EECS332 Digital Image Analysis

Color Models and Color Segmentation

Ying Wu

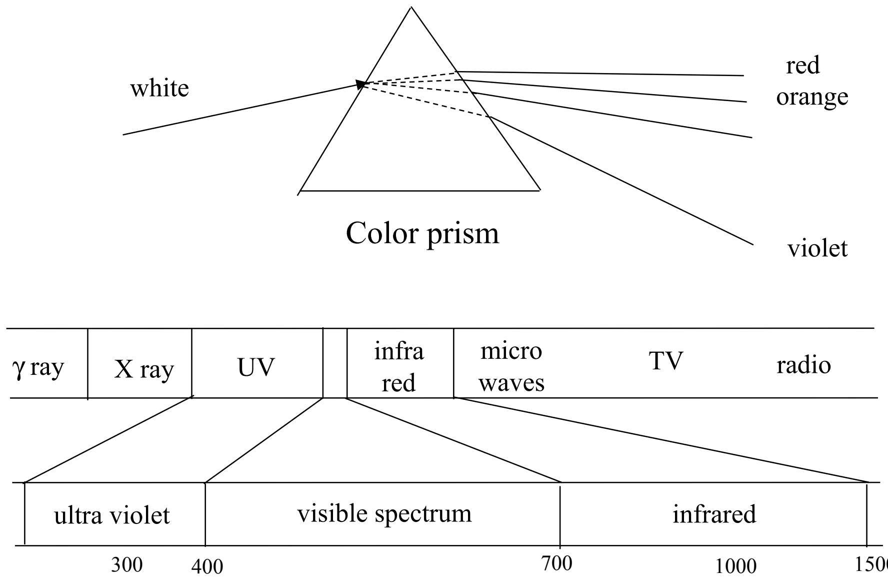
Dept. Electrical Engineering and Computer Science
Northwestern University
Evanston, IL 60208

http://www.ece.northwestern.edu/~yingwu
yingwu@ece.northwestern.edu

Outline

- Color fundamentals
  - Primary color and secondary color
  - Human perception
  - Tristimulus
  - Chromaticity diagram
- Color models
  - RGB
  - CMY
  - HSI
  - HSV
- Color-based image segmentation

Color Fundamentals



Visual cones

- Cones are the sensors in the eye responsible for color vision
- We have 6-7 millions cones
- Cones can be divided into three principal sensing categories, corresponding roughly to
  - R (65%)
  - G (33%)
  - B (2%, but most sensitive)
- Color are seen as variable combination of the primary color, i.e., R/G/B (additive)
- Or the secondary color (subtractive)
  - Magenta --- R+B → W-G
  - Cyan --- G+B → W-R
  - Yellow --- R+G → W-B
- The difference between a CRT display and a printer

Question

- Does any color come from the additive combination of R/G/B?

- Let's keep this in mind and we'll see the answer shortly.

Human Perception

- Human distinguish colors based on "brightness", "hue" and "saturation", rather than R/G/B combinations. (If you ever do painting).
- Hue
  - the attribute associated with the dominant wavelength in a mixture of light waves, representing the dominant color
- Saturation
  - The relative purity, or the amount of white light mixed with its hue. The pure colors are fully saturated.
- Chromaticity
  - H and S capture the chromaticity of a particular color

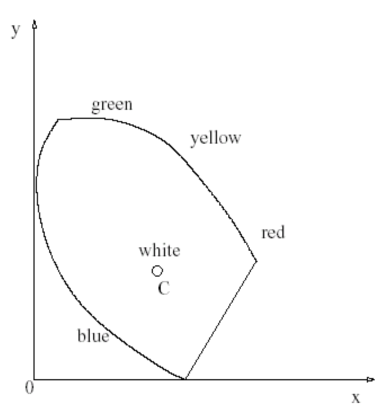
Tristimulus

- Tristimulus
  - The amount of R/G/B needed to form a certain color
  - Denoted by X/Y/Z
- Trichromatic coefficients
  - A color is specified by its trichromatic coefficients

$$\begin{cases} x = \dfrac{X}{X+Y+Z} \\ y = \dfrac{Y}{X+Y+Z} \\ z = \dfrac{Z}{X+Y+Z} \end{cases}$$

Chromaticity diagram

- Defined by the CIE (int'l committee of illumination)
- Used to specify a color
- Shows color composition as a function of x (red) and y (green)

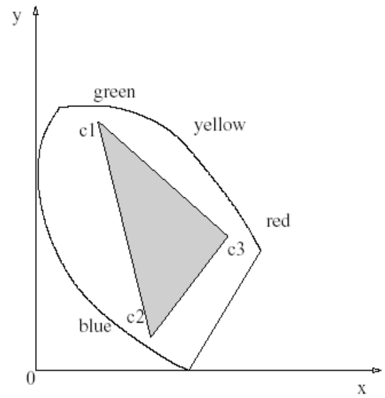


Tone-shaped chromaticity diagram

Boundary: fully saturated color, i.e., pure colors

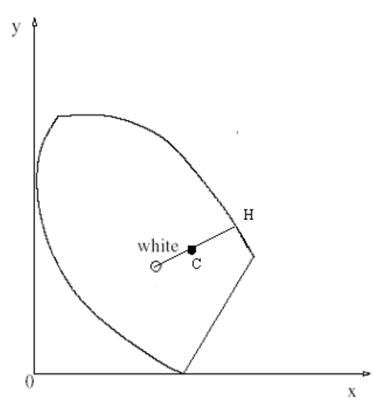
Center: point of equal energy, i.e., white color

Got the answer?



The additive combination of the three primaries can only generate colors inside or on the bounding edges of the triangle. THUS, no set of three primaries can be additively combined to generate all colors.

The use of the Diagram



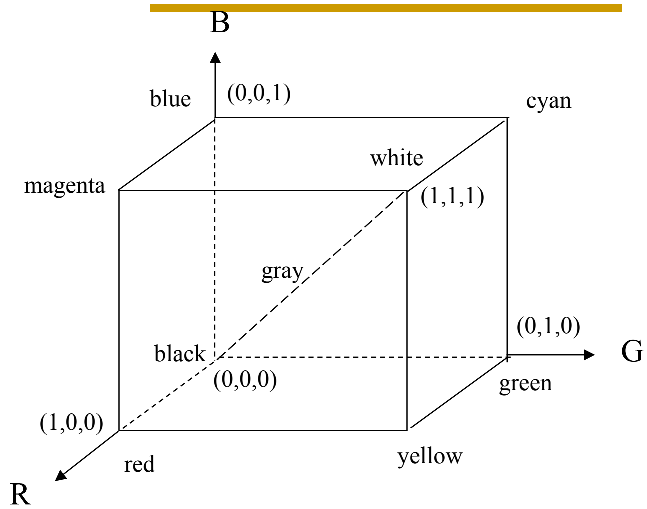
Given a color c, we can find its H and S, by using the chromaticity diagram.

The intersection of the boundary of the diagram and the line passing through the white color and the given color gives H

$s = \frac{\overline{CW}}{\overline{HW}}$ gives the saturation

More importantly, we have c = sH+(1-s)W, meaning a color can be specified by a linear combination of H and W.

RGB color space



<u>Pixel depth</u>: # of bits used to encode the color pixels

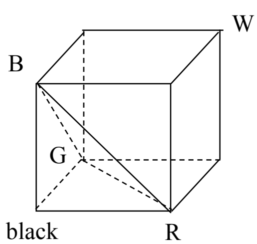
24 bits, full-color, $(2^8)^3$*=16,777,216 colors*

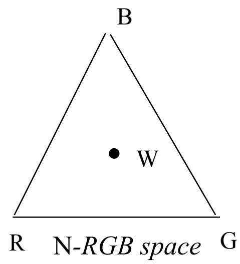
CMY space

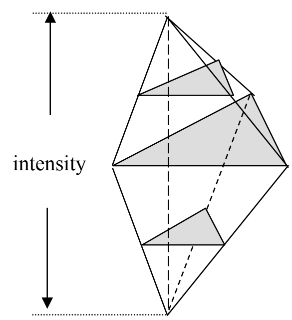
$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

HSI space

- Hue describes a pure color
- Saturation measures the degree of purity (a pure color diluted by white)
- Intensity encodes the brightness
- RGB → HSI ?







$$I=\frac{R+G+B}{3} \qquad r=\frac{R}{R+G+B},\quad g=\frac{G}{R+G+B},\quad b=\frac{B}{R+G+B}$$

RGB → HSI

- N-rgb space is a triangle with corners at (1,0,0), (0,1,0) and (0,0,1)
- Let

$$p=(r,g,b)$$
$$p_r=(1,0,0)\ \rightarrow\ \text{red}$$
$$w=(\tfrac{1}{3},\tfrac{1}{3},\tfrac{1}{3})\ \rightarrow\ \text{white}$$

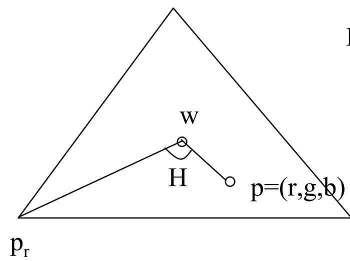
- We can prove:

$$\cos H=\frac{(2R-G-B)}{2\sqrt{(R-G)^2+(R-B)(G-B)}}$$

$$H=\begin{cases}\theta & \mathrm{B}\le\mathrm{G}\\ 2\pi-\theta & \mathrm{B}>\mathrm{G}\end{cases}$$

$$S=1-\frac{3}{R+G+B}\min(R,G,B)$$

Proof?



H is defined by the angle between P_rW and PW

$$\| p-w \| = \sqrt{(r-\tfrac{1}{3})^2+(g-\tfrac{1}{3})^2+(b-\tfrac{1}{3})^2}$$

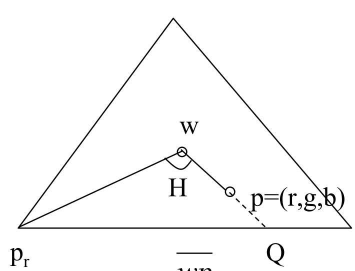
$$= \sqrt{\frac{2(R-G)^2+2(R-B)(G-B)}{3(R+G+B)^2}}$$

$$(p-w)^T(p_r-w) = \frac{2R-G-B}{3(R+G+B)}$$

$$\| p_r-w \| = \sqrt{\frac{2}{3}}$$

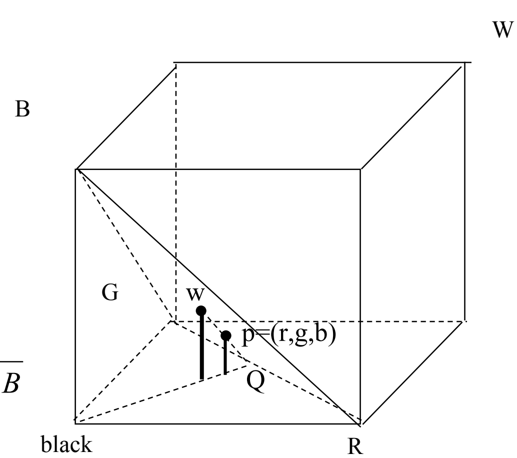
$$\Rightarrow \quad \cos H = \frac{(p-w)^T(p_r-w)}{\| p-w \| \| p_r-w \|} = \frac{2R-G-B}{2\sqrt{(R-G)^2+(R-B)(G-B)}}$$

Proof?



$$S = \frac{\overline{wp}}{\overline{wQ}}$$

$$1-\frac{b}{1/3} = S \Rightarrow S = 1-\frac{3B}{R+G+B}$$



HSI → RGB

RG sector

$$0 \le H < 120^o$$
$$B = I(1-S)$$
$$R = I\left[1 + \frac{S\cos H}{\cos(60^o - H)}\right]$$
$$G = 3I - (R+B)$$

RG sector

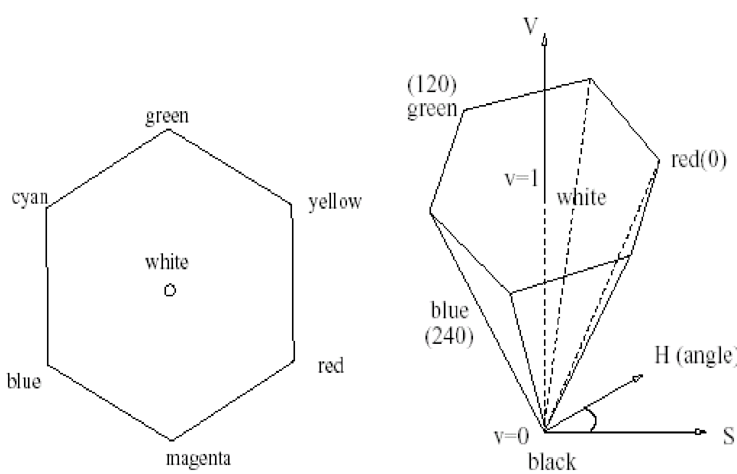
$$120^o \le H < 240^o$$
$$H = H - 120^o$$
$$R = I(1-S)$$
$$G = I\left[1 + \frac{S\cos H}{\cos(60^o - H)}\right]$$
$$B = 3I - (R+G)$$

RG sector

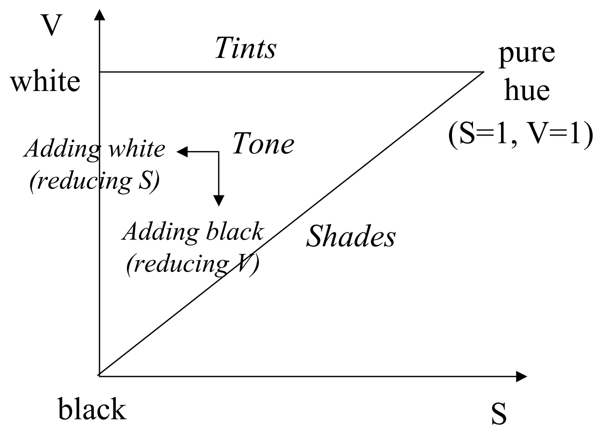
$$240^o \le H < 360^o$$
$$H = H - 240^o$$
$$G = I(1-S)$$
$$B = I\left[1 + \frac{S\cos H}{\cos(60^o - H)}\right]$$
$$R = 3I - (G+B)$$

Let's prove them.

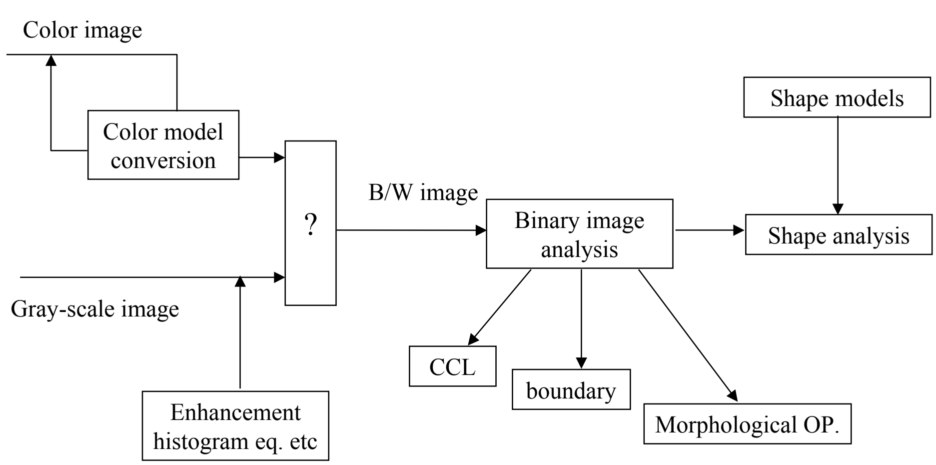
HSV space



HSV space



Where are we?



Segmentation

- Segmentation refers to the partitioning of an image into non-overlapping regions.
- There are all sorts of segmentation methods.
- It is a hard problem. (WHY?)

Basics

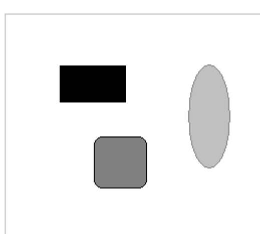
- What a computer cannot do …
  - It is hard to obtain the semantic-level segmentation, e.g., segment out a human, a desk, etc.
- What a computer can do …
  - We can expect the segmentation of low-level image regions (NOT objects).
- How?
  - According to what? → pixel/region similarities

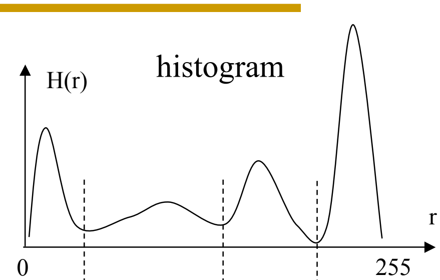
Similarities

- Similarity is a critical issue for segmentation.
- So far, this part is more art than scientific.
- What we generally use:
  - Pixel intensities
  - Pixel colors
  - Texture features/local statistics
  - Filter responses
  - …?

We are going to spend two weeks in segmentation.
Today, let's have a first crack.

Intensity Thresholding





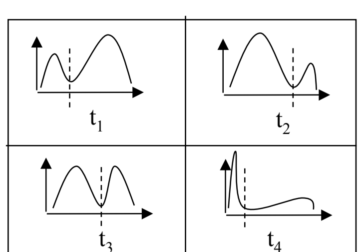
- Obtain the histogram H(r)
- Based on H(r), identify the best thresholds
- How?
  - You may use the K-means algorithm

K-Means

- The task:
  - You are given a set of data points $\{x_1, \ldots, x_n\}$
  - Suppose these data points cluster into K clusters
  - You need to find the centers of these K clusters and then classify each point.
- The K-means algorithm
  - S1: You can guess the initial positions of all the K centers
  - S2: For each data point, put it into the cluster whose center is closest to this point
  - S3: Based on this assignment, re-calculate the center
  - S4: Then iterate between S2 and S3, until it converges.
- Let's run an example
  - Data set: {0 6 1 0 7 2 9 8 3 9 9 1 9 10}

Comments

- Is this global thresholding method good?
- This method may break for many reasons:
  - The perfect threshold may not exist
  - This method is global, but different subregions may need different thresholds



Color Segmentation

- Can we find the regions with a specific color?
  - Application: detect skin-color regions, such as face hands
- Two issues
  - What color feature do you want to use?
    - ✓ RGB?
    - ✓ N-RGB?
    - ✓ HSI?
    - ✓ …?
  - The classification method
    - ✓ Given a color pixel c(u,v), you need to make a binary decision, i.e., is it the color you want, or not.
    - ✓ Methods you can use
      - Histogram models
      - Gaussian models

Using 2D-Histogram

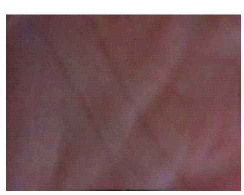
- Suppose you want to find skin color regions
- First, you need to build a skin color model (offline training)
  - S1: collect a large amount of skin color samples
  - S2: find the H and S components of these samples
  - S3: build a 2D color histogram `Histo(H,S)`
    - ✓ Where each bin index a particular (H, S) pair
    - ✓ The value of the bin is the frequency of such a (H,S) pattern appearing in your training sample set.
    - ✓ Normalize it.
- Second, for a testing image
  - For each pixel c(u,v), find its H and S components
  - Use this (H,S) value to index the 2D histogram
  - If `Histo(H,S)` is above a threshold → c(u,v) is a skin color pixel
  - Otherwise, not.

Using a Gaussian Model

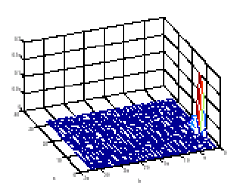
- Roughly the same as before.
- First, you need to build a Gaussian skin color model offline
  - S1: collect a large amount of skin color samples
  - S2: you may choose whatever color model, here I use NR and NG, and denoted by x =(NR, NG)
  - S3: build the Gaussian model $p(x) \sim N(\mu, \Sigma)$
    - ✓ μ is the mean of these color samples
    - ✓ Σ is the variance
- Second, for a testing image
  - For each pixel c(u,v), find its NR-NG components c=(NR, NG)
  - If $(c-\mu)^T \Sigma^{-1} (c-\mu) > t \rightarrow$ c(u,v) is a skin color pixel
  - Otherwise, not.

Examples

(a) A sample skin color region



(a)



(b)

(b) 2D H-S histogram