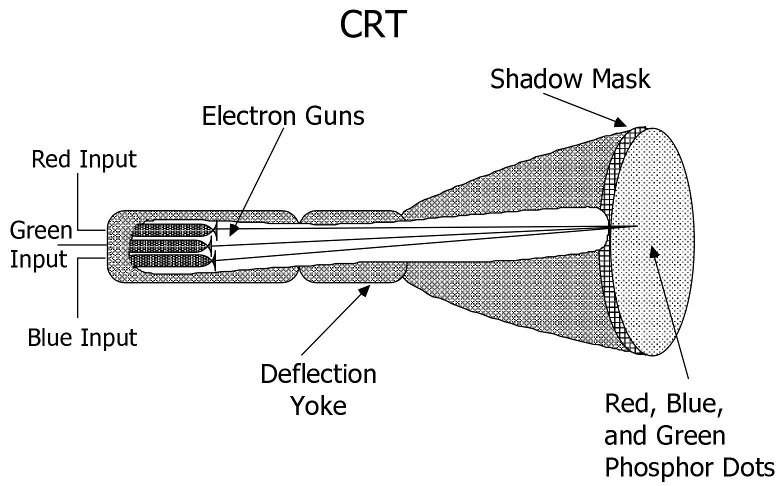
Color Map Look-Up Tables



Pseudo Color: 2^8 x 24 Color Map LUT

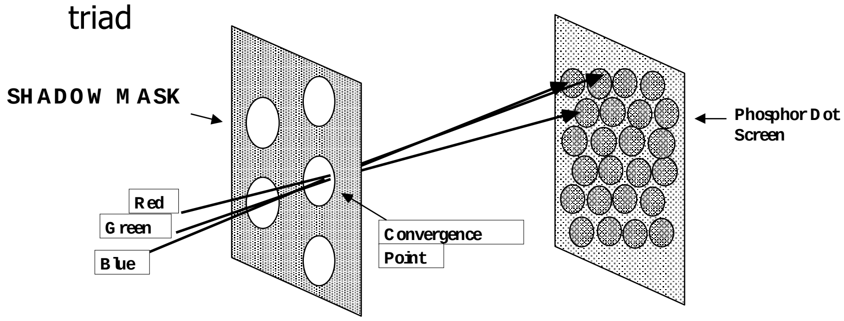


CRT Monitor



Shadow Mask

- Phosphors arranged in triads
- Each triad has one R/G/B phosphor dot
- Typically 2.3 to 2.5 triads per pixel
- Shadow mask has one small hole for each phosphor triad



Abstract Canvas Representation of FrameBuffer

- Virtual Screen
- Can hide (or not) display properties
  - Windows/X vs. Mac/NeXT/NeWS

Drawing

- Paths
  - Lines, circles, arcs, ellipses, splines, closed shapes
- Text
  - Specify font, style, size
  - Get font information

Clipping

- Rectangular
- Rectilinear
- Rectilinear with holes

Set operations

- Operations
- Closure

Color & Graphics

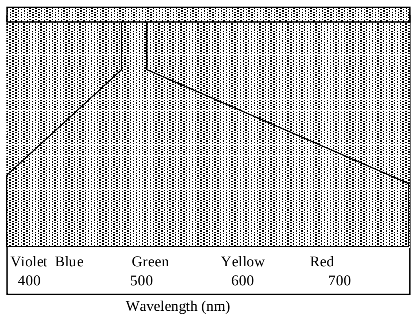
- The complete display system is:
  - Model
  - Frame Buffer
  - Screen
  - Eye
  - Brain

Color & Vision

- We'll talk about:
  - Light
  - Visions
  - Psychophysics, Colorimetry
  - Color
    - Perceptually based models
    - Hardware models

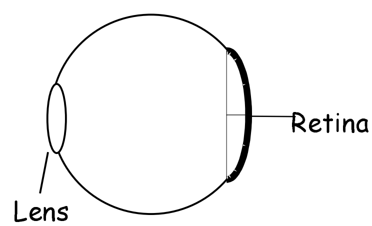
Light

- Vision = perception of electromagnetic energy (EM radiation)
- Very small portion of EM spectrum perceptible:



Vision: The Eye

- A dynamic, biological camera!
  - a lens
  - a focal length
  - an equivalent of film



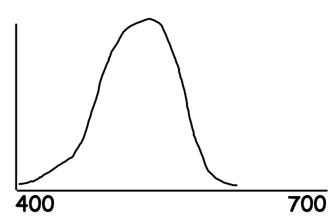
- The lens must focus directly on the retina for perfect vision

Vision: The Retina

- The eye's "film"
- Covered with cells sensitive to light
  - turn light into electrochemical impulses
- Two types of cells
  - rods
  - cones

Vision: Rods

- Sensitive to most wavelengths (brightness)
- About 120 million in eye
- Most outside of fovea (center of retina)
- Used for low light vision
- Absorption function:

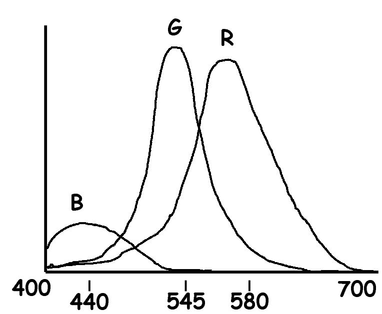


Vision: Cones

- Three kinds
  - R sensitive to long wavelengths
  - G to middle
  - B to short
- About 8 million in eye
- Highly concentrated in fovea
  - B cones more evenly distributed than others
- Used for high detail color vision

Vision: Cones

- The absorption functions of the cones are:

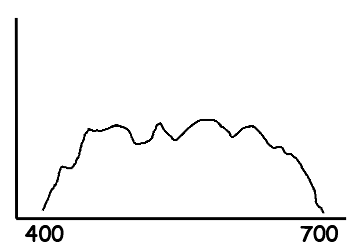


UI Issues

- Can’t focus on R/B simultaneously
  - Refractive index of lens
- Need contrast to read
  - Luminance or color
    - Color blindness, dist. of rods => luminance
- Color bleed

Psychophysics

- Spectral Energy Distribution
  - measure intensity of light at unit wavelength intervals of electromagnetic spectrum from ~400 nm to ~700 nm
- To mix colors
  - mix power distributions!



Color Mixing: Additive

- Luminous objects emit s.e.d.
- Linearly add s.e.d.'s
- Primaries: red green blue
- Complements: cyan magenta yellow

- e.g. Monitors, lights

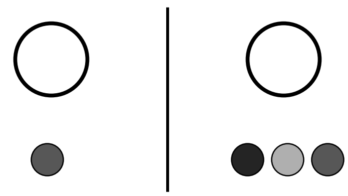
Color Mixing: Subtractive

- Reflective objects absorb (or filter) light
- Can't subtract s.e.d.'s
  - Filters: transmission functions
  - Pigment: suspension, scattering of light
- Primaries: red yellow blue
- Complements: green violet orange

- E.g., ink, film, paint, dye

Colorimetry

- Based on matching colors using additive color mixing



- Tristimulous Values
- Metamers
  - Different s.e.d.'s that appear the same
  - Same tristimulous values

CIE 1931 Imaginary Primaries

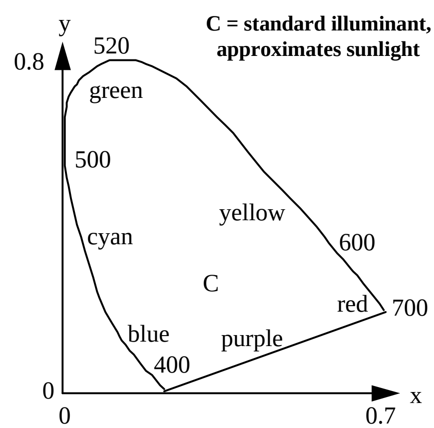
- Defines three new primary "colors"
  - X, Y and Z
  - Color match functions all positive valued
  - Y's fcn corresponds to luminance-efficiency function
- To define a color
  - weights x,y,z for the X,Y,Z primaries (e.g. color = xX + yY + zZ)

CIE 1931 Chromaticity

- X, Y and Z form a three dimensional color volume
  - Y is luminance, others aren't intuitive
- Factor luminance by normalizing x+y+z = 1
- Gives *chromaticity* values:
  - x' = x/(x+y+z)
  - y' = y/(x+y+z)
  - z' = 1 - x' - y'

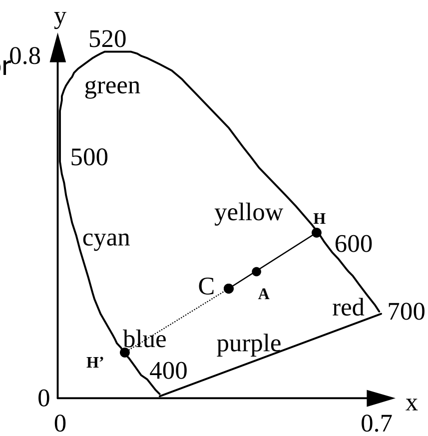
CIE 1931 Chromaticity Diagram

- Chromaticity diagram
  - Plot of x' vs. y'
- Additive color mixing
  - linear interpolation
- Color gamuts
  - range of possible colors for a device
  - convex hull of primary colors



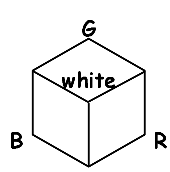
CIE 1931 Chromaticity Diagram

- Dominant Wavelength/Hue:
  - inscribe line from C through color (A) to edge of diagram (H)
- Saturation
  - distance C-A / distance C-H
- Complements
  - inscribe line through C to the edge of the diagram (H’)
- What if edge is bottom?



Hardware Models: RGB (Additive Color)

- (red, green, blue)
- Parameters vary between 0 and 1

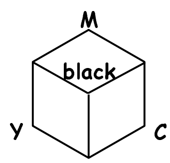


Hard to achieve intuitive effects:

- Hue is defined by the one or two largest parameters
- Saturation controlled by varying the collective minumum value of R, G and B
- Luminance controlled by varying magnitudes while keeping ratios constant

Hardware Models: CMY, CMYK (Subtractive Color)

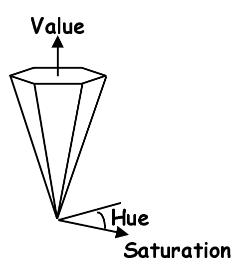
- (cyan, magenta, yellow, +blacK)
- All parameters vary between 0 and 1



- K = min (C,M,Y)
- subtract K from each

Intuitive Hardware Models: HSV

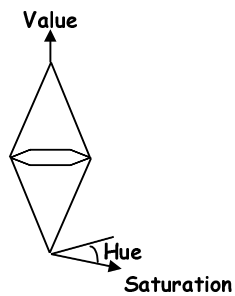
- (hue, saturation, value)
  - value roughly luminance
  - hue: (0...360), saturation/value: (0...1)



- Simple xform of RGB
- What do hexagonal and triangle cross sections look like?

Intuitive Hardware Models: HLS

- (hue, lightness, saturation)
  - lightness roughly luminance
  - hue: (0...360), saturation/value: (0...1)



- saturated colors at l=0.5
- *tints* above, *shades* below
- What do hexagonal and triangle cross sections look like?

Problem: Value/Lightness NOT Luminance

- Fully saturated colors (same v/l) have far different Y values in XYZ (Sun 17" monitor, 1991):

Colour	RGB	XYZ	Chromaticity
White	1 1 1	0.951 1.000 1.088	0.313 0.329
Red	1 0 0	0.589 0.290 0.000	0.670 0.330
Green	0 1 0	0.179 0.605 0.068	0.210 0.710
Blue	0 0 1	0.183 0.105 1.020	0.140 0.080
Cyan	0 1 1	0.362 0.710 1.088	0.168 0.329
Magenta	1 0 1	0.772 0.395 1.020	0.363 0.181
Yellow	1 1 0	0.768 0.895 0.068	0.444 0.517

Problem: None of these models are perceptually uniform

- Perceived distance between two colors not proportional to linear distance
- Uniform Color Spaces
  - Non-linear deformations
  - OSA Uniform Color Space (limited range)
  - CIELUV
  - CIELAB

Issue: Device-independent color

- Must use CIEXYZ
  - ie. Apple Colorsync
- RGB = (0.3,0.2,0.55) tells you what computer generates, not what the monitor will display!
  - Depends on phosphors, room lighting, monitor adjustment
- Moving between devices (and media)
  - Go through XYZ
  - Must know properties of devices