color

- we have been using r, g, b..
- why
- what is a color?
- can i get all colors this way?
- how does wavelength fit in here
- can i use r, g, b for simulation of reflection
- what is gamma.

the answer

- we will discover that many different light distributions can appear the same
- we will say these all have the same color
- we will see that we can treat color as a 3D linear space
- we will look at various coordinate systems

wavelength

- light is electromagnetic radiation
- monochromatic e.r. has a specific wavelength λ
- Humans are sensitive to light with $380 > \lambda > 770$ measured in nanometers.
- Most real world lights not monochromatic,
  - they are but are a combination of lights of various frequencies.
  - can be described as a light distribution $l(\lambda)$

inside your eye

- humans have various light sensitive cells
- rods: black and white, low lighting, peripheral
- cones: color, good lighting, center of vision
  - we have three types of cones
- each type of cone cell has particular sensitivities to various wavelengths.
- when looking at light, each type of cell fires with some intensity
- the amount of firing gives us some ”color” experience
- if two different light distributions gives us the same firings we get the same experience
  - we will call these two distributions “metamers”
- if you only have two kinds of cones, then you may not be able to distinguish two distributions than I can
  - you are called colorblind

simple model

- the eye has three cone sensitives $k_s(\lambda), k_m(\lambda), k_l(\lambda)$
- how sensitive is each cone type to a pure light of wavelength λ?
- we can draw 3 graphs

- we can show it as a single curve in 3d
- note that we can never get the middle cone to fire alone.
- given light $l(\lambda)$
- we can compute three linear responses

$$S = \int d\lambda \;\; l(\lambda)k_s(\lambda)$$
$$M = \int d\lambda \;\; l(\lambda)k_m(\lambda)$$
$$L = \int d\lambda \;\; l(\lambda)k_l(\lambda)$$

- linear in l means that if i add two lights, i add the responses.
- so all lights will create responses "inside the cone"
- many ways to add together pure lights to get to same point inside the cone.
- we could think of each of these responses as a unique experience and call that a color
- note this data then undergoes some non-linear transforms, so the final signal in the brain is not linear in l.

what is a color

- i would like to derive something like the SML linear color response using simple (19th century) perceptual experiments
- i would like a notion of color as a vector space
- take all light distributions that give create the same color experience as $l_1(\lambda)$.
- call this entire group of light distributions $\vec{c}(l_1(\lambda))$.
- If l_1 and l_2 are metamers, then $\vec{c}(l_1) = \vec{c}(l_2)$
- let call these equivalence classes of light distributions "colors".

color as vectors

- lets try to treat these as vectors
- lets define the vector ops

$$\vec{c}(l_1(\lambda)) + \vec{c}(l_2(\lambda)) \quad \equiv \quad \vec{c}(l_1(\lambda) + l_2(\lambda))$$

  and

$$\alpha\vec{c}(l(\lambda)) \quad \equiv \quad \vec{c}(\alpha l(\lambda))$$

- these must be invariant to metamers
- for example, if

$$\vec{c}(l_1(\lambda)) = \vec{c}(l_2(\lambda))$$

- then we want

$$\vec{c}(l_1(\lambda)) + \vec{c}(l_3(\lambda)) = \vec{c}(l_2(\lambda)) + \vec{c}(l_3(\lambda))$$

- verified by experiment
- same for non-negative scalar multiples

closure by unreal colors

- linear space must be closed under the ops.
- what about $-\vec{c}l$
  - $-\vec{c}(l) = \vec{c}(-l)$
- we must "extend" our space of stuff to include these vectors

extended colors: a formality

- define an extended color to be an expression of the form $\alpha_1\vec{c}_1 - \alpha_2\vec{c}_2$ for positive α_i.
- define two extended colors to be equivalent by moving negatives across the "=" to obtain an expression about real colors
- for example, we say: $+\alpha_1\vec{c}_1 - \alpha_2\vec{c}_2 = +\alpha_3\vec{c}_3 - \alpha_4\vec{c}_4$
  - iff $+\alpha_1\vec{c}_1 + \alpha_4\vec{c}_4 = +\alpha_3\vec{c}_3 + \alpha_2\vec{c}_2$
  - a statement about real colors!
  - colloq: A-B is equal to C iff A looks like B+C.
- we can define addition and scalar multiplies on extended colors.
- so extended colors forms a real vector space.
- some extended colors will be actual colors
- some may not be.
  - we will call those unreal colors.

color matching experiment goals

- will demonstrate that dimension of linear space is 3
- will give us one basis for the space
- will give us a formula for transforming a light distribution into a color coordinate vector

color matching experiment setup

- person watches two screens
- on left screen he is shown a monochromatic light of wavelength λ of some fixed intensity
- on right screen, user is shown superposition of three dial controllable monochromatic lights
  - The wavelengths of the light source happen to be 444, 526 and 645.
- user moves dials to get the lights to look the same.
- if user succeeds, the dial settings would be recorded as $k_{444}(\lambda)$,$k_{526}(\lambda)$,$k_{645}(\lambda)$.
- sometimes user cannot succeed
- so we let user subtract color from the right side.
- this means they moving one of the three over to the left side and see if the sides match.
- user now can succeed
- so extended color space is three dimensional linear space
- resulting three k functions are called matching functions.

coordinates

- Our first choice for a basis will be the colors of the three control lights $[\vec{c}_{444}, \vec{c}_{526}, \vec{c}_{645}]$

- knob settings tell us the coordinates of the color of the monochromatic light $\vec{c}(\lambda)$

$$\vec{c}(\lambda) = \begin{bmatrix} \vec{c}_{444} & \vec{c}_{526} & \vec{c}_{645} \end{bmatrix} \begin{bmatrix} k_{444}(\lambda) \\ k_{526}(\lambda) \\ k_{645}(\lambda) \end{bmatrix}$$

coordinates of general lights

- Since color is linear, the color of a general distribution $l(\lambda)$ can be expressed as

$$\vec{c}(l(\lambda)) = \begin{bmatrix} \vec{c}_{444} & \vec{c}_{526} & \vec{c}_{645} \end{bmatrix} \begin{bmatrix} \int & d\lambda & l(\lambda)k_{444}(\lambda) \\ \int & d\lambda & l(\lambda)k_{526}(\lambda) \\ \int & d\lambda & l(\lambda)k_{645}(\lambda) \end{bmatrix}$$

- these matching functions are behaving just like “sensitivity functions”

shape of color space

- coordinates of pure colors traces out curve
- starts and ends at zero
- all colors are non-negative combinations of pure colors
- pure colors bound the convex cone
- pure colors don’t go inside the cone
- three pure colors cannot generate all colors
- any basis with all real colors in 1st octant must be made of 3 non real colors.

map of color space

- lets normalize out scales of colors
- get 2d diagram
- outer horseshoe represents pure colors
  - red to blue
- combinations of red and blue appear purple
  - not a pure color
- as one goes inside of the horseshoe, we get less “saturation”.
- pick some point in the ctr and call it white.

other bases

- there are three ways to describe a new basis
- give me three new colors
  - by giving three example light distributions
- give me a starting basis and a 3by3 matrix
- give me three new matching functions

give me three colors, i’ll give you a matrix

- give me three new light distributions $e_a(\lambda)$, $e_b(\lambda)$, $e_c(\lambda)$.

- each can be written in original basis

$$\left[\begin{array}{ccc} \vec{c}_a & \vec{c}_b & \vec{c}_c \end{array}\right] = \left[\begin{array}{ccc} \vec{c}_{444} & \vec{c}_{526} & \vec{c}_{645} \end{array}\right] \left[\begin{array}{ccc} M_{1,1} & M_{1,2} & M_{1,3} \\ M_{2,1} & M_{2,2} & M_{2,3} \\ M_{3,1} & M_{3,2} & M_{3,3} \end{array}\right]$$

- one column is the coordinates of one of the basis colors in written in the original basis
- matrix elements can be discovered by integrating against the matching functions

$$M_{1,1} = \int d\lambda \;\; e_a(\lambda) k_{444}(\lambda)$$

give me a matrix, i'll give you new matching functions

- we can derive new matching functions

$$\begin{aligned} \vec{c}(\lambda) &= \left[\begin{array}{ccc} \vec{c}_{444} & \vec{c}_{526} & \vec{c}_{645} \end{array}\right] \left[\begin{array}{c} k_{444}(\lambda) \\ k_{526}(\lambda) \\ k_{645}(\lambda) \end{array}\right] \\ &= \left[\begin{array}{ccc} \vec{c}_{444} & \vec{c}_{526} & \vec{c}_{645} \end{array}\right] \mathbf{M}\mathbf{M}^{-1} \left[\begin{array}{c} k_{444}(\lambda) \\ k_{526}(\lambda) \\ k_{645}(\lambda) \end{array}\right] \\ &= \left[\begin{array}{ccc} \vec{c}_a & \vec{c}_b & \vec{c}_c \end{array}\right] \left[\begin{array}{c} k_a(\lambda) \\ k_b(\lambda) \\ k_c(\lambda) \end{array}\right] \end{aligned}$$

- And so

$$\mathbf{M}^{-1} \left[\begin{array}{c} k_{444}(\lambda) \\ k_{526}(\lambda) \\ k_{645}(\lambda) \end{array}\right] = \left[\begin{array}{c} k_a(\lambda) \\ k_b(\lambda) \\ k_c(\lambda) \end{array}\right]$$

- new m.f are some linear combination of original m.f.s

give 3 matching functions

- that are linear combs of the orig 3 mfs
- 3 are not linearly dependent.
- then can define a basis that uses those mfs.
- implication: when building a color camera, we effectively need 3 sensor response functions.
- will be computing a color iff the sensors are a proper set of mfs.

XYZ basis

- most standard color basis is called the XYZ basis.
- it is made up of three basis colors $[\vec{c}_x, \vec{c}_y, \vec{c}_z]$.
- colors are described with three coordinates $[X, Y, Z]^t$.
- color coordinates can be computed by integrating against the matching functions $k_x(\lambda), k_y(\lambda), k_z(\lambda)$.

xyz matching functions

- the XYZ matching functions were chosen to have some useful properties as well.
- k_y was chosen so that

$$Y = \int d\lambda \;\; l(\lambda) k_y(\lambda)$$

describes the overall perceived brightness of the light $l(\lambda)$.

- so Y is used for black and white
- the 3 mfs are positive everywhere
- so all monochrome light has positive $[X, Y, Z]^t$ coordinates.
  - boundary of horseshoe in first octant
- so all real colors have positive $[X, Y, Z]^t$ coordinates.
  - interior of horseshoe
- basis colors $\vec{c}_x$, $\vec{c}_y$ and $\vec{c}_z$. are outside the horseshoe
- so the basis colors are actually unreal colors

x y color diagram

- in any basis $[0, 0, 0]^t$ is black
- in any basis $\alpha\vec{c}$ is a brighter version of $\vec{c}$
- sometime it is nice to factor out the brightness
- A color $[X, Y, Z]^t$ is normalized to x, y as follows

$$
\begin{aligned}
x &= \frac{X}{X+Y+Z} \\
y &= \frac{Y}{X+Y+Z}
\end{aligned}
$$

- this gives us the xy color diagram
- monochrome is along boundary
  - called saturated
- non monochrome is in interior
- gray/white is in center

SML space

- the 3 SML sensitivity functions are also valid matching functions
- since they are all positive, their corresponding basis colors are unreal.

RGB space

- monitor uses three phosphors, called red, green, blue
  - each produces some fixed distribution
  - some color
- we can use these 3 colors as a basis
- We call the coordinates of a color with respect to this basis $[R, G, B]^t$.
- relationship to XYZ coordinates is

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} .4124 & .3576 & .1804 \\ .2126 & .7152 & .0722 \\ .0193 & .1191 & .9502 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}
$$

$$
\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.2405 & -1.5372 & -.4985 \\ -.9693 & 1.8760 & .0416 \\ .0556 & -.2040 & 1.057 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}
$$

- $R = G = B$ are Gray values
- $[1, 1, 1]^t$ is white.
  - clamped brightness
- some colors need negative coordinates
- out of the the "gamut" of the monitor

Gamma

- In a computer monitor, the voltage levels fed to the pixels determines their output colors
- relationship between voltage and color is not linear.

$$R = R'^{\frac{1}{.45}}$$
$$G = G'^{\frac{1}{.45}}$$
$$B = B'^{\frac{1}{.45}}$$

  - where R' is the voltage sent to the red phosphor.
  - where R is red coordinate of observed color
- The exponent $\frac{1}{.45}$ is known as gamma.

gamma correction

- video cameras purposely correct for this behavior
- record $[R', G', B',]^t$, "gamma corrected" values

$$R' = R^{.45}$$
$$G' = G^{.45}$$
$$B' = B^{.45}$$

- when this is fed to monitor, we get the input color
- many computers have lookup table to apply gamma correction
- so one can send the desired $[R, G, B]^t$ coordinates to the frame buffer.

Perceptual distance

- equidistant colors in the linear XYZ or RGB color coordinates do not appear equally distant
  - we are very sensitive to small differences between dark colors
- There exists a non linear transformation to a color space known as (L^*, a^*, b^*)
- In this color space, coordinate distance is similar to perceptual distance. In this transformation, L^* is defined as

$$L^* = 116Y^{.33} - 16;$$

- similar to gamma correction
- so gamma corrected $R'G'B'$ is a pretty good color space when it comes to perceptual distance.
- important when only 8 bits, 256 different red colors
- into 256 evenly spaced bins in R' coordinates is "smooth"
- into 256 evenly spaced bins in R coordinates is "banded"
- but for image manipulation (compositing), we need to use linear color space

colors for compression

- for compression we like a black and white plus color type space
- since humans are not high res sensitive to color
- we can downsample the color components
- better bit usage is obtained in gamma corrected domain
- so use $Y'P'_BP'_R$ coordinates

$$\begin{bmatrix} Y' \\ B'-Y' \\ R'-Y' \end{bmatrix} = \begin{bmatrix} .299 & .587 & .114 \\ -.299 & -.587 & .886 \\ .701 & -.587 & -.114 \end{bmatrix} \cdot \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix}$$

- Note that due to the nonlinearity of the gamma correction, $Y' \neq Y^{.45}$.
- In this transformation, if the second two coordinates are zero, then we are describing a grayscale color.
- Next we scale the second two coordinates so they fit in the range $[-5...5]$.

$$\begin{bmatrix} Y' \\ P'_B \\ P'_R \end{bmatrix} = \begin{bmatrix} .299 & .587 & .114 \\ -.169 & -.331 & .5 \\ .5 & -.419 & -.081 \end{bmatrix} \cdot \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix}$$

perception research

- human perception is rich and difficult topic
- needed to know how best to allocate various finite resources
  - dynamic range

reflection

- 3 coordinates do not loose any color information
- does loose spectral information
- can get reflection to be incorrect
- take two lights $l_1(\lambda)$ and $l_2(\lambda)$ that are me tamers, $\vec{c}(l_1) = \vec{c}(l_2)$,
- light reflects of off of material with reflectivity per wavelength $m(\lambda)$.
- then reflected light r_1 and r_2 is

$$r_1(\lambda) = l_1(\lambda)m(\lambda)$$
$$r_2(\lambda) = l_2(\lambda)m(\lambda)$$

- the colors of the reflected light are not the same $\vec{c}(r_1) \neq \vec{c}(r_2)$
- must explicitly model $l_1(\lambda)$ until we are ready to look at it
  - high sampling rate in wavelength
  - few people do this.