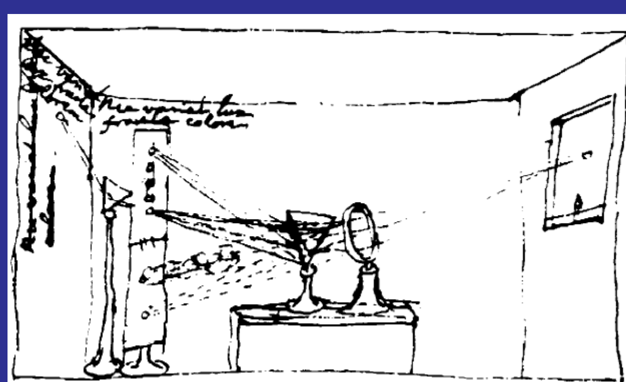
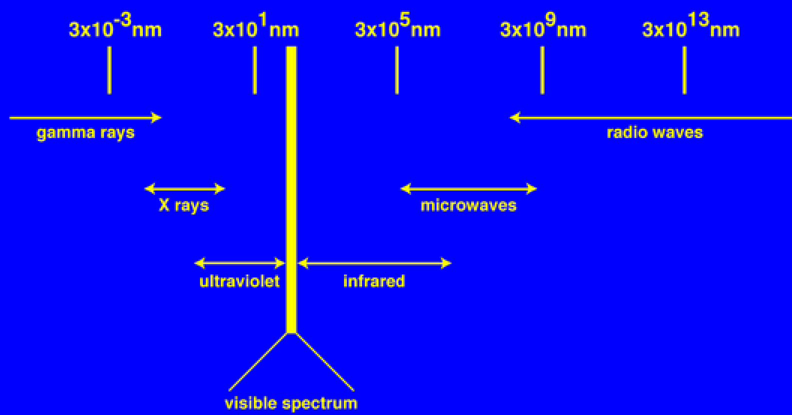
Color

- Physics
  - Light is E-M radiation of different frequencies.
  - Superposition principle
- Perception
  - 3 cones -> 3D color space. (Metamers).
  - Convex subset of 3D linear space.
  - Color matching: can't represent w/ 3 primaries.
- Color Spaces
  - CIE – a standard
  - RGB – a bit more intuitive, Monitors, OpenGL
  - CMYK – subtractive, what we learn in art class.
  - HSV – More intuitive
- More Perception
  - Perceptual distance
  - Context
- Refs: H&B Chapter 12; "The Foundations of Color Measurement and Color Perception", by Brian Wandell: ftp://white.stanford.edu/users/brian/ise/sid-colornotes.pdf

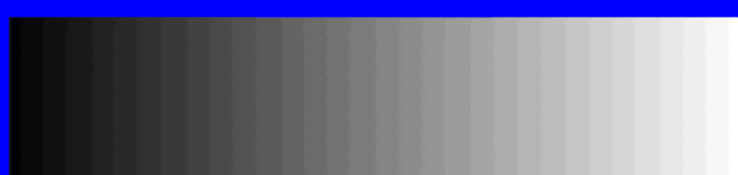
Newton's drawing:



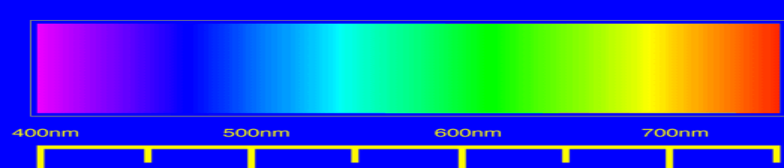
(Wandell)



(Varshney)

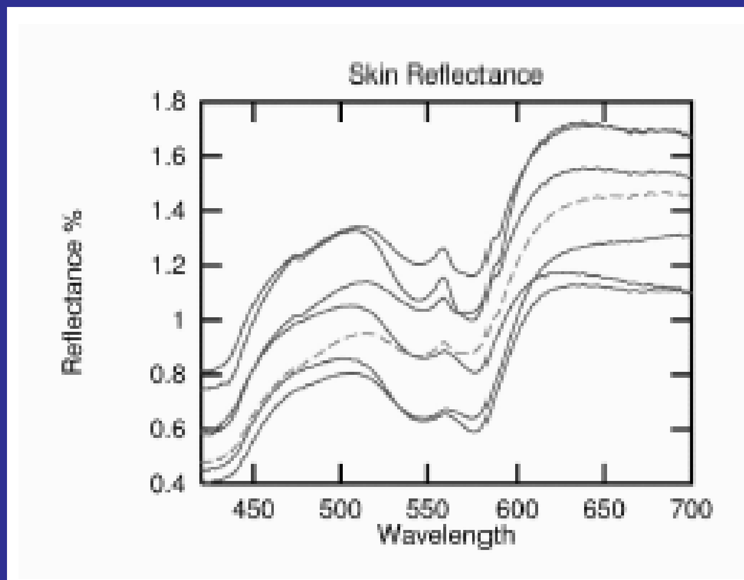


Achromatic light is light without color



Colored light is electromagnetic energy between 400 and 740 nanometers (nm)

Color is a function



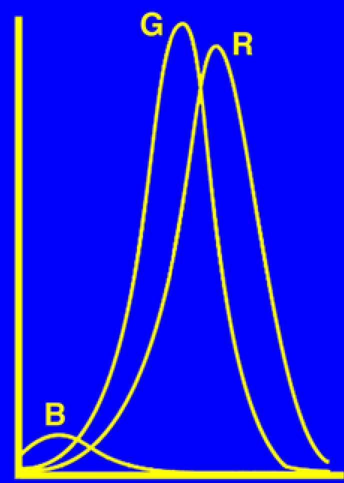
(Angelopoulou)

Superposition

- Light is linear.
- Light from source A + light from source B = Light from sources A & B.
  - Any color is a combination of pure colors.
- Doubling intensity of source doubles amount of light reaching us.

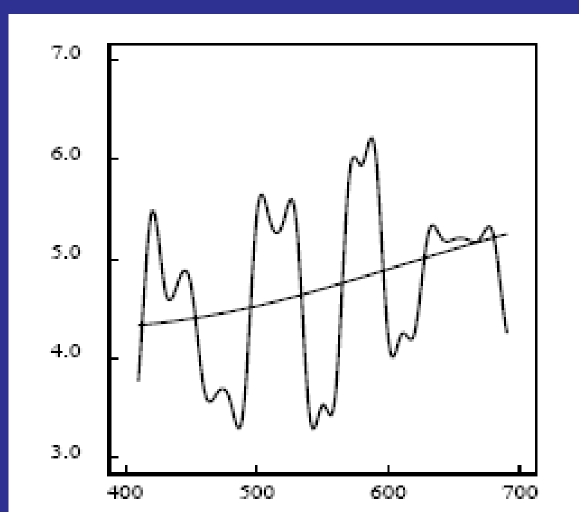
Human Color Perception

- *Cones* allow color perception.
- 3 types of cones sensitive to different frequencies
- Perceptual color depends on how these are stimulated.



tristimulus theory

Metamers

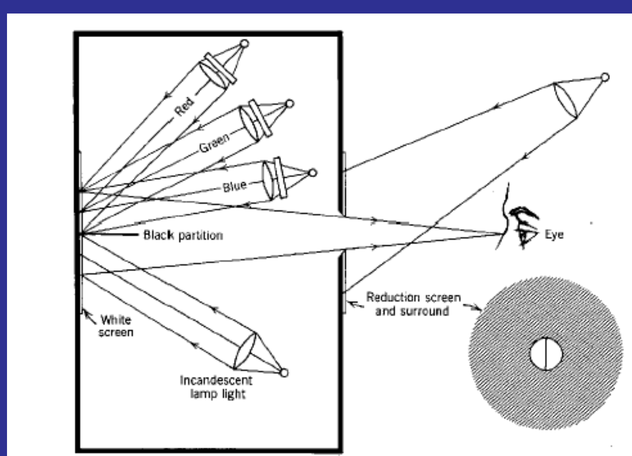


(Wandell)

Perceptual color space

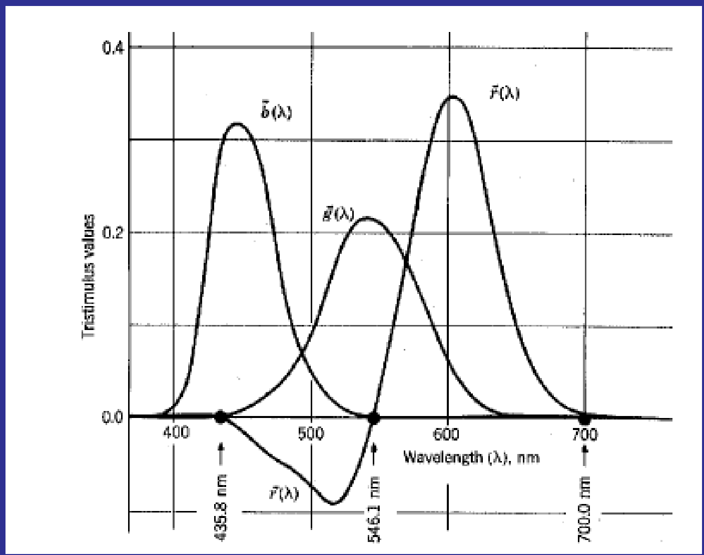
- 3D
- Convex subspace
  - Cones don't have negative response
- So why are there so many color spaces, instead of just one?

Color Matching



Some colors can't be matched

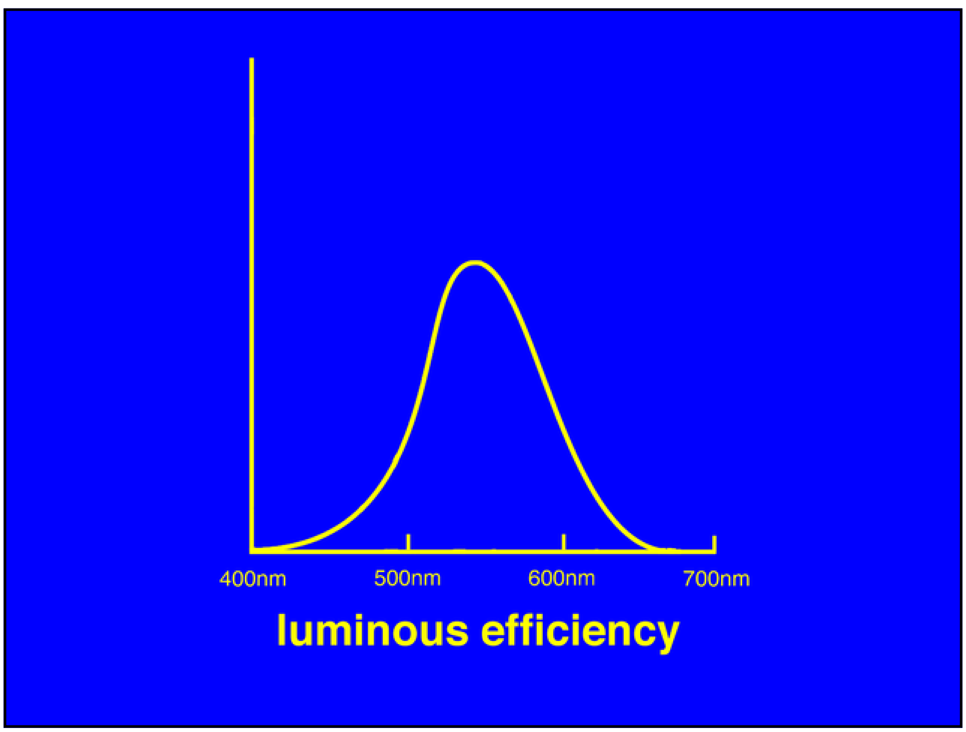
- There isn't a unique color for each cone.
  - "Green" light also excites "red" cones.
  - So to produce some greenish lights we need negative red light.
- But we can match that color + a primary color, using the other two primaries.
- Adding red to our color is like matching it with negative red.
- All colors can be matched like this
  - Shows perceptually color is 3D
  - But we can't have negative light in a display.
  - Display space is convex too, but can't match perceptual convex space.



(Wandell)

Grassman's Additivity Law

- Color matching follows superposition
- If we know how to produce all pure colors, we can produce any color.
- To produce a full spectrum of light from R,G,B
  - Sample it (say, every nm)
  - Do matching experiment for each pure frequency you sample to get (r,g,b).
  - Add up all (r,g,b).
  - Hope you don't need any negative light.

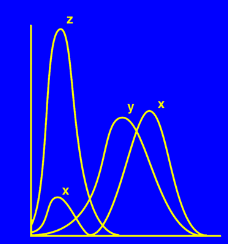


CIE Model

- CIE: International Commission on Illumination (1931)
- Describes any visible color with only positive primaries
- Primaries are called: X, Y, Z

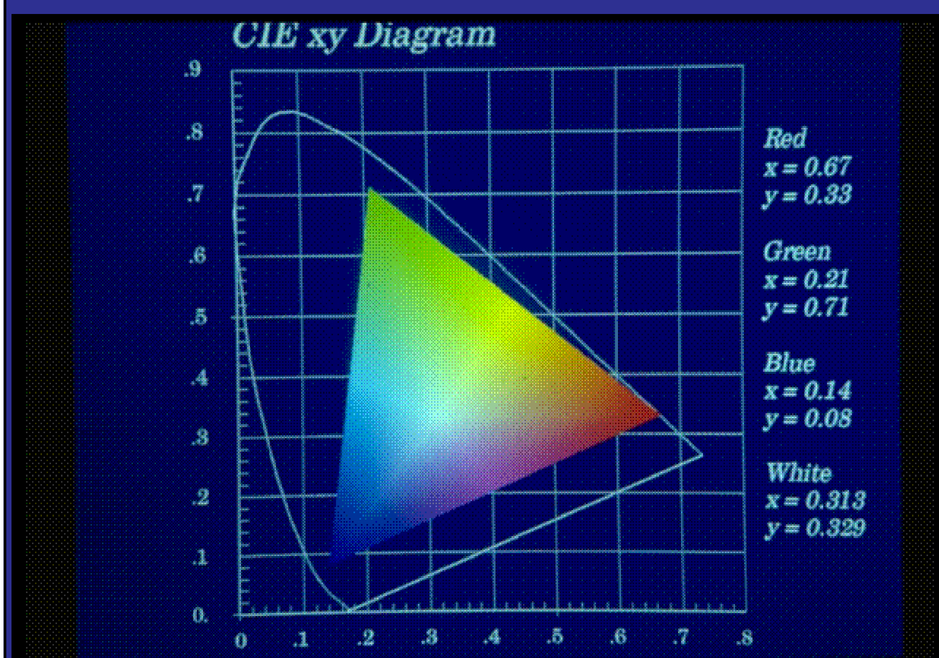
$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = \frac{Z}{X+Y+Z}$$

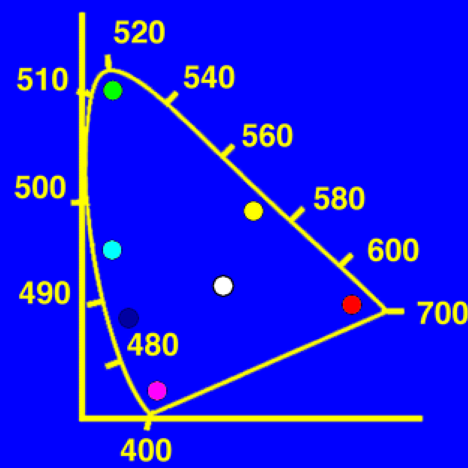
- Color is described by chrominance x, y, and luminance Y



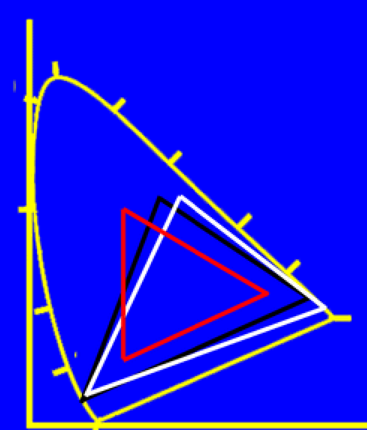
x, y, z weighting functions

"Apart from the very approximate relationship between Y and brightness, there is almost nothing intuitive about the XYZ color-matching functions. While they have served us quite well as a technical standard, they have served us quite poorly in explaining the discipline to new students and colleagues or as an intuition about color appearance."
- Wandell





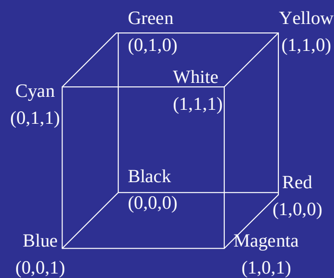
- CIE chromaticity diagram
- plots dominant wavelength and excitation purity

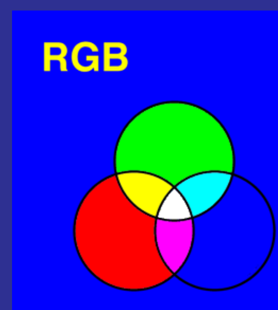


gamut is device's displayable range

Additive Color Model RGB

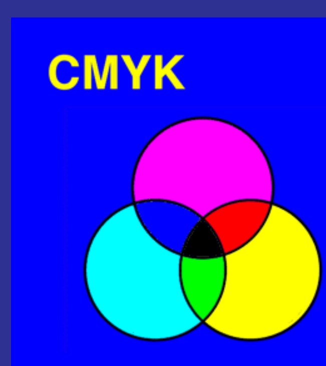
- Mix Red, Green, Blue primaries to get colors
- Cartesian Coordinate System with origin as black.
- Used in display devices: CRTs, LCDs.





Subtractive Color Model CMYK

- Start with white and subtract different subtractive primaries
  - Cyan ink: Absorbs red
  - Magenta ink: Absorbs green
  - Yellow ink: Absorbs blue
- Used in color printing
- C+M+Y = Black, but added fourth black ink for good black color and also to preserve CMY ink for black text

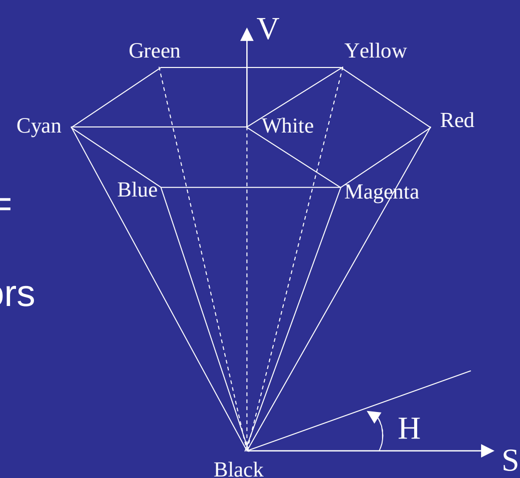


Color Specification

- Hue: Distinguishes among colors
  - red, yellow, blue
- Saturation: Color *Purity* (difference from white)
  - blue and sky blue
- Lightness: Perceived intensity of reflected light
  - blue and darker shades of blue
- Brightness: Perceived intensity of self-luminous objects
- Artists:
  - Tint: Add white (decrease saturation)
  - Shade: Add black (decrease lightness)

Perceptual Color Model HSV

- R = 0^o, G = 120^o, B = 240^o
- Complementary colors are 180^o apart
- S = 0: Gray levels

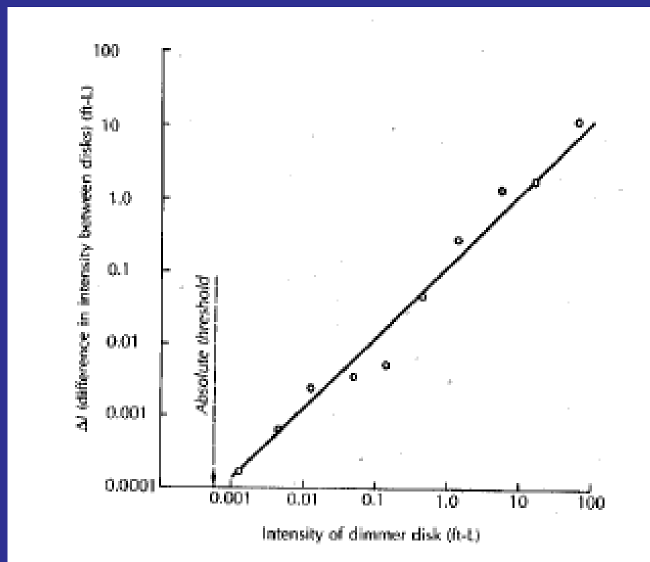


RGB to HSV

- v = max(r,g,b).
- To get pure hue, let m = min(r,g,b). Then (r,g,b) – (m,m,m) is pure hue.
- s is value of pure hue color / v. That is: s = (v – m)/v
- Finally, we can describe pure hue as an angle.

Logarithmic Perception

- Constant ratios of intensities are perceived as being equally different
- Example: intensities of 0.01 and 0.02 will be perceived to be have the same difference between them as intensities of .5 and 1.0
- Other examples of logarithmic perception:
  - Decibel scale: sound
  - Richter scale: earthquakes



Logarithmic Display

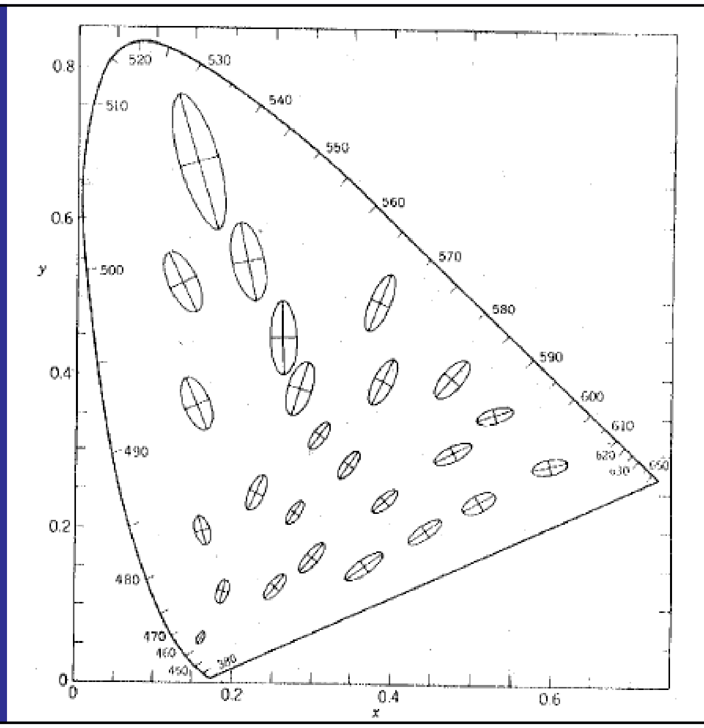
- Uniformly partitioning the displayable intensity range into equidistant arithmetic intervals is wasteful
- Each intensity should differ from the previous by a constant ratio.

I_0 = minimum non-zero displayable intensity

I_n = maximum displayable intensity = 1.0

$I_1 = r\, I_0,\quad I_2 = r\, I_1 = r^2\, I_0,\quad I_3 = r\, I_2 = r^2\, I_1 = r^3\, I_0$

$I_n = r^n\, I_0 \Rightarrow r = (1/\, I_0)^{1/n}$



Gamma Correction

- $I = KV^{\gamma}$

  I = intensity, V = voltage

  K and γ are CRT dependent
- Given an intensity I, find j as:

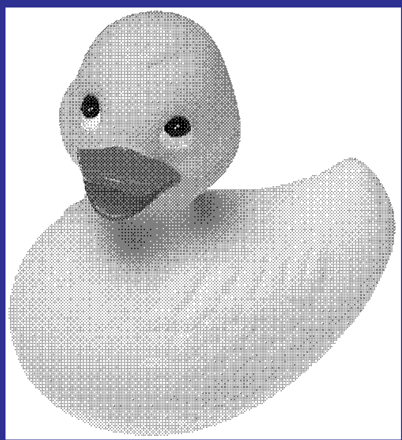
  $j = \text{round}(\log_r (I/I_0))$
- Given j, find $I_j = r_j I_0$ and then find the voltage V_j for a given pixel as:

  $V_j = \text{round}(I_j/K)^{1/\gamma}$

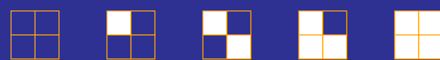
Halftoning

- Halftoning: Smaller black disks for brighter and larger disks for darker areas
- Eyes do the intensity averaging
- Allows more displayable intensity levels at a cost of lower spatial resolution

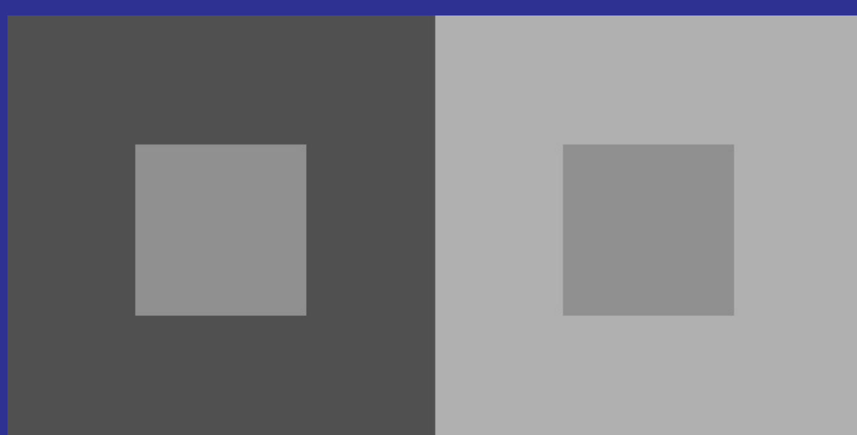
Halftoning



Spatial versus Intensity Resolution

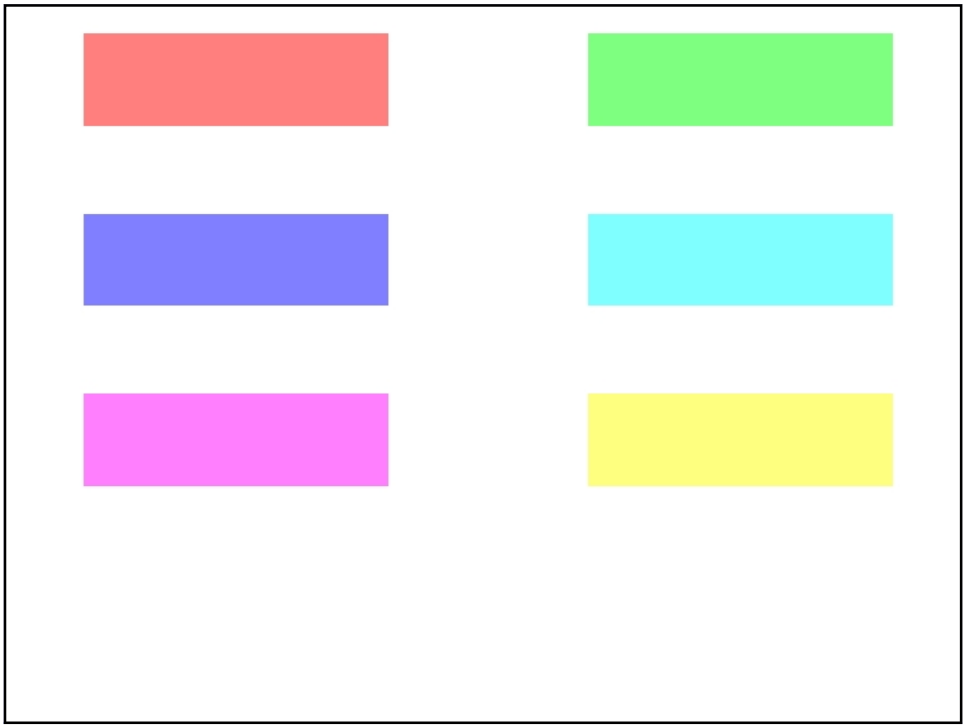


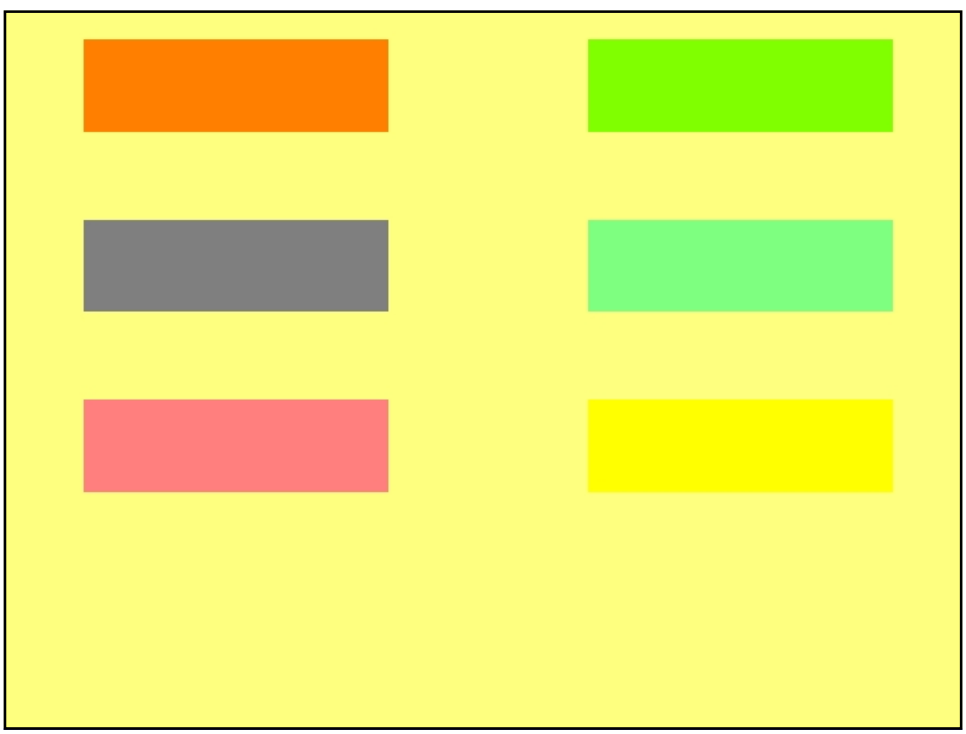
- Halftone Approximation: Dither
  - $n \times n$ pixels encode $n^2 + 1$ intensity levels
- The distribution of intensities is randomized: dither noise, to avoid repeating visual artifacts

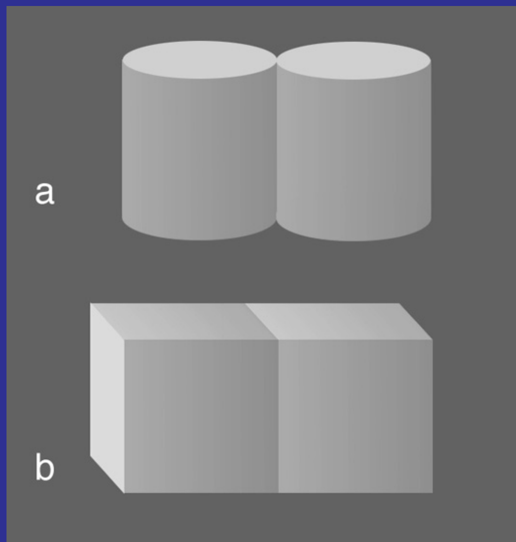


Intensity depends on context

(Adelson)

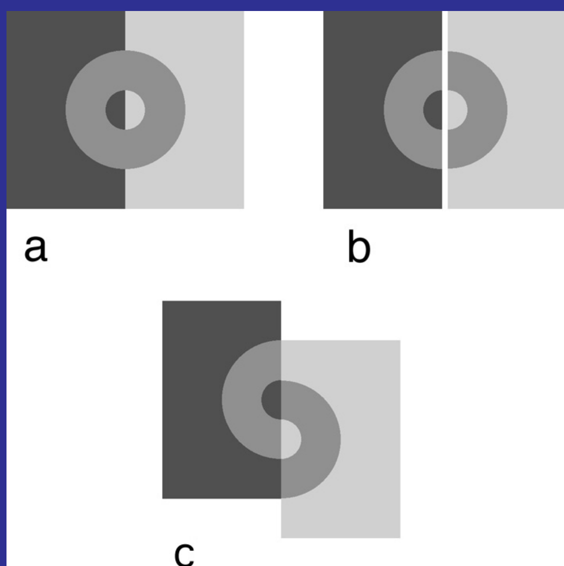






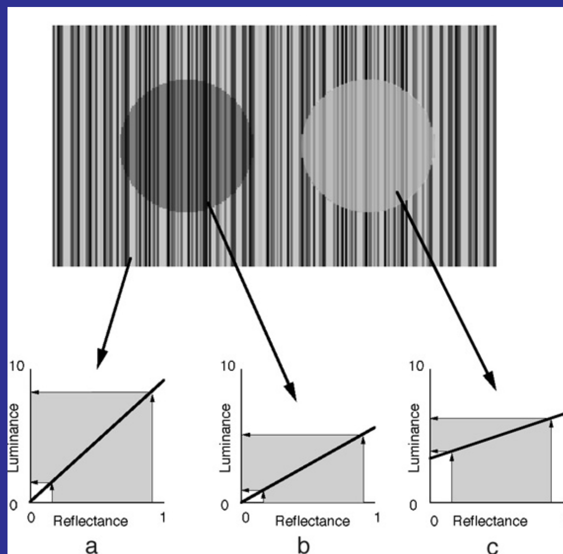
But context is a subtle thing.

(Knill and Kersten)



Context can also create transparency

(Adelson)



Opacity and Blending

- Alpha channel to allow different levels of opacity amongst objects:

  $\alpha = 1 \Rightarrow$ Perfectly opaque

  $\alpha = 0 \Rightarrow$ Perfectly transparent

  $0 < \alpha < 1 \Rightarrow$ Different levels of translucency
- Blending is mixing colors of two sets of pixels: source and destination
  - source and destination each have relative weights or blending factors to control the operation

Blending in OpenGL

- *glEnable* ***(GL_BLEND)***
- *glBlendFunc* (source_factor, destination_factor)
  - GL_ONE, GL_ZERO, GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA, GL_DST_ALPHA, GL_ONE_MINUS_DST_ALPHA
- Eg: *glBlendFunc* (GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA)

Demo

Depth Cueing and Fog

- Depth Cueing: Draw objects farther from the viewer darker
- Fog: Draw objects farther from the viewer whiter
- Let the color to be added with depth be C_f , the color of the pixel be C_s and the factor for blending be f, then:

  $C_s = f\, C_s + (1 - f)\, C_f$
- Depth cueing: f varies linearly with depth, eg: f = 1 - 0.5z
- Fog: f varies exponentially with depth, eg: $f = e^{-0.5z^2}$

glEnable ***(GL_FOG);***
glFogf ***(GL_FOG_MODE, GL_EXP);***
glFogf ***(GL_FOG_DENSITY, 0.5);***
glFogfv ***(GL_FOG_COLOR, fcolor);*** // Glfloat fcolor[4] = {…};


Illumination Attenuation