IMAGES AND COLOR

N. C. State University

CSC557 ♦ Multimedia Computing and Networking

Fall 2001

Lecture # 10

IMAGES AND COLOR

N. C. State University

CSC557 ♦ Multimedia Computing and Networking

Fall 2001

Lecture # 10

Questions / Problems / Announcements

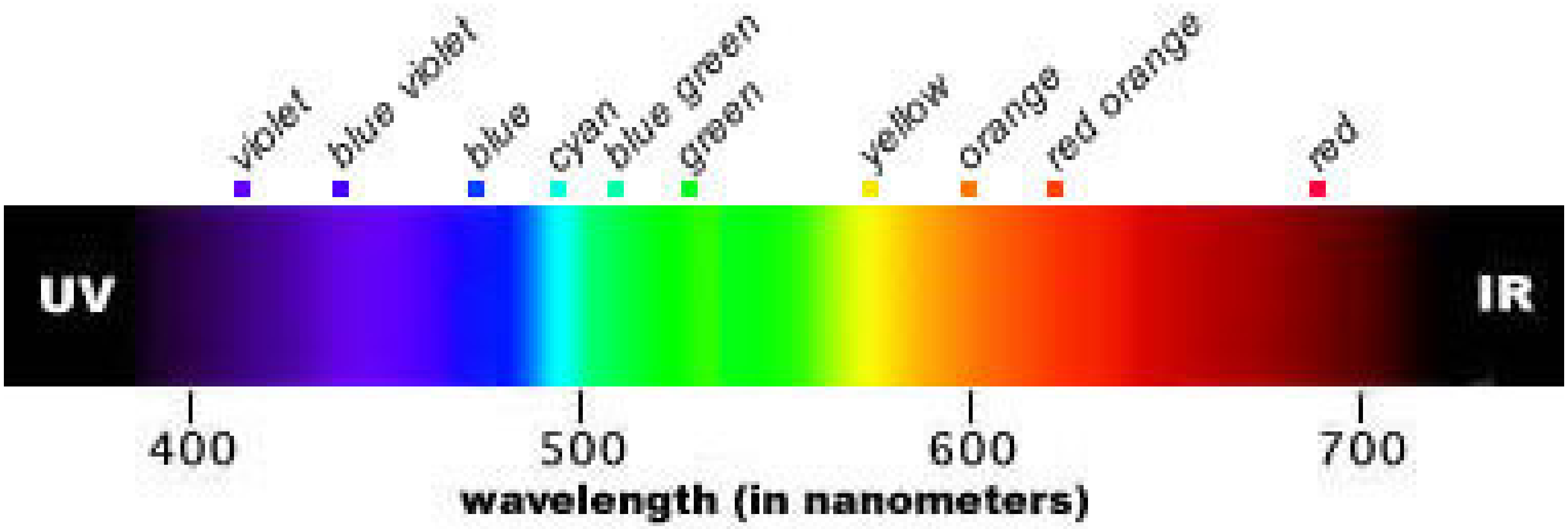
- ???

Converting Light to/from Digital Images

- Devices for converting analog-to-digital
  - Digital cameras
  - Scanners
- Devices for converting digital-to-analog
  - Monitors
  - Projectors
- "Samples" of light = pixels of an image
  - Sampling rate / spatial resolution → how many pixels are there in the image?
- Quantization of light = bits / pixel
  - Color resolution → how many colors can be represented?

Physics of Light

- Spectrum of light
  - which frequencies are present
  - Energy, or intensity, of each frequency

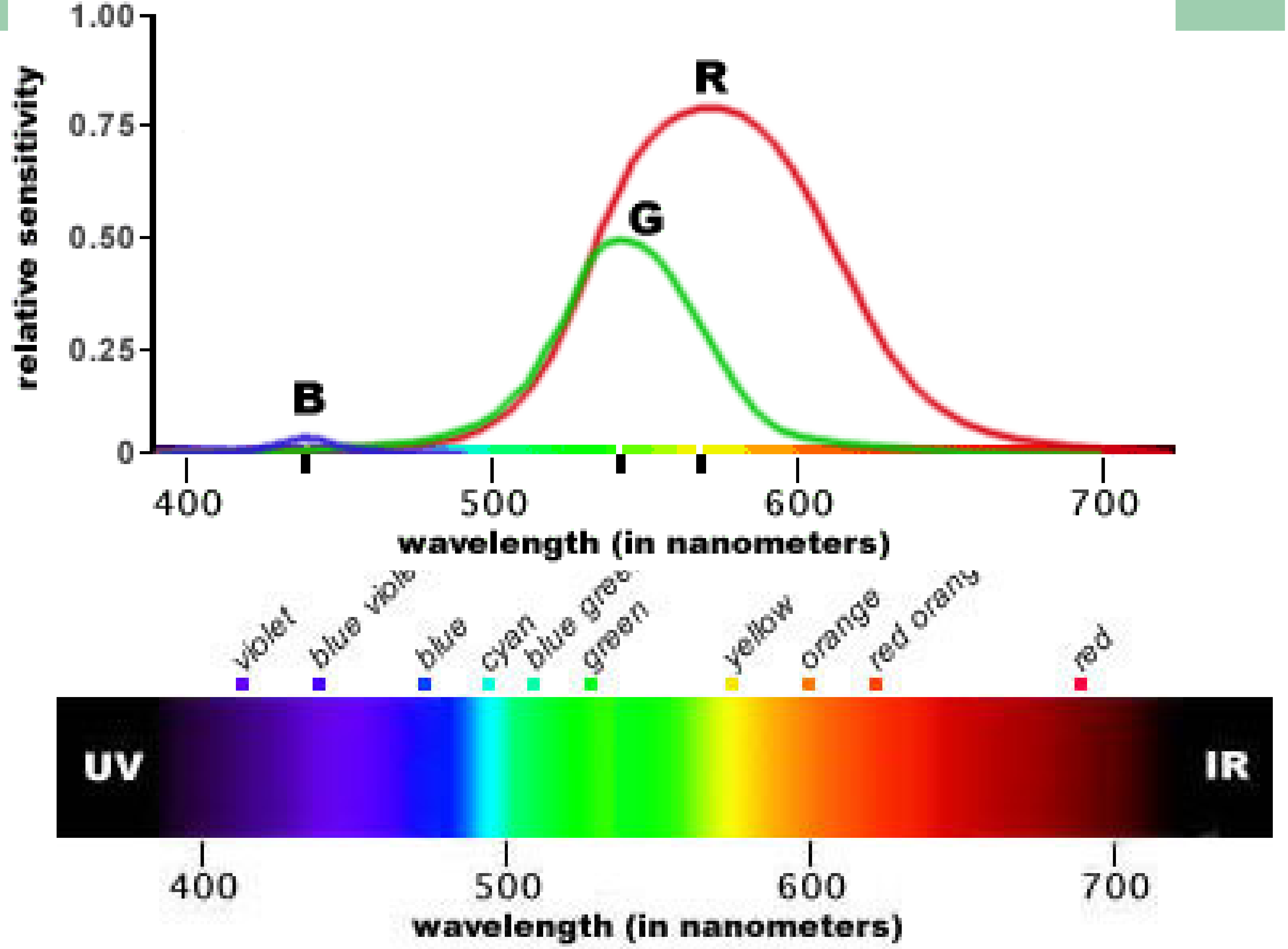


Source:
http://www.handprint.com/HP/WCL/color1.html

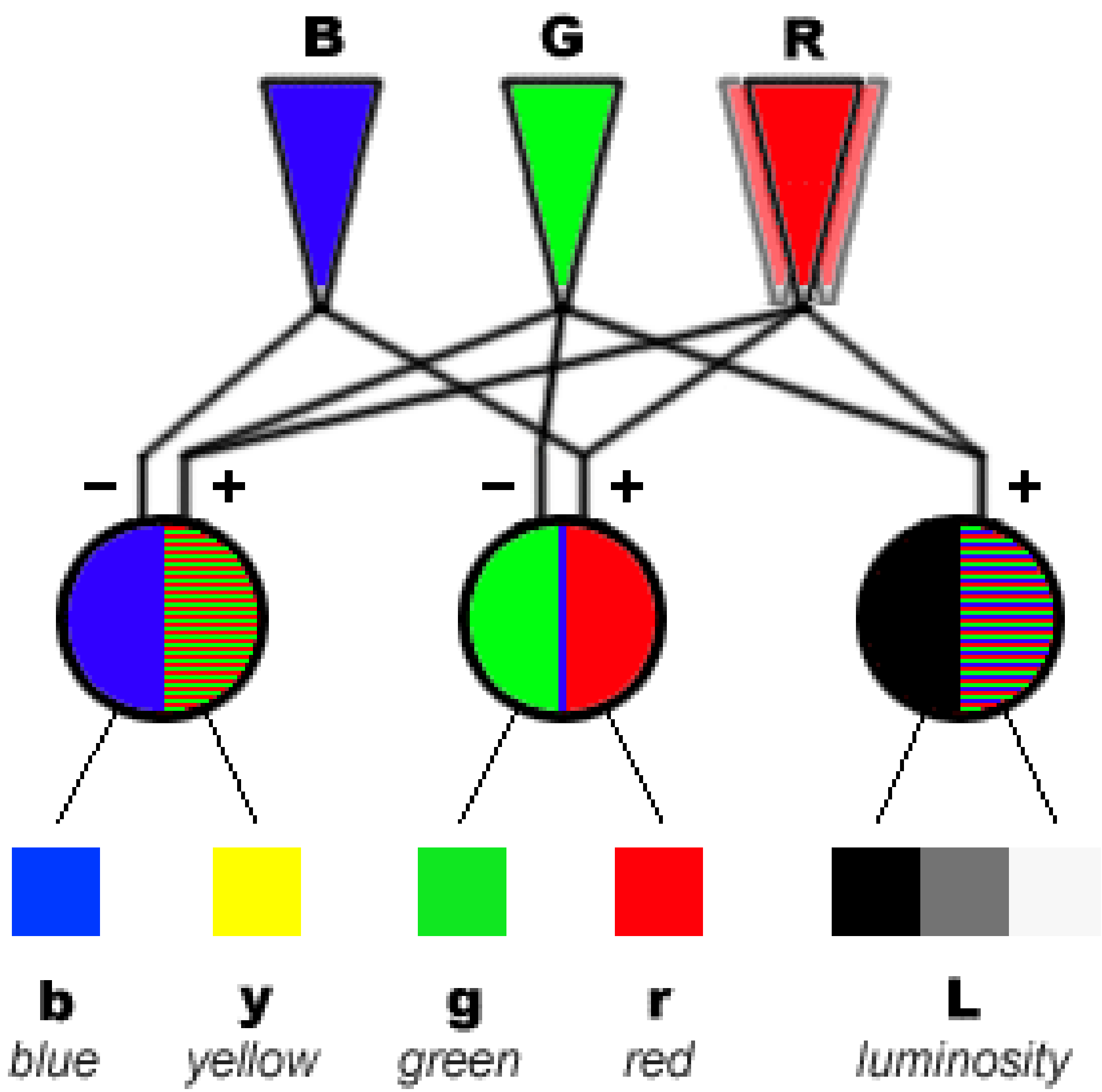
Perception of Light: Color

- *Hue* is determined by the dominant frequency
  - "What color is it?"
  - black, gray, and white have no dominant frequency, thus no hue
- *Brightness, or intensity, or luminance*, is related to total energy from all frequencies present
  - "Is the color bright or dark?"
  - relationship is roughly logarithmic
  - We can see (and tolerate) a range of brightness that spans at least 9 orders of magnitude
- *Saturation (or purity)* is determined by the ratio of (dominant_frequency_energy) / (total_energy)
  - "Is the color deep/intense, or is it pale or washed out?"
  - Stronger dominant frequency or less other frequencies → more saturated color

Sensitivity of Cones (Receptors) in Human Eye



Opponent Model of Color Vision



The Opponent Model (cont'd)

1. The combined stimulation to the R and G cones is interpreted as the brightness or lightness of a color. Luminosity is the dominant visual information recorded by the eye.

2. The relative proportion of stimulation received by the R cones in contrast to the G cones creates the perception of red or green. (The B cones contribute slightly to clarify warm color saturation.)

3. If the R and G cones are stimulated approximately equally (and much more than the B cones), we see the color yellow.

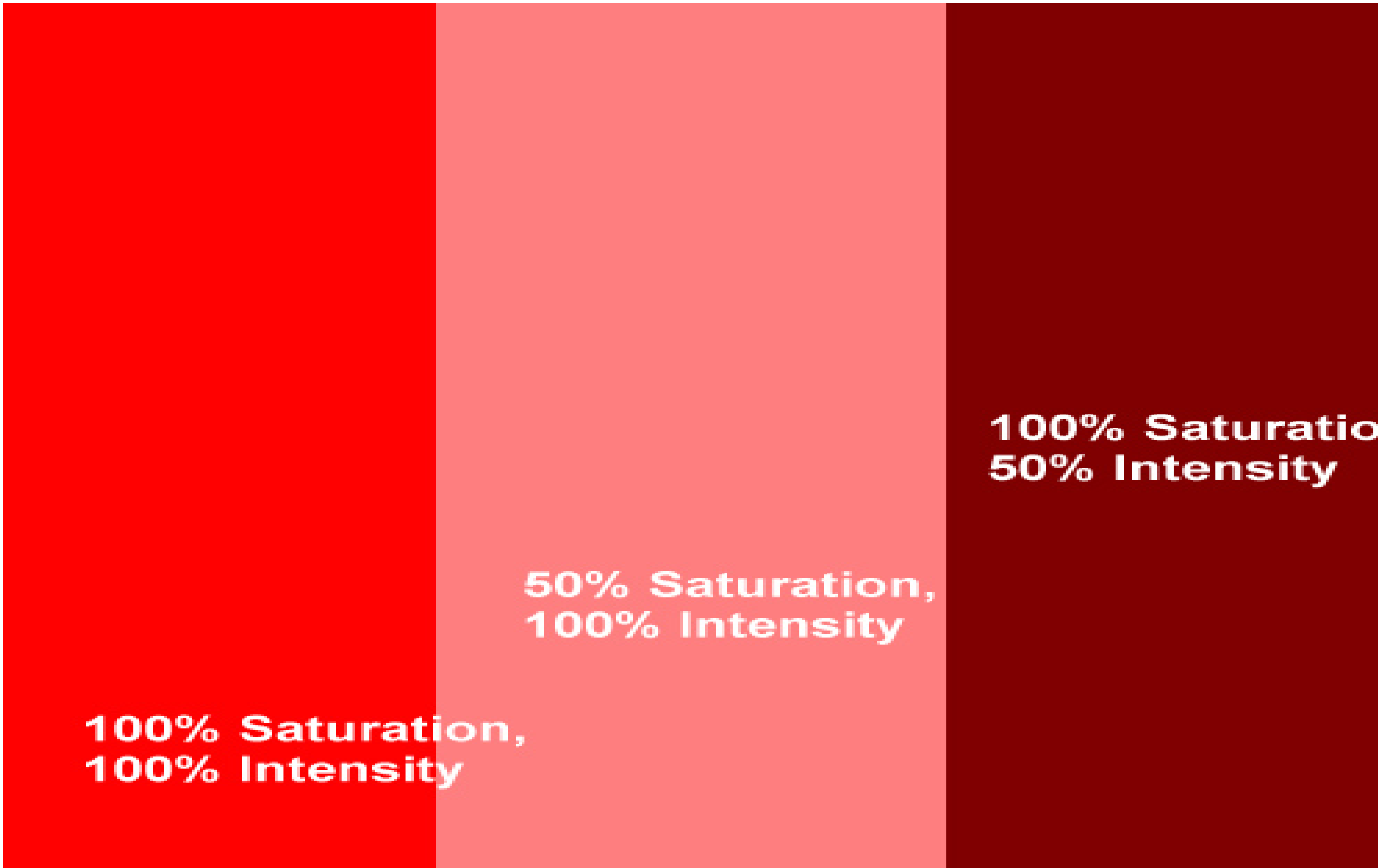
4. The relative proportion of stimulation received by the B cones, in contrast to the R and G cones combined, creates the perception of blue.

5. If all three types of cones are stimulated approximately equally, we see no specific hue -- that is, we see white, gray or black.

The HSI Color System

- Advantage: corresponds naturally to our perception of color
- Hue = shade of color
  - ranging from 0 to 2*Pi Radians (or 0 to 360 degrees)
- Saturation = purity of color
  - ranging from 0 to 1
- Intensity = intensity of color
  - ranging from 0 to 1

The HSI Color System Example

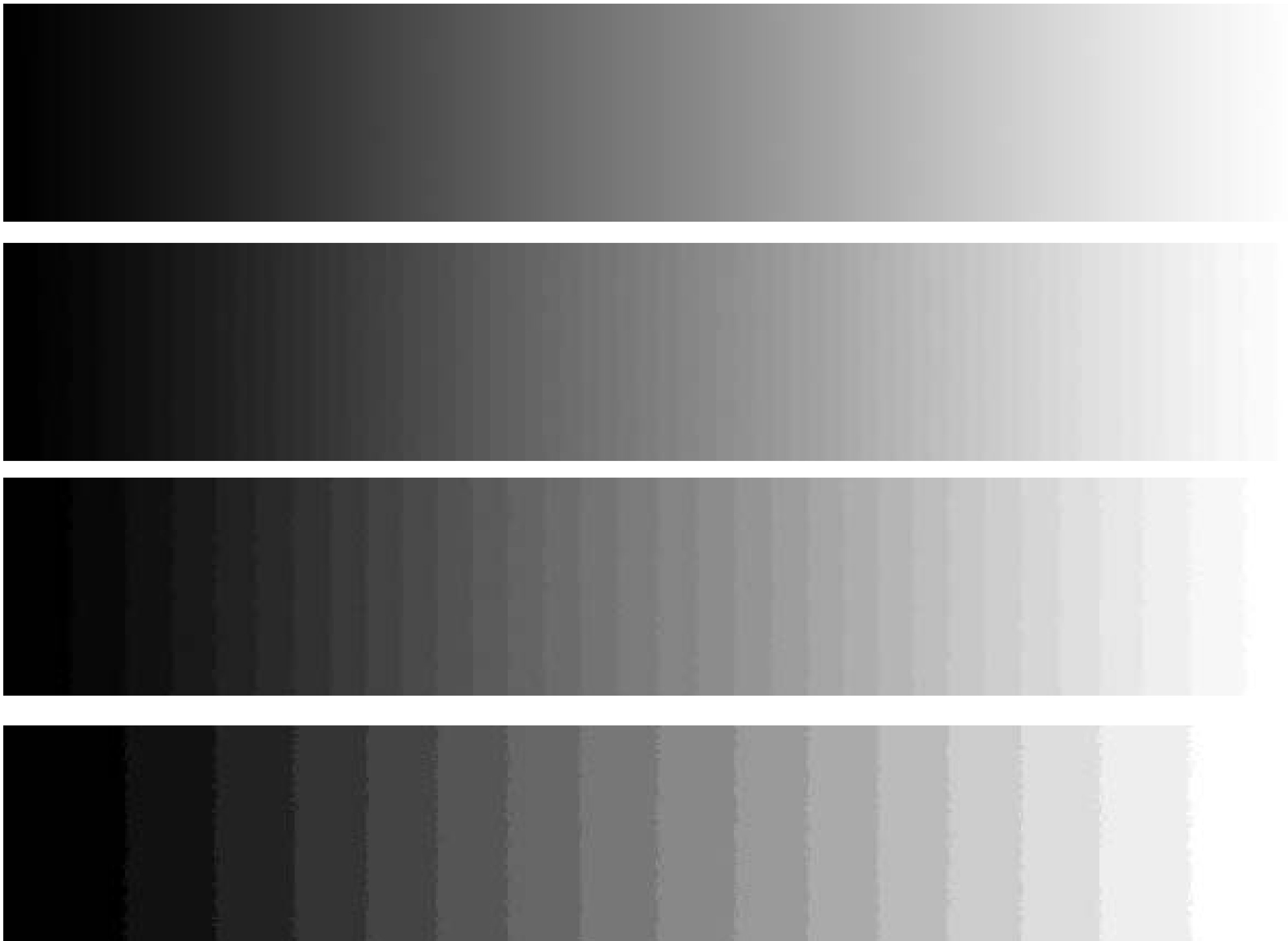


The Gray-Scale Color System

- Gray-scale = intensity component of color only
  - No hue or saturation information
- Resolution
  - 8 bits / pixel is gray-scale
  - 1 bit / pixel is monochrome (black and white)

Grayscale

- 8 bits
- 6 bits
- 5 bits
- 4 bits

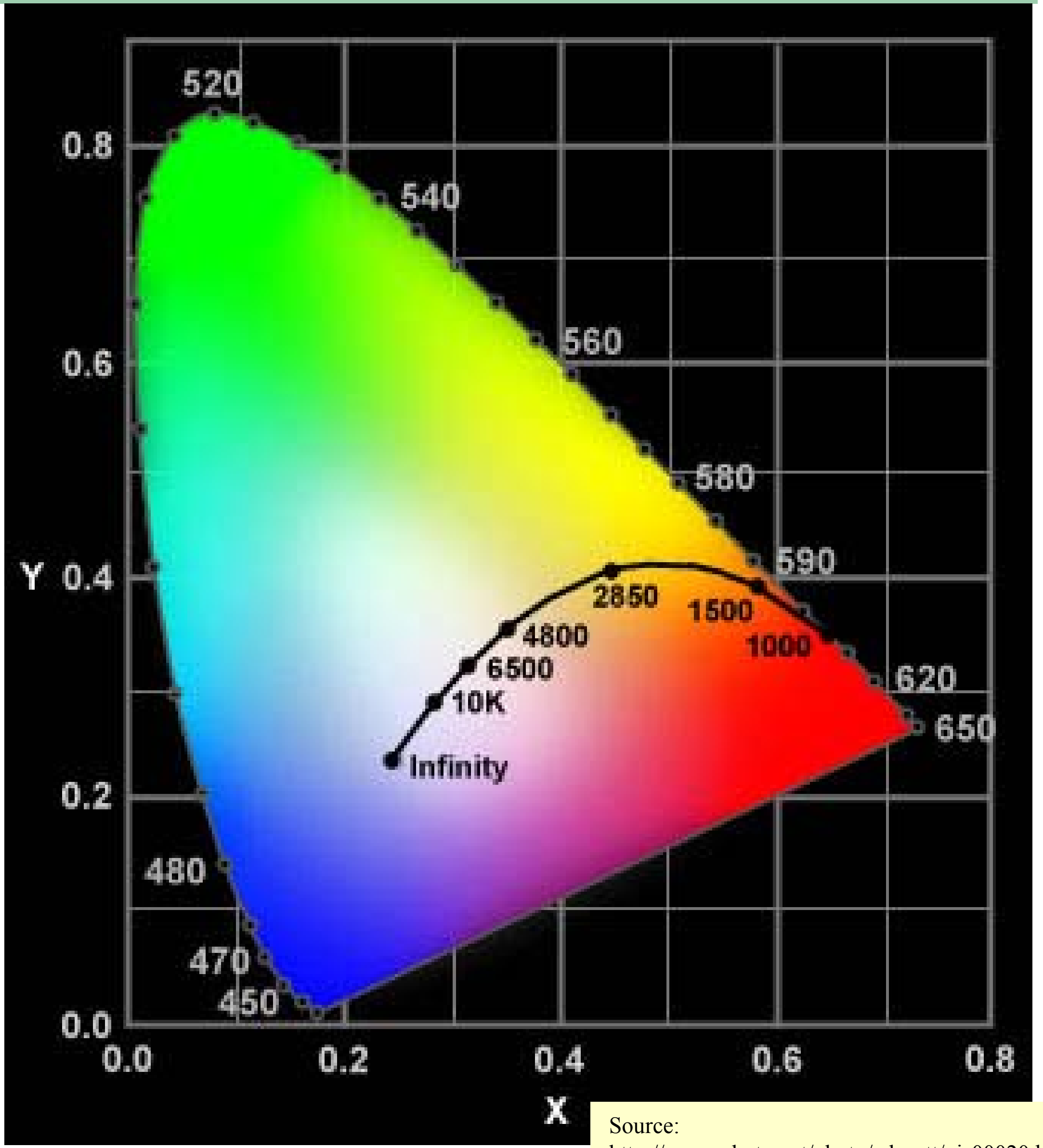


Primary Colors

- Any 3 independent colors describe a region (a triangle) of color
- The 3 corners of the triangle = the primary colors
- No triangle, no matter what primaries are picked, contains all the visible colors

The CIE Color System

- An international color standard
- Represents saturation and luminance only (not brightness)

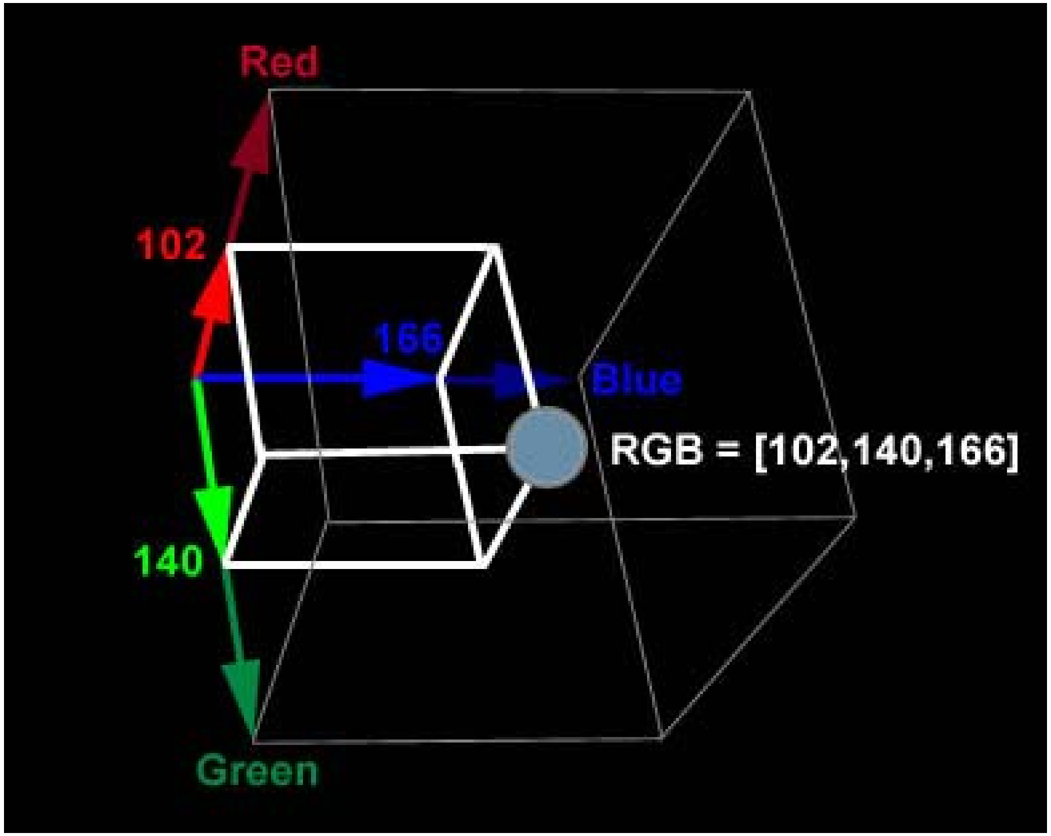


Source:
http://www.photo.net/photo/edscott/vis00020.htm

The RGB Color System

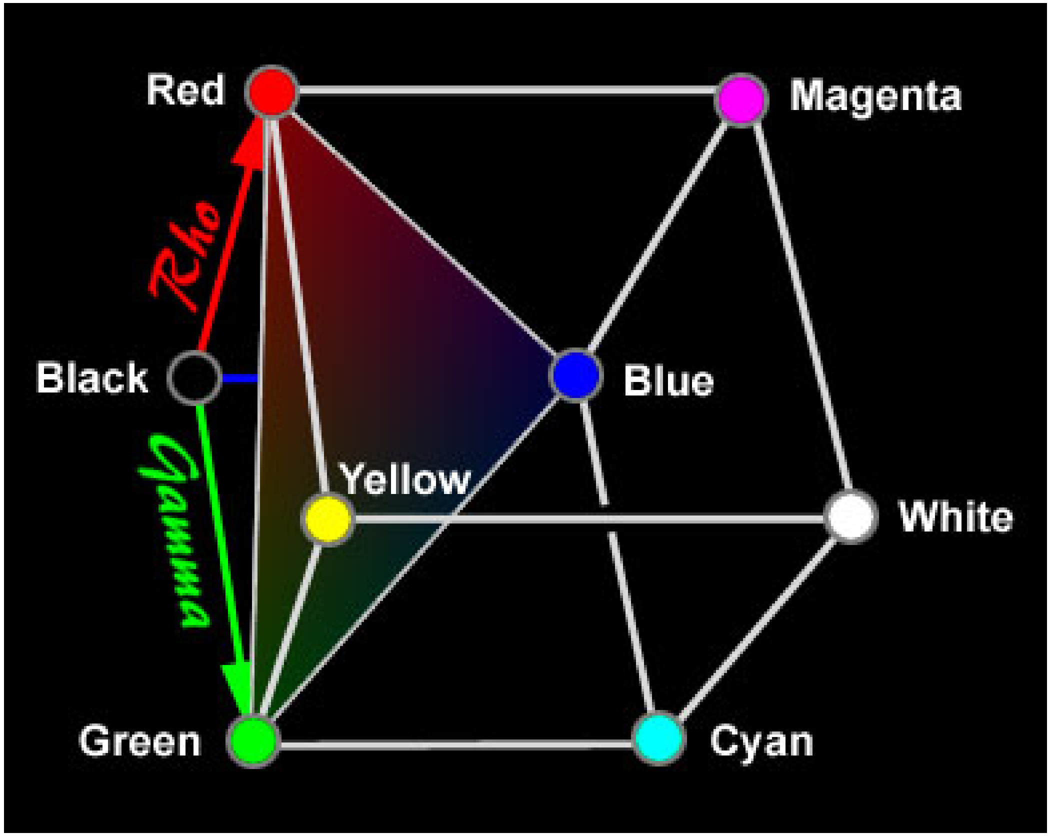
- An additive (emitted light) color system
  - How much R, G, or B is emitted by the light source?
- Advantage: this is the system our retinas are based on
  - and it's the way that monitors are constructed
- Color resolution
  - 8 bits for each primary → 24 bits / pixel

RGB Color Specification



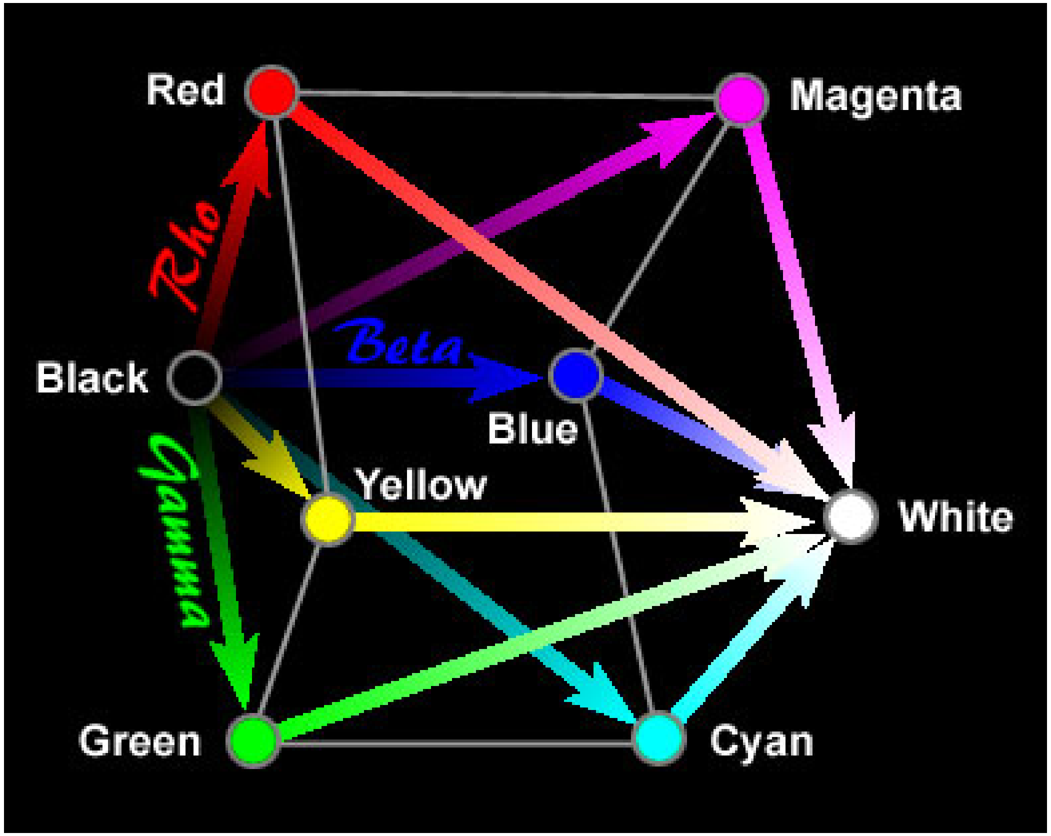
Source:
http://www.photo.net/photo/edscott/vis00020.htm

The RGB "Cube"



Source:
http://www.photo.net/photo/edscott/vis00020.htm

RGB Mixing



Source:
http://www.photo.net/photo/edscott/vis00020.htm

Converting HSI from RGB

- Ranges
  - Let R, G, and B be normalized to be in the range 0…1
  - H = 0…2π radians
  - S, I = 0…1

$\theta = \cos^{-1}(1/2*[(R-G) + (R-B)] / [(R-G)^2 + (R-B)*(G-B)]^{1/2})$

H = θ if G $\geq$ B, otherwise H = 2π - θ

S = 1 – 3 * min(R,G,B)/(R+G+B)

I = (R+G+B)/3

Example

R = 53/255 = .2078

G = 190/255 = .7451

B = 206/255 = .8078

$\theta = \cos^{-1}(1/2*[(.2078-.7451) + (.2078-.8078)] / [(.2078-.7451)^2 + (.2078-.8078)*(.7451-.8078)^{1/2}]) =$

$\cos^{-1}(-.5687/ [.2887 + (-.6)*(-.0627)]^{1/2}) =$

3.046

$H = 2*\pi - \theta$ = 3.23 radians = 185 degrees

S = 1 – 3* min(.2078,.7451,.8078) / (.2078+.7451+.8078) = .6459, or 65%

I = (.2078+.7451+.8078) / 3 = .5869, or 59%

The YCbCr Color System

- The standard used for JPEG, MPEG, etc.
- Y = brightness or luminance
- Cb, Cr = chrominance (color) information
- Ranges
  - Let R, G, and B be normalized to be in the range 0..1
  - Y = 0..1
  - Cb, Cr = -.5..+.5

Converting RGB ↔ YCbCr

$$Y = .299R + .587G + .114B$$

$$Cb = -.169R - .331G + .500B$$

$$Cr = .500R - .419G - .081B$$

$$R = 1.000Y + 1.402Cr$$

$$G = 1.000Y - .344Cb - .714Cr$$

$$B = 1.000Y + 1.772Cb$$

Example

R = 53/255 = .2078

G = 190/255 = .7451

B = 206/255 = .8078

Y = .592

Cb = .122

Cr = -.274

The YIQ Color System

- Used in the NTSC color TV standard
- One luminance channel, two color (chroma) channels
- Conversion from RGB

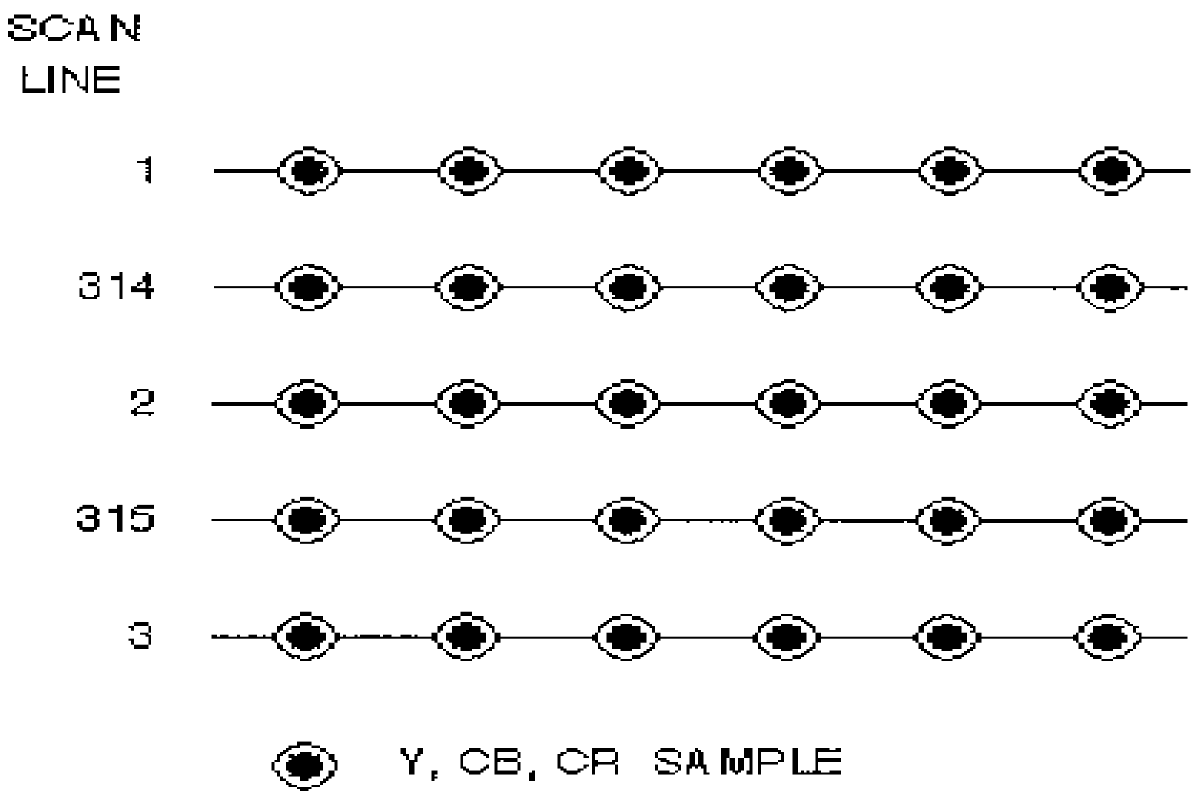
$$Y = .30R + .59G + .11B$$
$$I = .60R - .28G - .32B$$
$$Q = .21R - .52G + .31B$$

- The YUV color system
  - Used in the PAL color TV standard
  - Another color system with one luminance channel and two primaries

Color Channel Subsampling

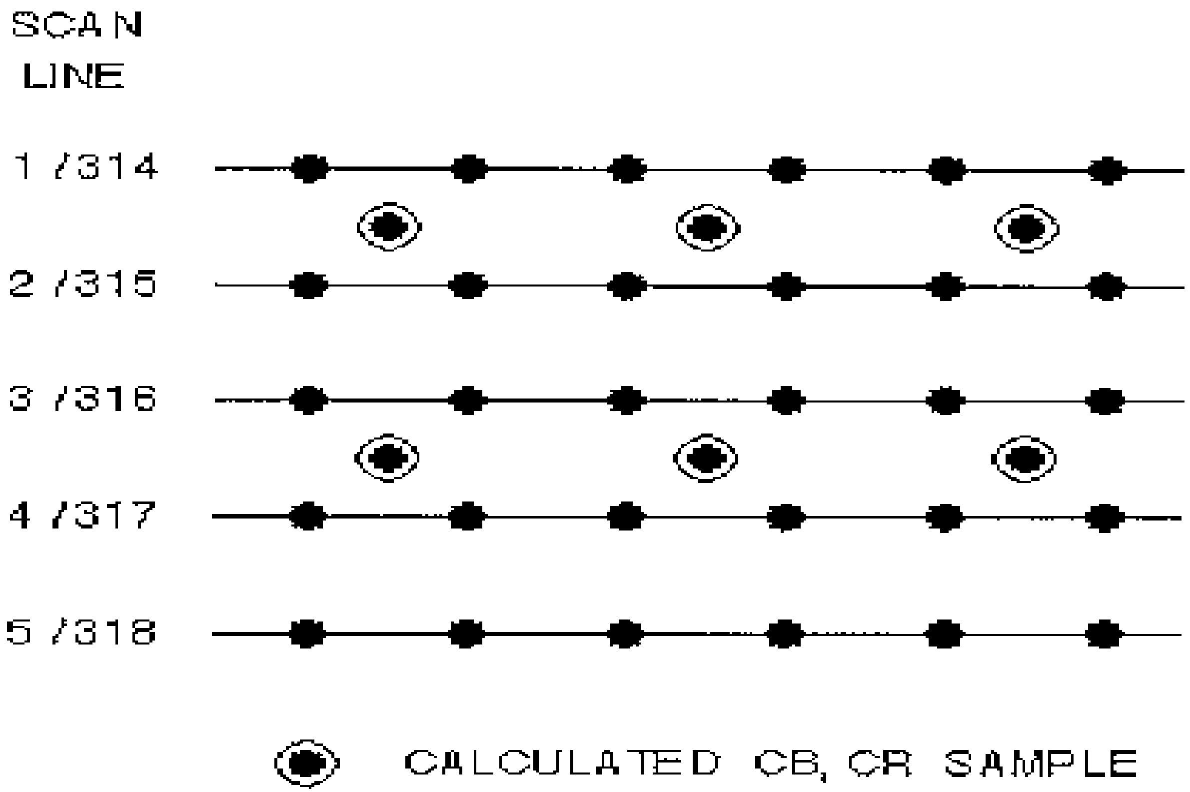
- Goal: compress an image by taking fewer samples
- The eye is more sensitive to brightness (Y) than hue or saturation (Cb, Cr)
  - therefore, take more samples for Y, fewer samples for Cb and Cr
  - Subsampling is used in many image compression formats

4:4:4 Format



- For every pixel, there is 1 sample of Y, 1 sample of Cb, and 1 sample of Cr

4:2:0 Format

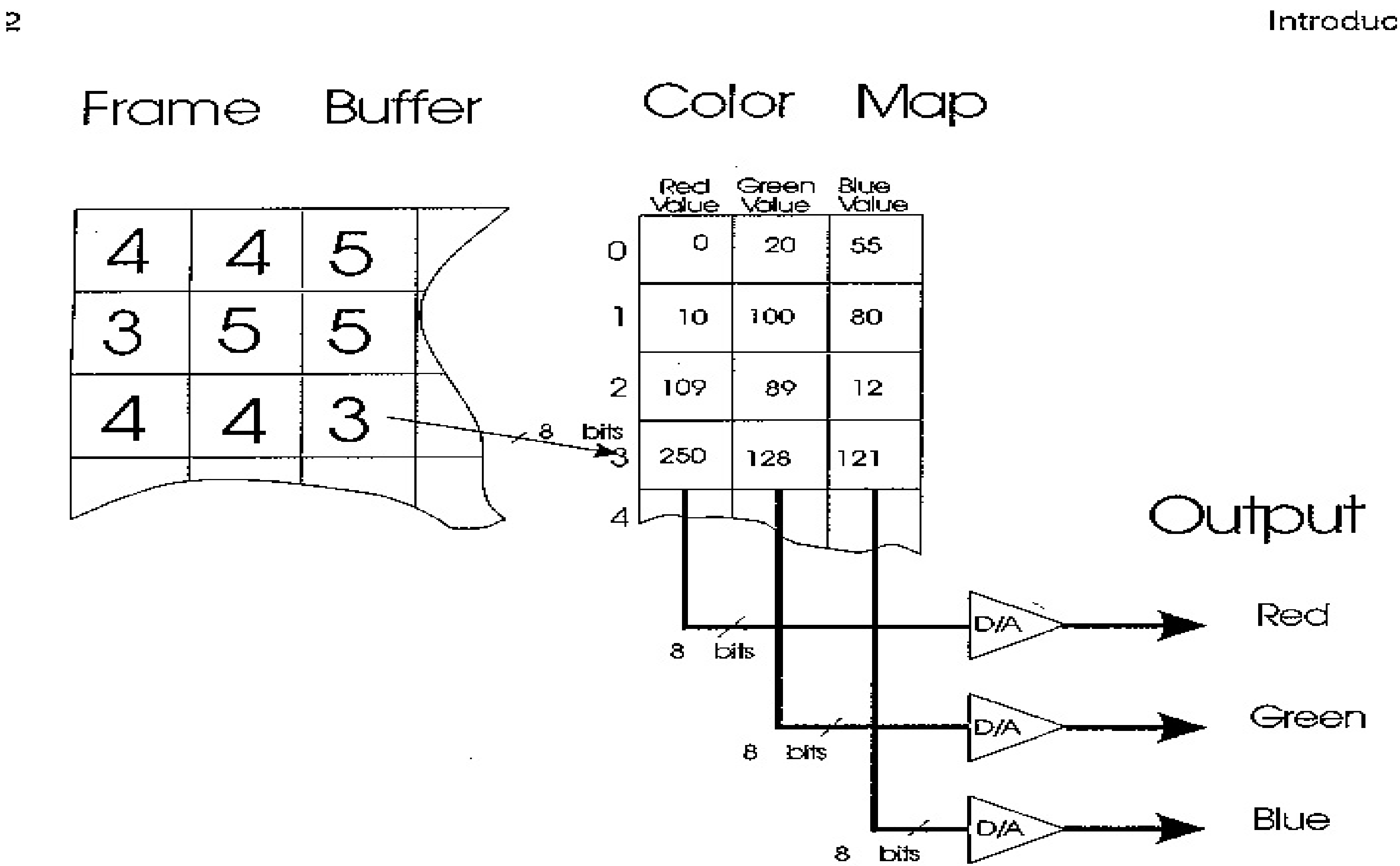


- For every pixel, there is 1 sample of Y
- For every *4* pixels is 1 sample of Cb, 1 sample of Cr

Color Lookup Tables

- Array, or table, of possible color values
  - quantization of color
  - saves memory
  - Improves compression effectiveness
- Palette = the set of colors in the table
  - 8 bits / pixel → 256-color pallette
- Pixel value
  - rather than being a color, it is an index into this array

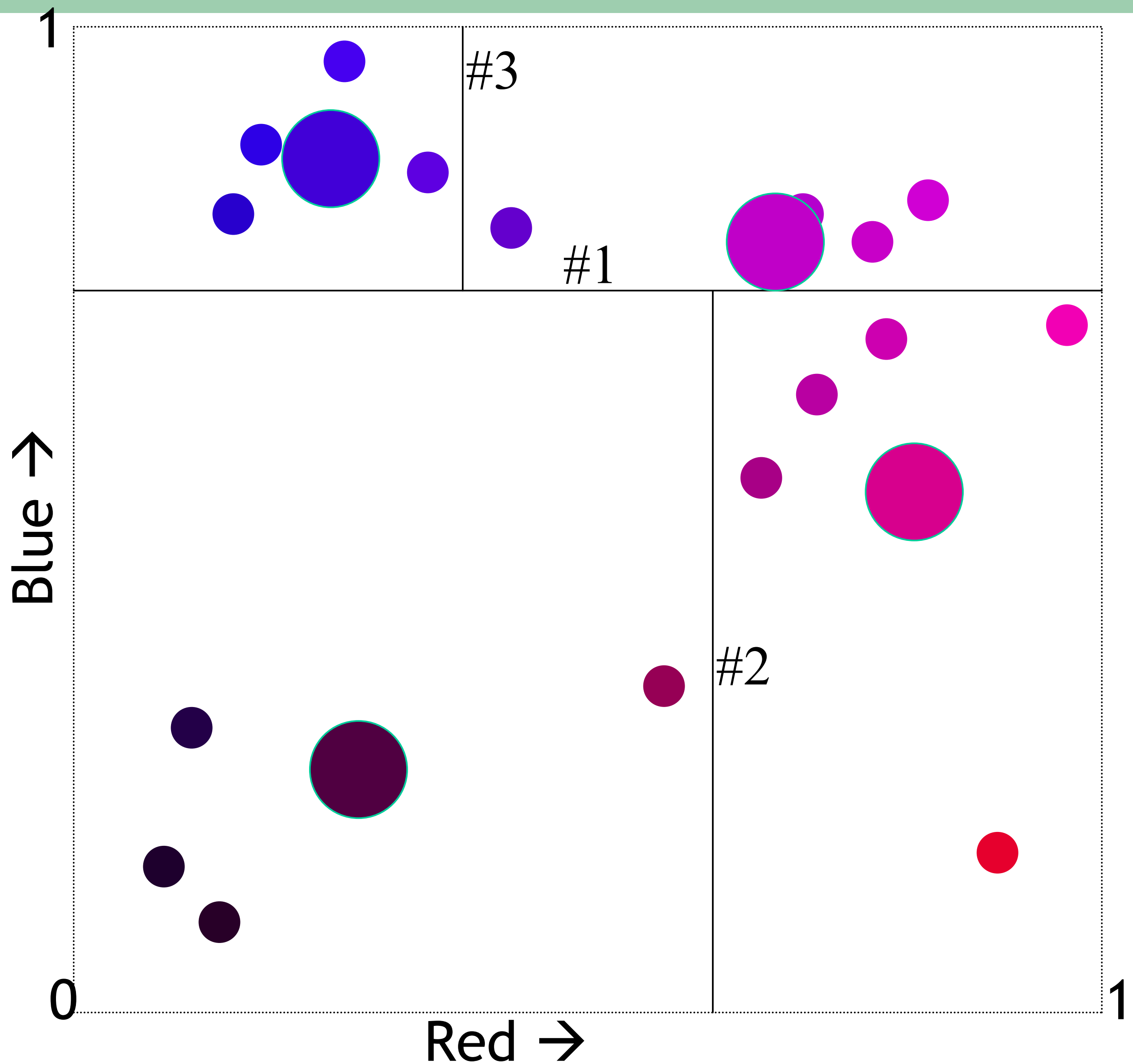
Color Lookup Example



Choosing a Palette

- What is the "best" set of colors to use for the palette?
- One technique: recursive bisection
  - Divide 3-D color space into regions containing approximately the same number of pixels per region
  - Substitute for all pixel colors in this region one of the colors
    - examples: center point in rectangular space, center of mass, …

Example (2-D only, R and B, G=0)



- Goal: replace 17 distinct colors with only 4 values

Diffusing, or Dithering, the Error

- Color sampling introduces errors
- If errors are spatially clustered, errors are obvious
  - "bands" of color
- Solution: spread the errors out
  - Low-pass filter (blur) the error terms

For c_p = next pixel p's color, scanning from left to right, top to bottom

  c_q = quantized color of p (from color palette)

  compute $c_e = c_p - c_q$ = error in c_q

  diffuse c_e to the c_p values of the pixels below and to the right

  replace pixels p's color c_p with c_q

endfor

Diffusion Example (Floyd-Steinberg Algorithm)

diffuse c_e to the c_p values of the pixels below and to the right

→ add 3/8* c_e to pixel immediately below

→ add 3/8* c_e to pixel immediately to right

→ add 1/4* c_e to pixel diagonally below and to right

Example Input Pixel Values

(red only) =

200	150
70	30

Upper left hand pixel value replaced by 160

160	150
70	30

Error (200-160=40) diffused to neighbors

160	165
85	40

Graphics File Info

- Size of image
- Pixel size and shape
- Bits per pixel, and pallette information
- Compression method
- Color value per pixel
- See: graphics file encyclopedias

Processing Image Files

1. Read headers
2. Read palette into 1-D array
3. Read pixel values into 3-D array
   - look up colors in palette array (RGB triple)
4. Process image pixels in 3-D array
5. Write to image buffer

Sources of Info

- [http://www.handprint.com/HP/WCL/color1.html] *Light and the Eye: Color*
- [Crane97] *A Simplified Approach to Image Processing*
- [Watt89] *Fundamentals of 3-D Computer Graphics*
  - Section 14-4 on color models