KE5108: Developing Intelligent Systems for Performing Business Analytics
Lecture 3: Machine Vision III

Dr. Matthew **CHUA**, PhD
Institute of System Science, NUS

© 2017 NUS. All Rights Reserved.

Main References

- Sonka, M., Hlavac, V., & Boyle, R. (2014). *Image processing, analysis, and machine vision*. Cengage Learning.
- Rafael C.. Gonzalez, Richard E.. Woods, & Steven L.. Eddins. (2010). *Digital Image Processing Using MATLAB®*. McGraw Hill Education.
- Ogata, K., & Yang, Y. (1970). Modern control engineering.
- CS131 Computer Vision: Foundations and Applications, Stanford University.http://vision.stanford.edu/teaching/cs131_fall1516/lectures/lecture5_edges_cs131.pdf
- David Forsyth and Jean Ponce –Chapter 8, Computer Vision: A Modern Approach.
- Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Contents of this lecture

1. Color Image Processing

2. Robotic Vision & Navigation

© 2017 NUS. All Rights Reserved.

Color Image Processing

© 2017 NUS. All Rights Reserved.

Spectrum of White Light

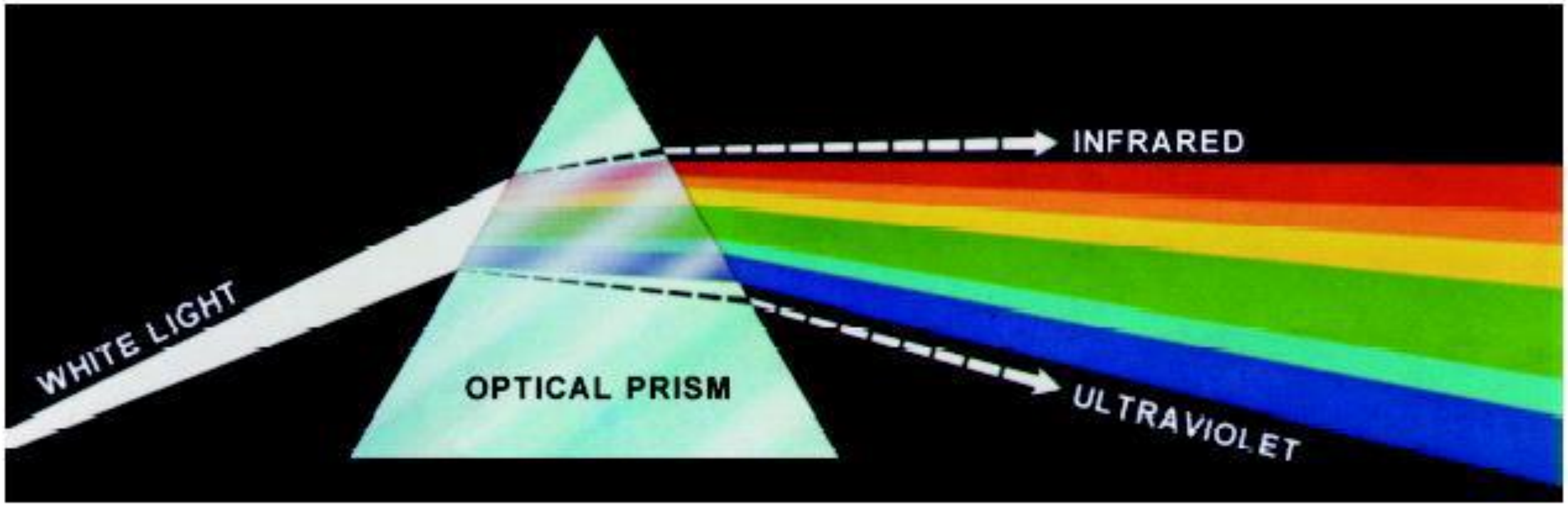
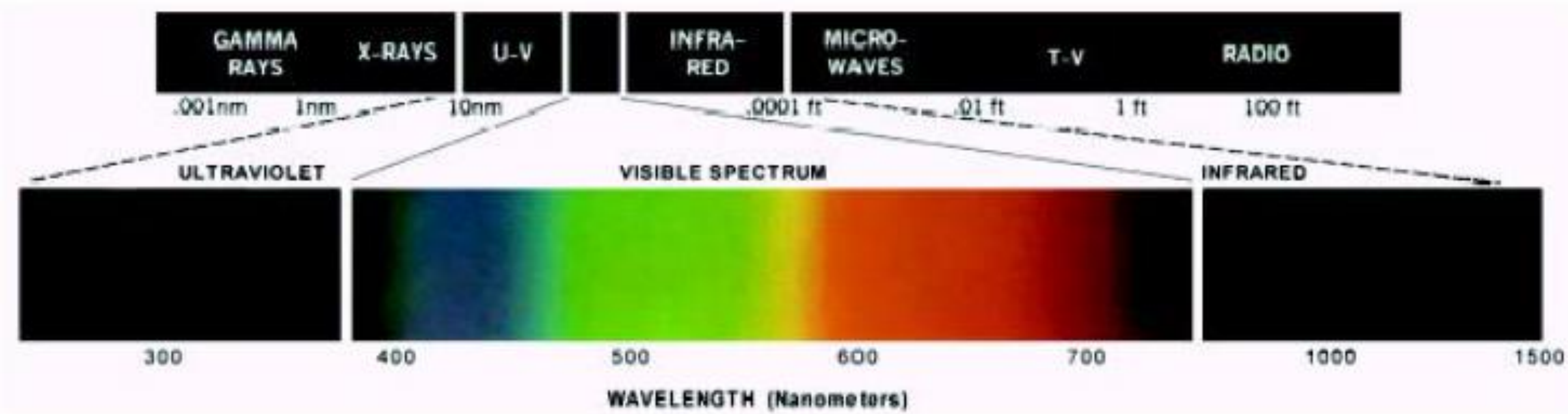


FIGURE 6.1 Color spectrum seen by passing white light through a prism. (Courtesy of the General Electric Co., Lamp Business Division.)

1666 Sir Isaac Newton, 24 year old, discovered white light spectrum.

© 2017 NUS. All Rights Reserved.

Electromagnetic Spectrum



Visible light wavelength: from around 400 to 700 nm

1. For an achromatic (monochrome) light source,
there is only 1 attribute to describe the quality: intensity

2. For a chromatic light source, there are 3 attributes to describe
the quality:

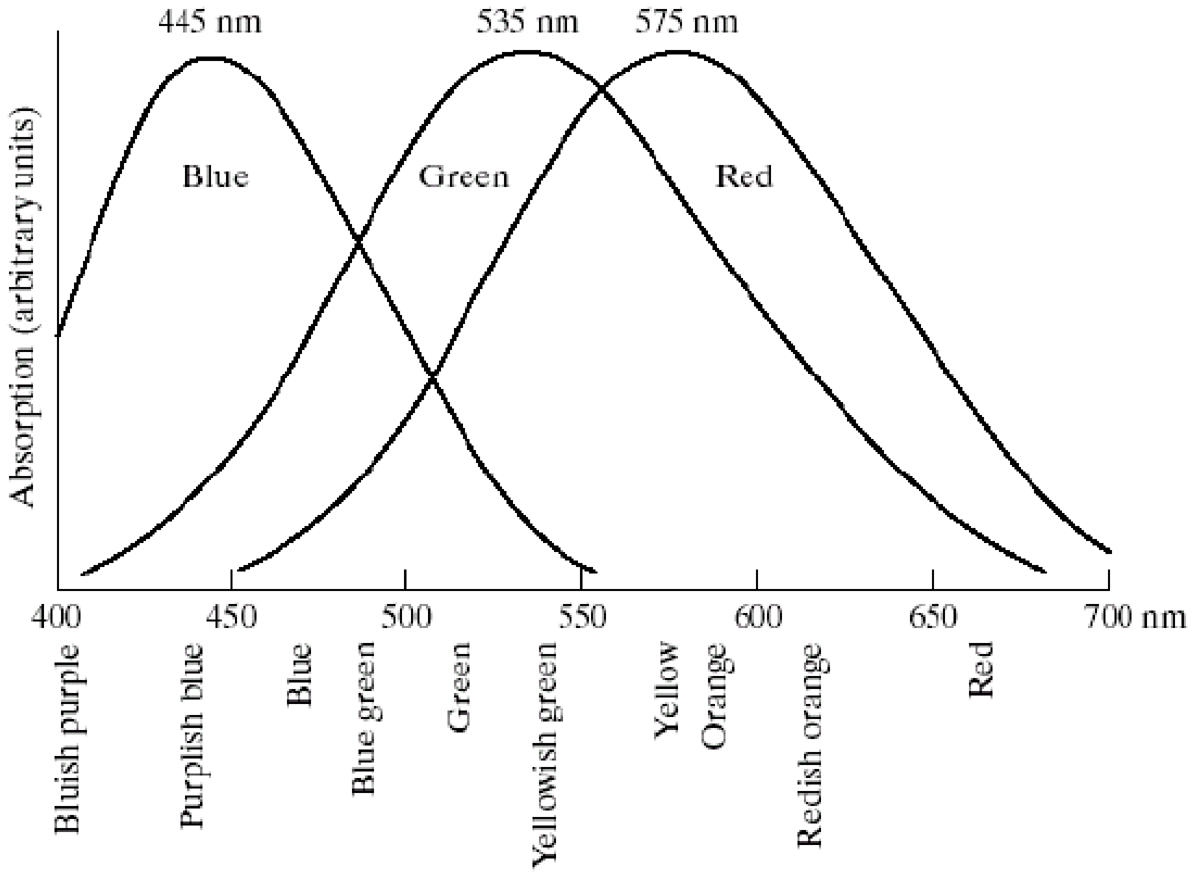
Radiance = total amount of energy flow from a light source (Watts)

Luminance = amount of energy received by an observer (lumens)

Brightness = intensity

© 2017 NUS. All Rights Reserved.

Sensitivity of Cones in the Human Eye



6-7 millions cones
in a human eye
- 65% sensitive to Red light
- 33% sensitive to Green light
- 2 % sensitive to Blue light

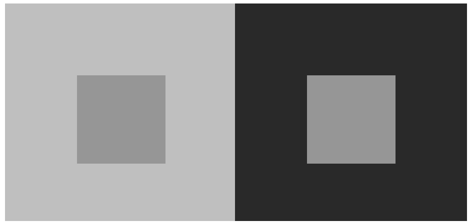
Primary colors:
Defined CIE in 1931
Red = 700 nm
Green = 546.1nm
Blue = 435.8 nm

CIE = Commission Internationale de l'Eclairage
(The International Commission on Illumination)

© 2017 NUS. All Rights Reserved.

Luminance vs. Brightness

Same lum.
Different
brightness



Different lum.
Similar
brightness

- Luminance (or intensity)
  – Independent of the luminance of surroundings

$$L(x,y) = \int_0^{inf} I(x,y,\lambda)V(\lambda)d\lambda$$

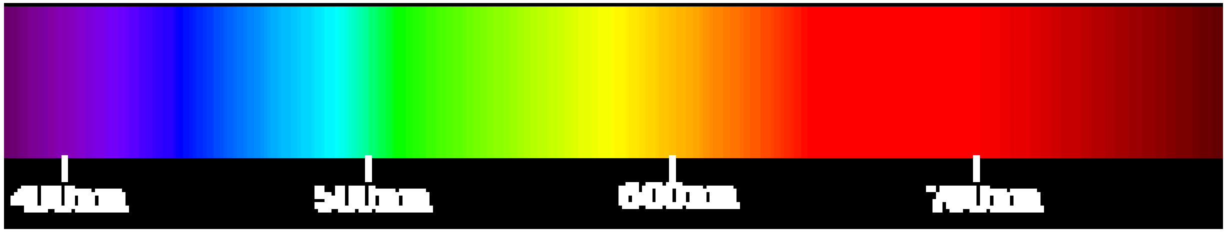
$I(x,y,\lambda)$ -- spatial light distribution
$V(\lambda)$ -- relative luminous efficiency func. of visual system ~ bell shape
(different for scotopic vs. photopic vision;
highest for green wavelength, second for red, and least for blue)

- Brightness
  – Perceived luminance
  – Depends on surrounding luminance

© 2017 NUS. All Rights Reserved.

Color of Light

- Perceived color depends on spectral content (wavelength composition)
  – e.g., 700nm ~ red.
  – “spectral color”
  – A light with very narrow bandwidth

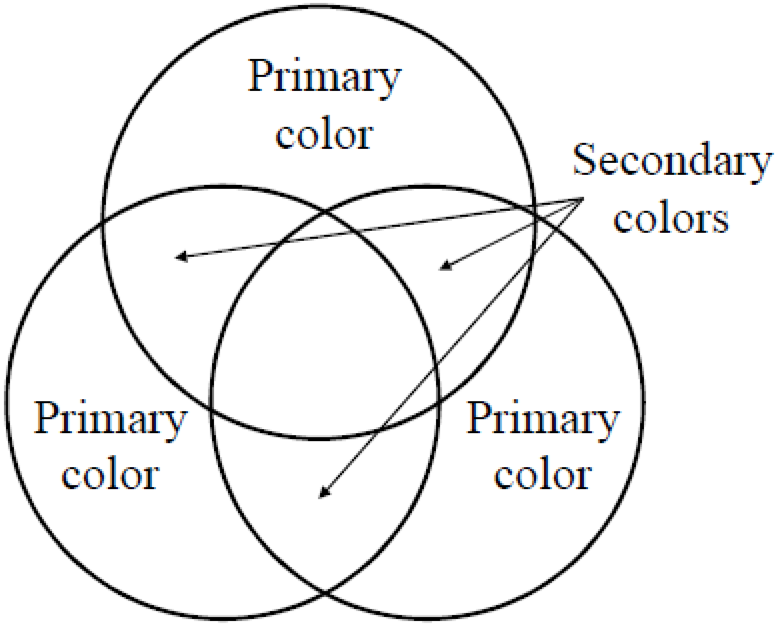


"Spectrum" from http://www.physics.sfasu.edu/astro/color.html

- A light with equal energy in all visible bands appears white

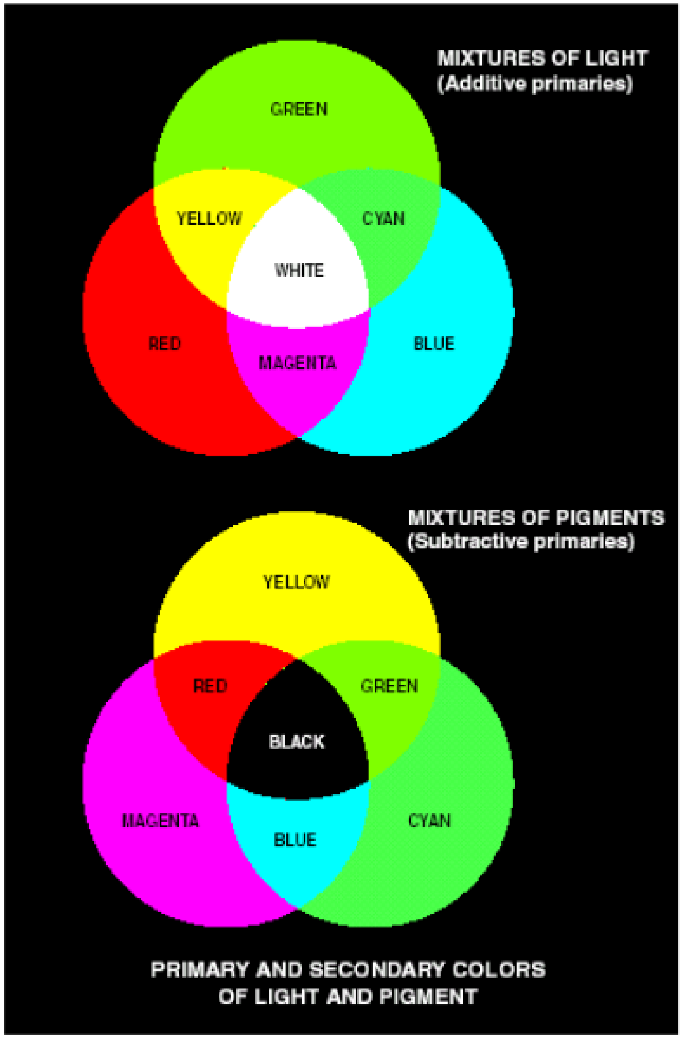
© 2017 NUS. All Rights Reserved.

Primary and Secondary Colors



© 2017 NUS. All Rights Reserved.

Primary and Secondary Colors



Additive primary colors: RGB use in the case of light sources such as color monitors

Subtractive primary colors: CMY use in the case of pigments in printing devices

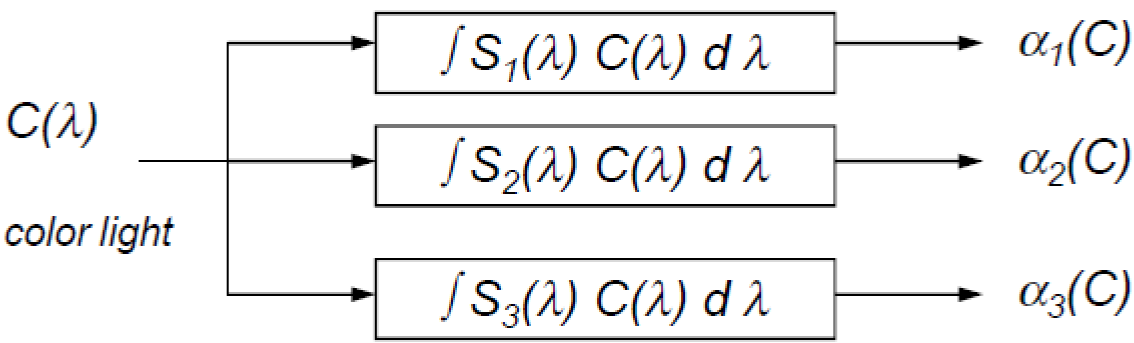
RGB add together to get white

White subtracted by CMY to get Black

© 2017 NUS. All Rights Reserved.

Representation b by Three Primary Colors

- Any color can be reproduced by mixing an appropriate set of three primary colors (Thomas Young, 1802)
- Three types of cones in human retina
  - Absorption response *Si(l)* has peaks around 450nm (blue), 550nm (green), 620nm (yellow-green)
  - Color sensation depends on the spectral response {*a1(C), a2(C), a3(C)* } rather than the complete light spectrum *C(l)*



Identically perceived colors if $\alpha_i(C_1) = \alpha_i(C_2)$

© 2017 NUS. All Rights Reserved.

Example: Seeing Yellow Without Yellow

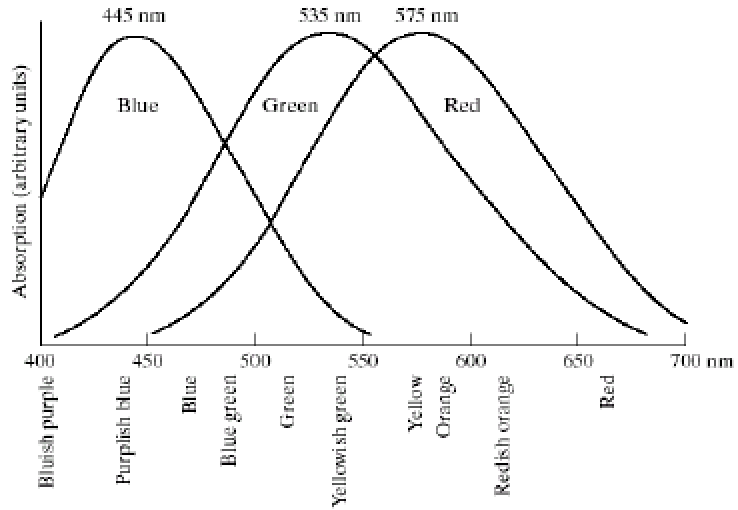
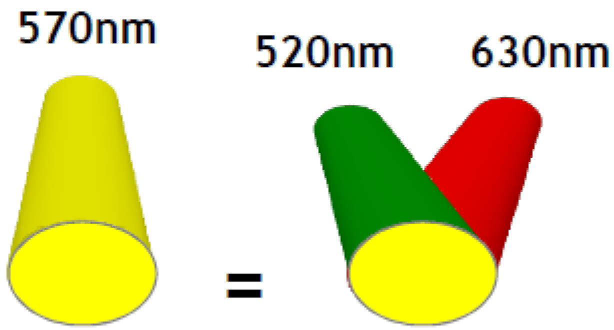


FIGURE 6.3 Absorption of light by the red, green, and blue cones in the human eye as a function of wavelength.



mix green and red light to obtain perception of yellow, without shining a single yellow photon

© 2017 NUS. All Rights Reserved.

Color Characterization

Hue: dominant color corresponding to a dominant wavelength of mixture light wave

Saturation: Relative purity or amount of white light mixed with a hue (inversely proportional to amount of white light added)

Brightness: Intensity

- Hue
- Saturation

Chromaticity

amount of red (X), green (Y) and blue (Z) to form any particular color is called tristimulus.

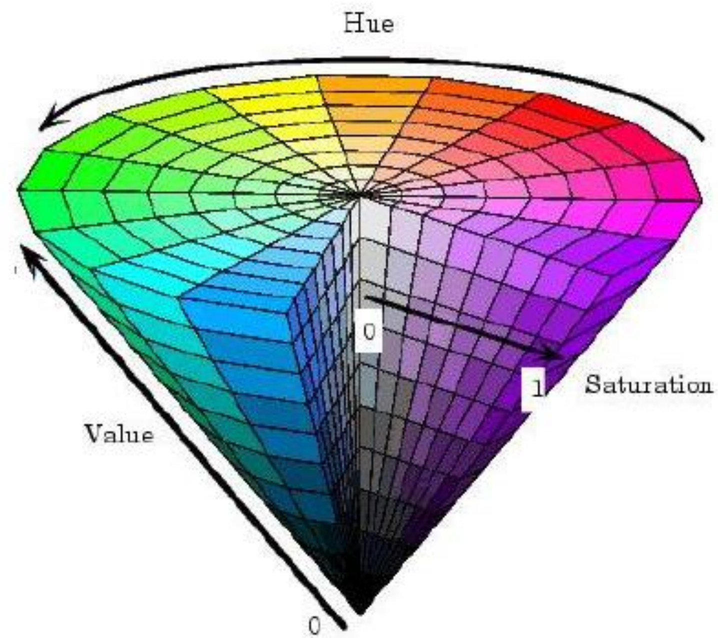
© 2017 NUS. All Rights Reserved.

Perceptual Attributes of Color

- Value of Brightness

(perceived luminance)

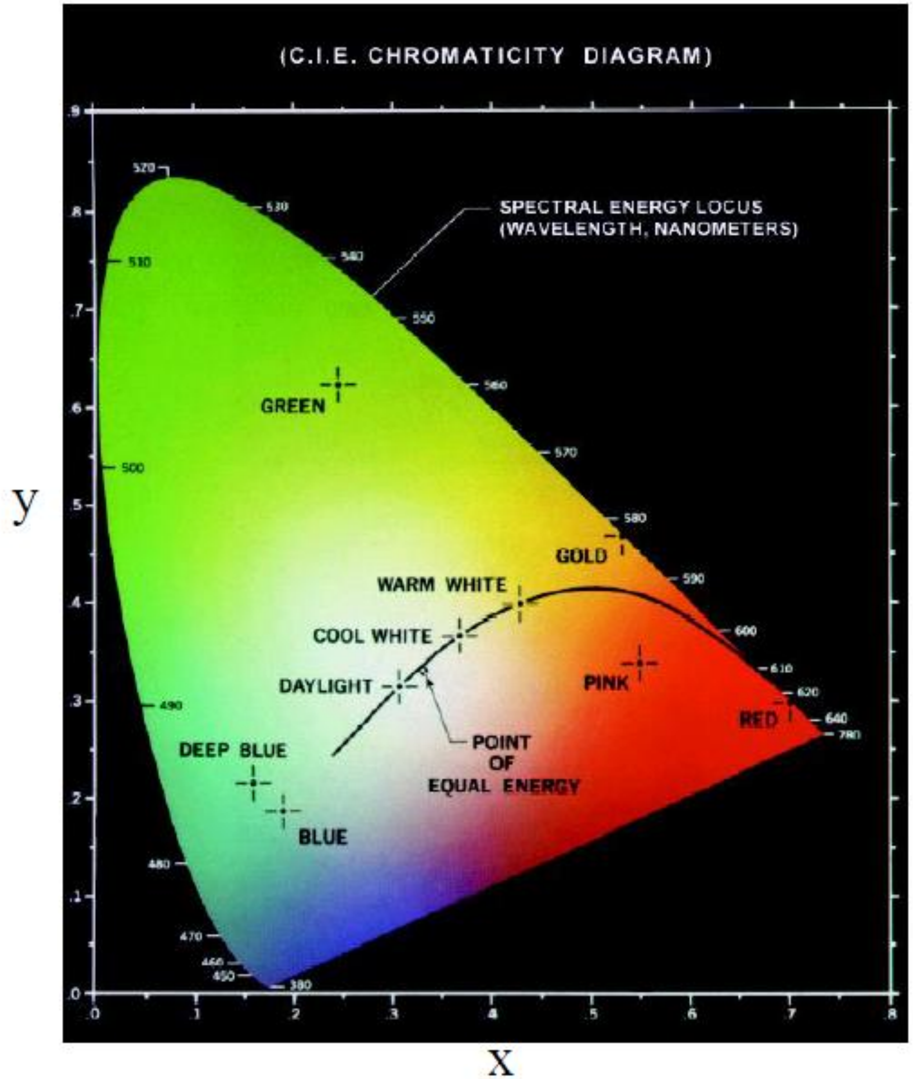
- Chrominance
  - Hue
    - specify color tone (redness, greenness, etc.)
    - depend on peak wavelength
  - Saturation
    - describe how pure the color is
    - depend on the spread (bandwidth) of light spectrum
    - reflect how much white light is added



- RGB óHSV Conversion ~ nonlinear

© 2017 NUS. All Rights Reserved.

CIE Chromaticity Diagram



Trichromatic coefficients:

$$x = \frac{X}{X+Y+Z}$$

$$y = \frac{Y}{X+Y+Z}$$

$$z = \frac{Z}{X+Y+Z}$$

$$x + y + z = 1$$

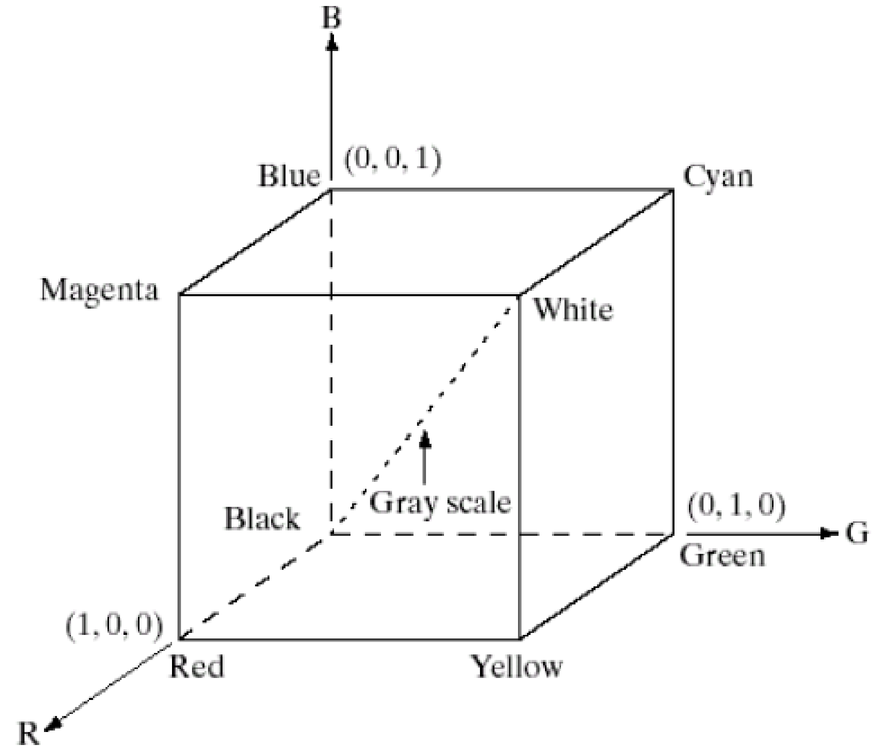
Points on the boundary are fully saturated colors

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

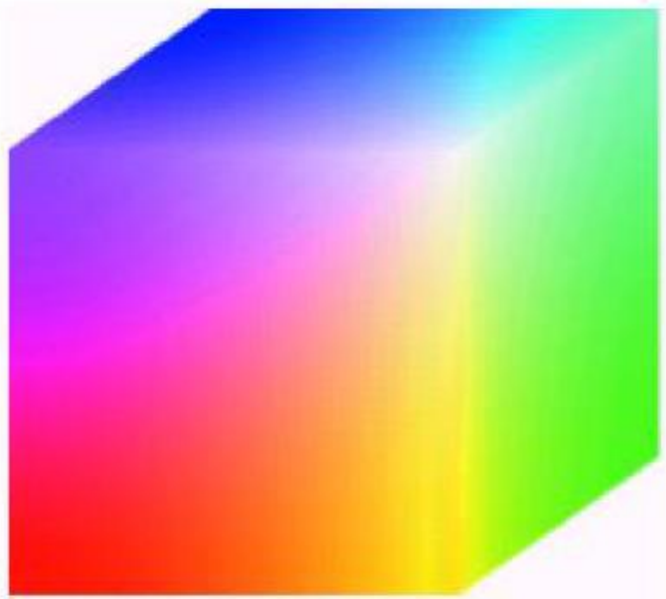
RGB Color Model

- Purpose of color models: to facilitate the specification of colors in some standard
- RGB color models:
  - based on Cartesian coordinate system



© 2017 NUS. All Rights Reserved.

RGB Color Cube



$$\left.\begin{array}{l} R = 8\text{ bits} \\ G = 8\text{ bits} \\ B = 8\text{ bits} \end{array}\right\} \text{Color depth 24 bits} = 16777216\text{ colors}$$

$(R = 0)$ $(G = 0)$ $(B = 0)$

Hidden faces of the cube

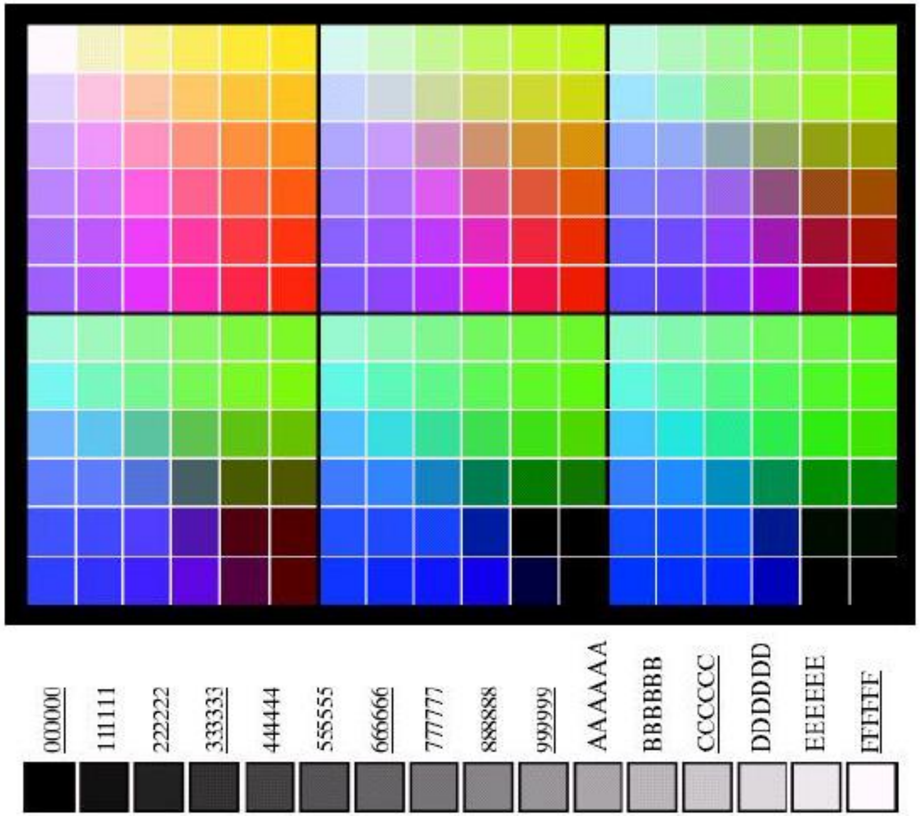
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Safe RGB Colors

Safe RGB colors: a subset of RGB colors.

There are 216 colors common in most operating systems.



a
b

FIGURE 6.10 (a) The 216 safe RGB colors. (b) All the grays in the 256-color RGB system (grays that are part of the safe color group are shown underlined).

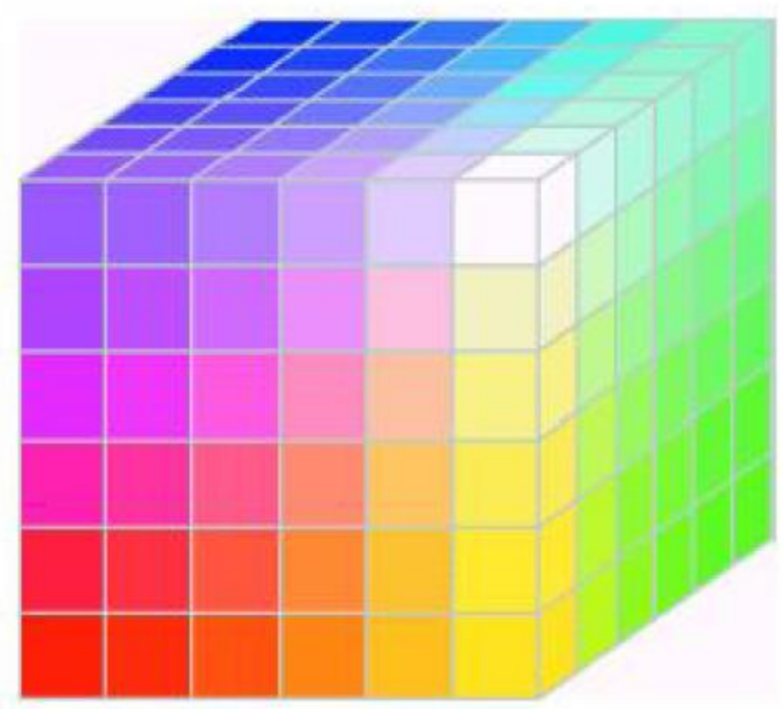
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

RGB Safe-color Cube

Number System	Color Equivalents					
Hex	00	33	66	99	CC	FF
Decimal	0	51	102	153	204	255

TABLE 6.1
Valid values of each RGB component in a safe color.



The RGB Cube is divided into 6 intervals on each axis to achieve the total $6^3 = 216$ common colors.

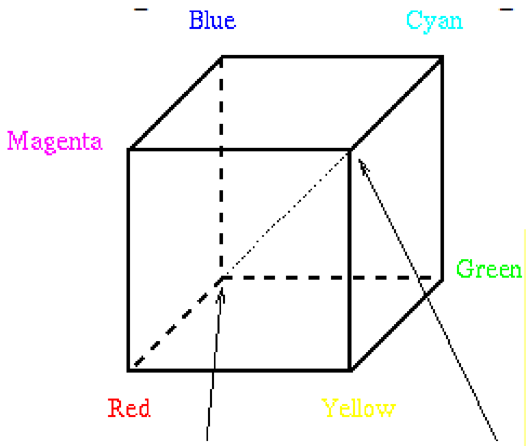
However, for 8 bit color representation, there are the total 256 colors. Therefore, the remaining 40 colors are left to OS.

© 2017 NUS. All Rights Reserved.

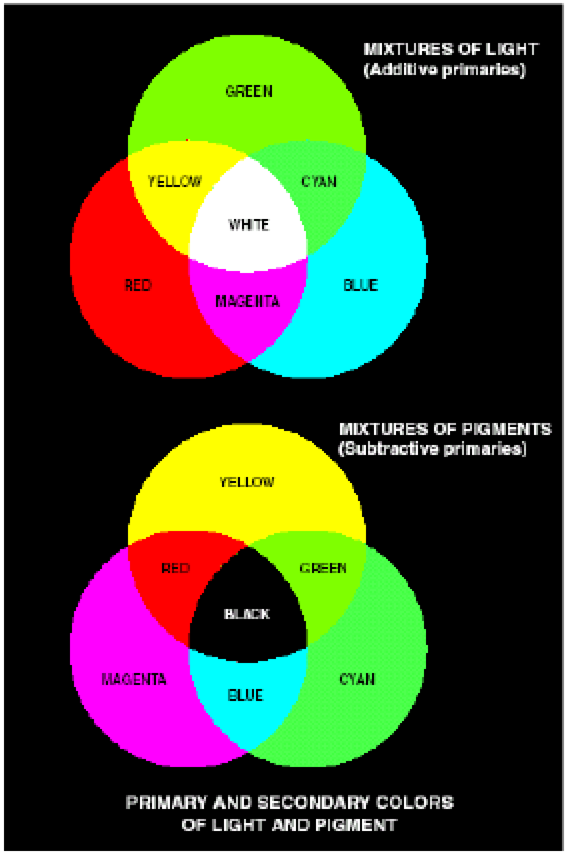
CMY and CMYK Color Models

C = Cyan
M = Magenta
Y = Yellow
K = Black

- Primary colors for pigment
  - Defined as one that subtracts/absorbs a primary color of light & reflects the other two
- CMY – Cyan, Magenta, Yellow
  - Complementary to RGB
  - Proper mix of them produces black



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



© 2017 NUS. All Rights Reserved.

HSI Color Model

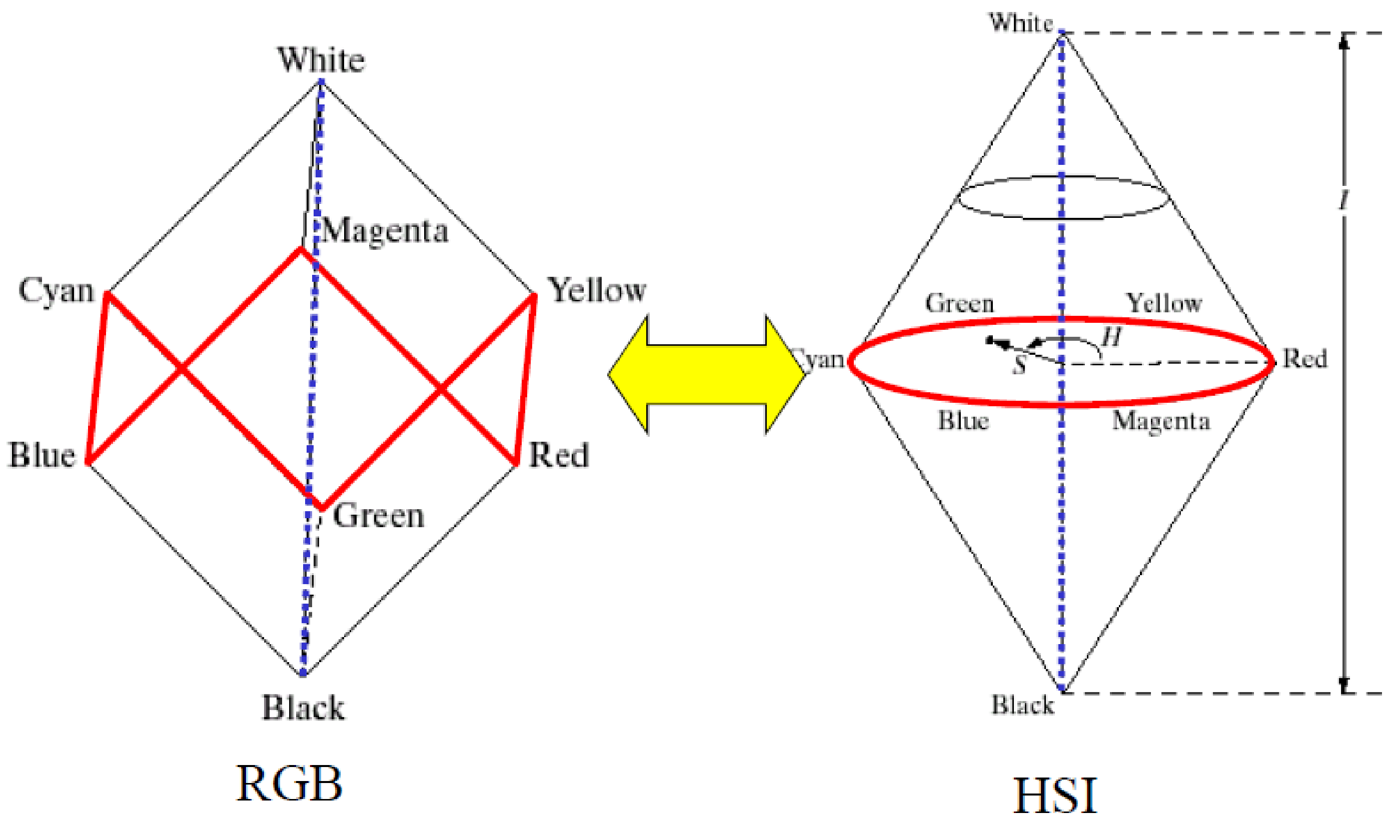
RGB, CMY models are not good for human interpreting

- HSI Color model:
  - Hue: Dominant color
  - Saturation: Relative purity (inversely proportional to amount of white light added)
  - Intensity: Brightness

Color carrying information

© 2017 NUS. All Rights Reserved.

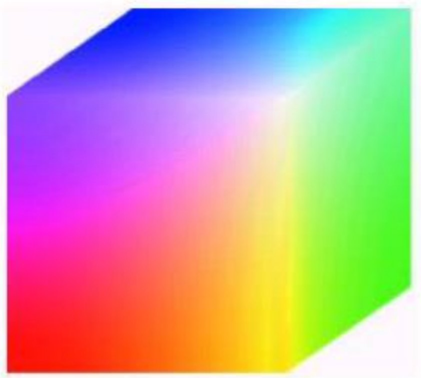
Relationship Between RGB and HSI Color Models



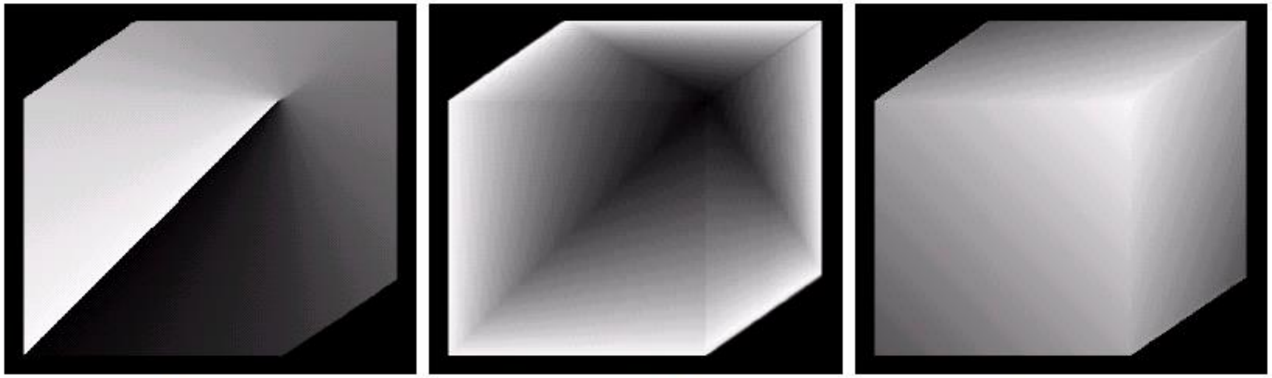
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Example: HSI Components of RGB Cube



RGB Cube



Hue Saturation Intensity

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Converting Co Colors from RGB to HSI

$$H = \begin{cases} \theta & \text{if } B \le G \\ 360 - \theta & \text{if } B > G \end{cases}$$

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

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

$$I = \frac{1}{3}(R+G+B)$$

© 2017 NUS. All Rights Reserved.

Converting Co Colors from HSI to RGB

RG sector: $0 \le H < 120$

$$R = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right]$$

$$B = I(1 - S)$$

$$G = 1 - (R + B)$$

GB sector: $120 \le H < 240$

$$H = H - 120$$

$$R = I(1 - S)$$

$$G = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right]$$

$$B = 1 - (R + G)$$

BR sector: $240 \le H \le 360$

$$H = H - 240$$

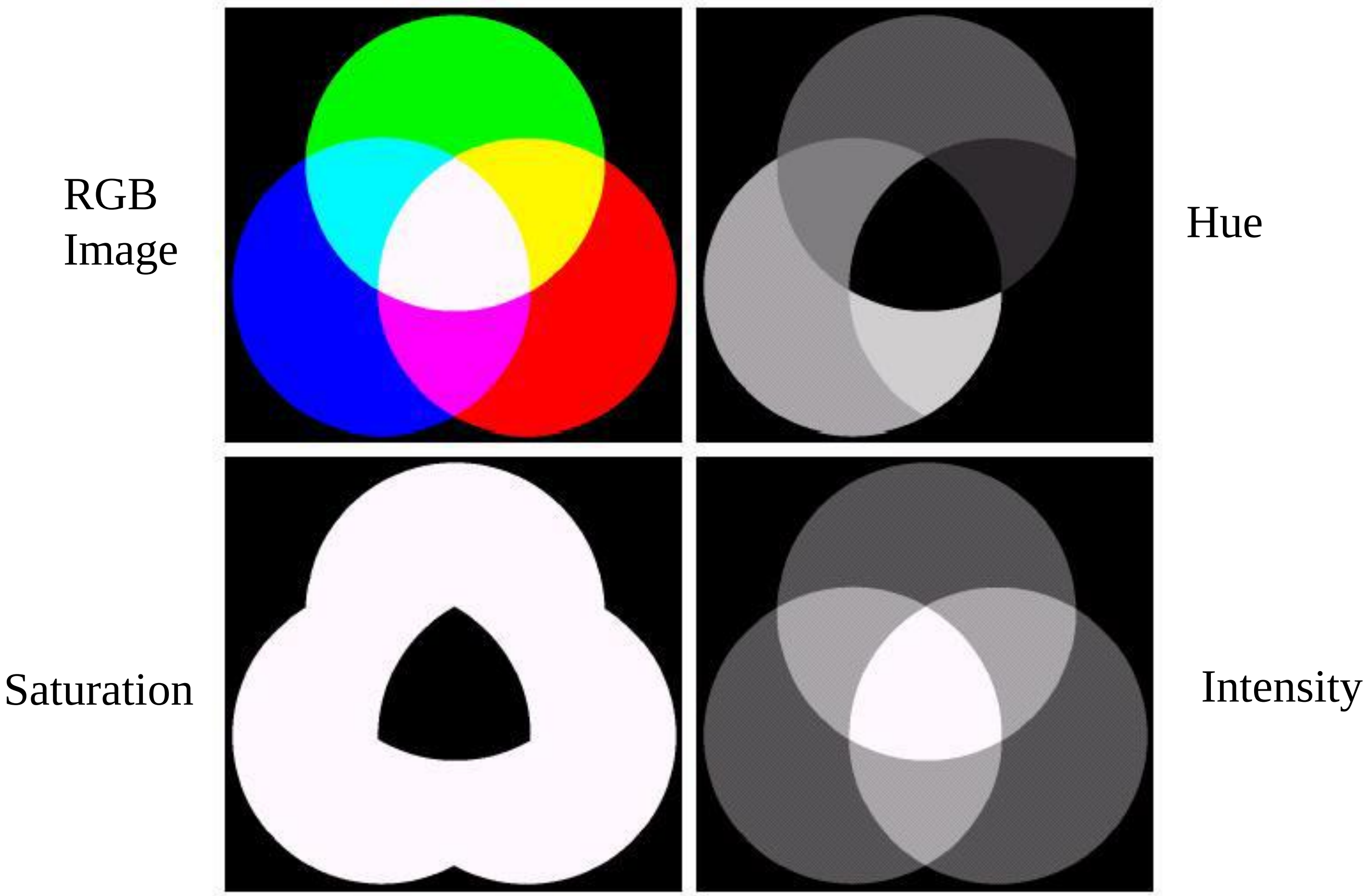
$$B = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right]$$

$$G = I(1 - S)$$

$$R = 1 - (G + B)$$

© 2017 NUS. All Rights Reserved.

Example: HSI Components of RGB Colors



© 2017 NUS. All Rights Reserved.

Color Coordinates Used in TV Transmission

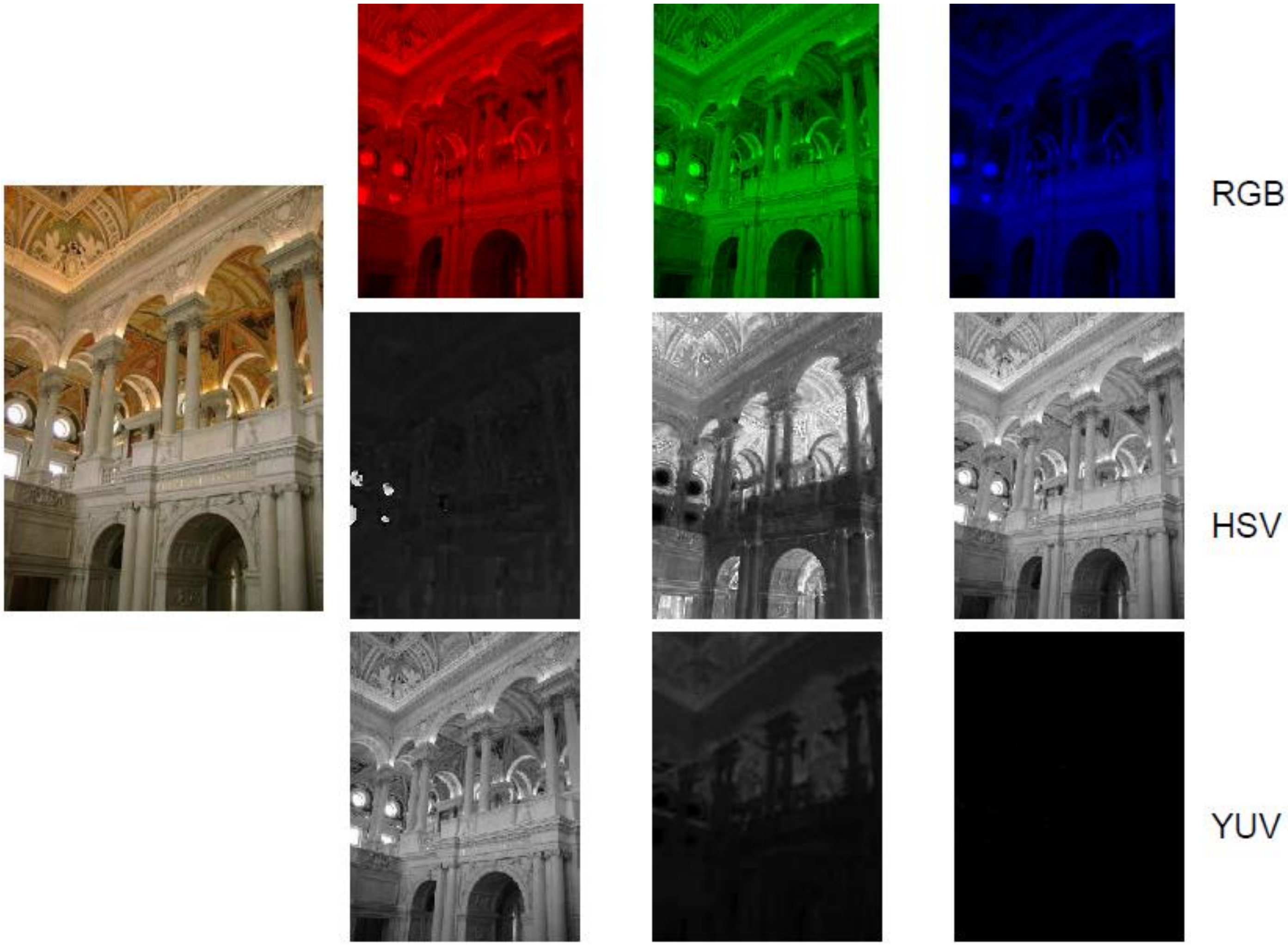
- Facilitate sending color video via 6MHz mono TV channel
- YIQ for NTSC (National Television Systems Committee) transmission system
  - Use receiver primary system (RN, GN, BN) as TV receivers standard
  - Transmission system use (Y, I, Q) color coordinate
    - Y ~ luminance, I & Q ~ chrominance
    - I & Q are transmitted in through orthogonal carriers at the same freq.

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R_N \\ G_N \\ B_N \end{bmatrix}. \qquad \begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.147 & -0.289 & 0.436 \\ 0.615 & -0.515 & -0.100 \end{bmatrix} \begin{bmatrix} R_P \\ G_P \\ B_P \end{bmatrix}.$$

- YUV (YCbCr) for PAL and digital video
  - Y ~ luminance, Cb and Cr ~ chrominance

© 2017 NUS. All Rights Reserved.

Examples



© 2017 NUS. All Rights Reserved.

Color Image Processing

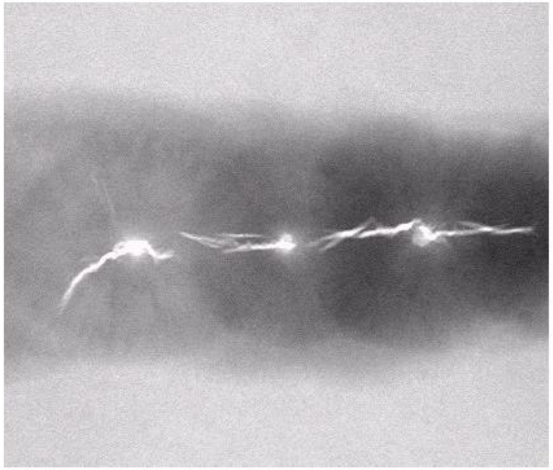
- There are 2 types of color image processes:
  - Pseudocolor image process: Assigning colors to gray values based on a specific criterion. Gray scale images to be processed may be a single image or multiple images such as multispectral images
  - Full color image process: The process to manipulate real color images such as color photographs.

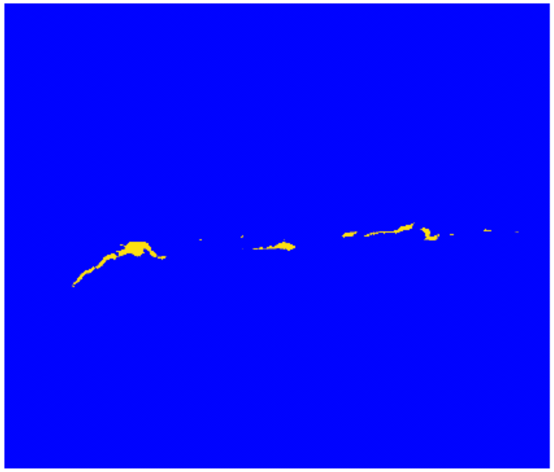
© 2017 NUS. All Rights Reserved.

Pseudocolor Image Processing

- Pseudo color = false color : In some case there is no “color” concept for a gray scale image but we can assign “false” colors to an image.

- Why we need to assign colors to gray scale image?
- Answer: Human can distinguish different colors better than different shades of gray.





(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

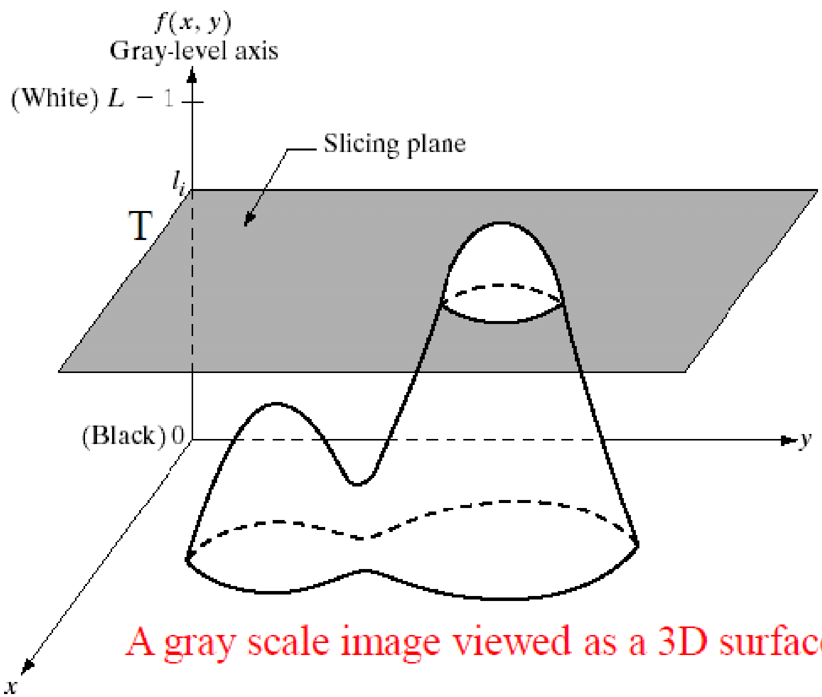
© 2017 NUS. All Rights Reserved.

Intensity Slicing or Density Slicing

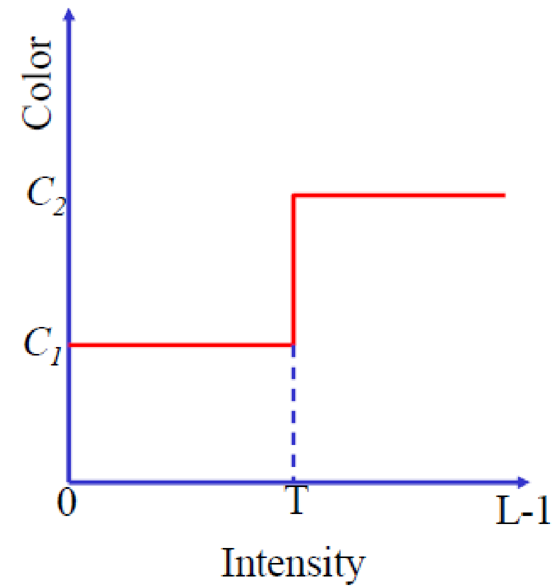
Formula:

$$g(x,y) = \begin{cases} C_1 & \text{if } f(x,y) \le T \\ C_2 & \text{if } f(x,y) > T \end{cases}$$

C_1 = Color No. 1
C_2 = Color No. 2

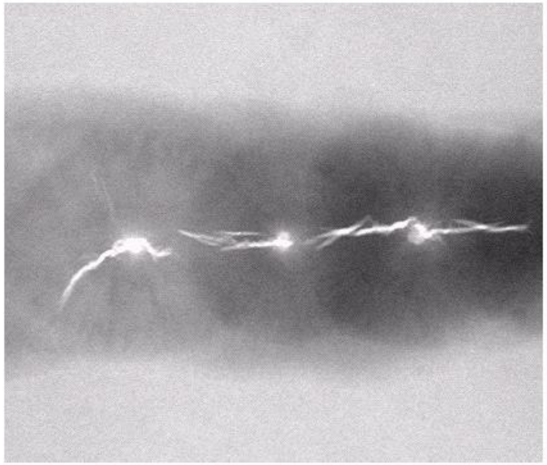


A gray scale image viewed as a 3D surface.

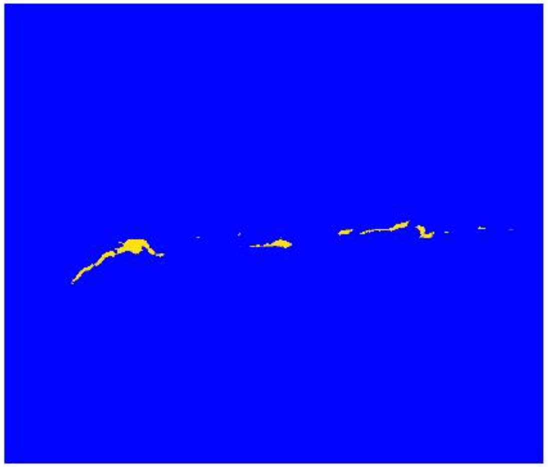


© 2017 NUS. All Rights Reserved.

Intensity Slicing Example



An X-ray image of a weld with cracks



After assigning a yellow color to pixels with value 255 and a blue color to all other pixels.

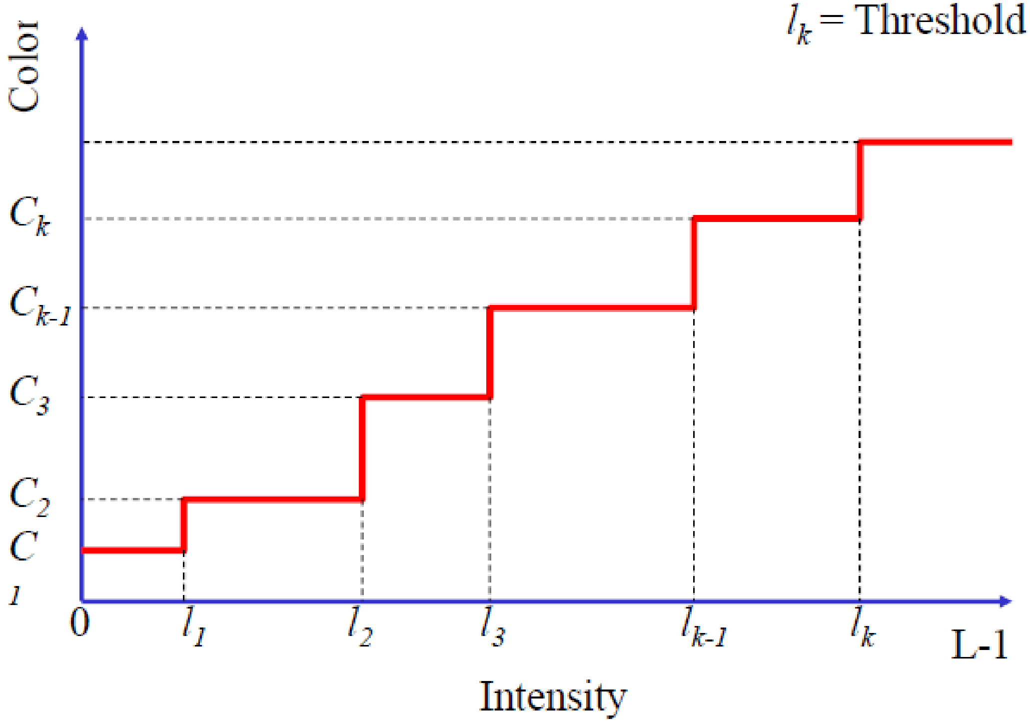
© 2017 NUS. All Rights Reserved.

Multi Level Intensity Slicing

$$g(x,y) = C_k \qquad \text{for } l_{k-1} < f(x,y) \le l_k$$

grey level:	0–63	64–127	128–191	192–255
colour:	blue	magenta	green	red

C_k = Color No. k
l_k = Threshold level k

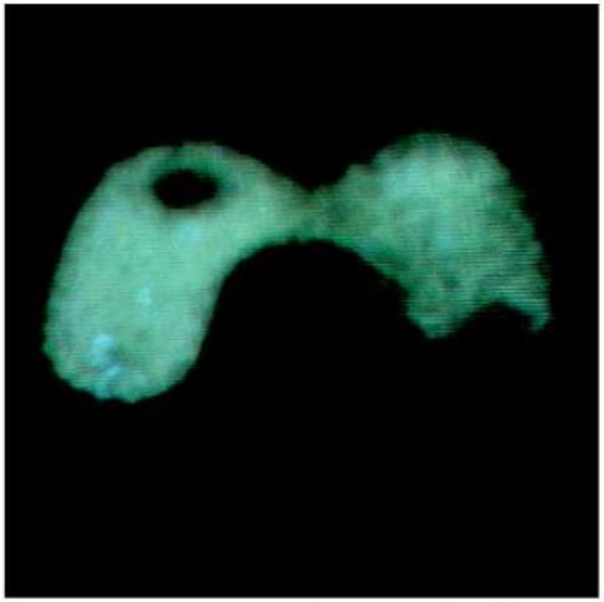


© 2017 NUS. All Rights Reserved.

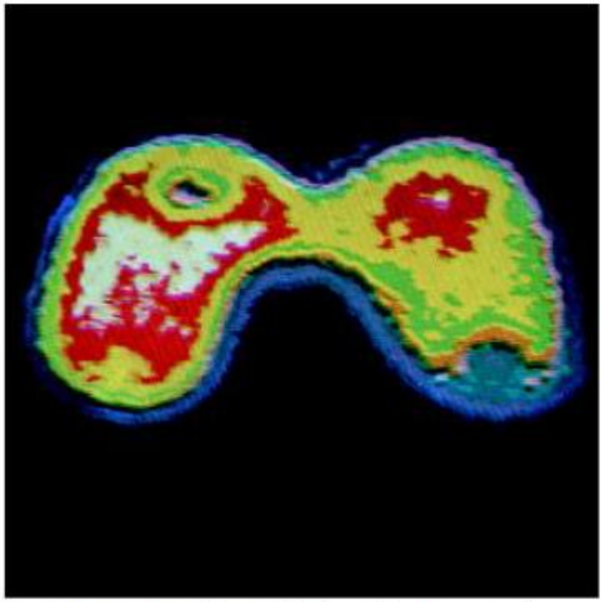
Multi Level Intensity Slicing Example

$$g(x,y) = C_k \qquad \text{for } l_{k-1} < f(x,y) \le l_k$$

C_k = Color No. k
l_k = Threshold level k



An X-ray image of the Picker Thyroid Phantom.

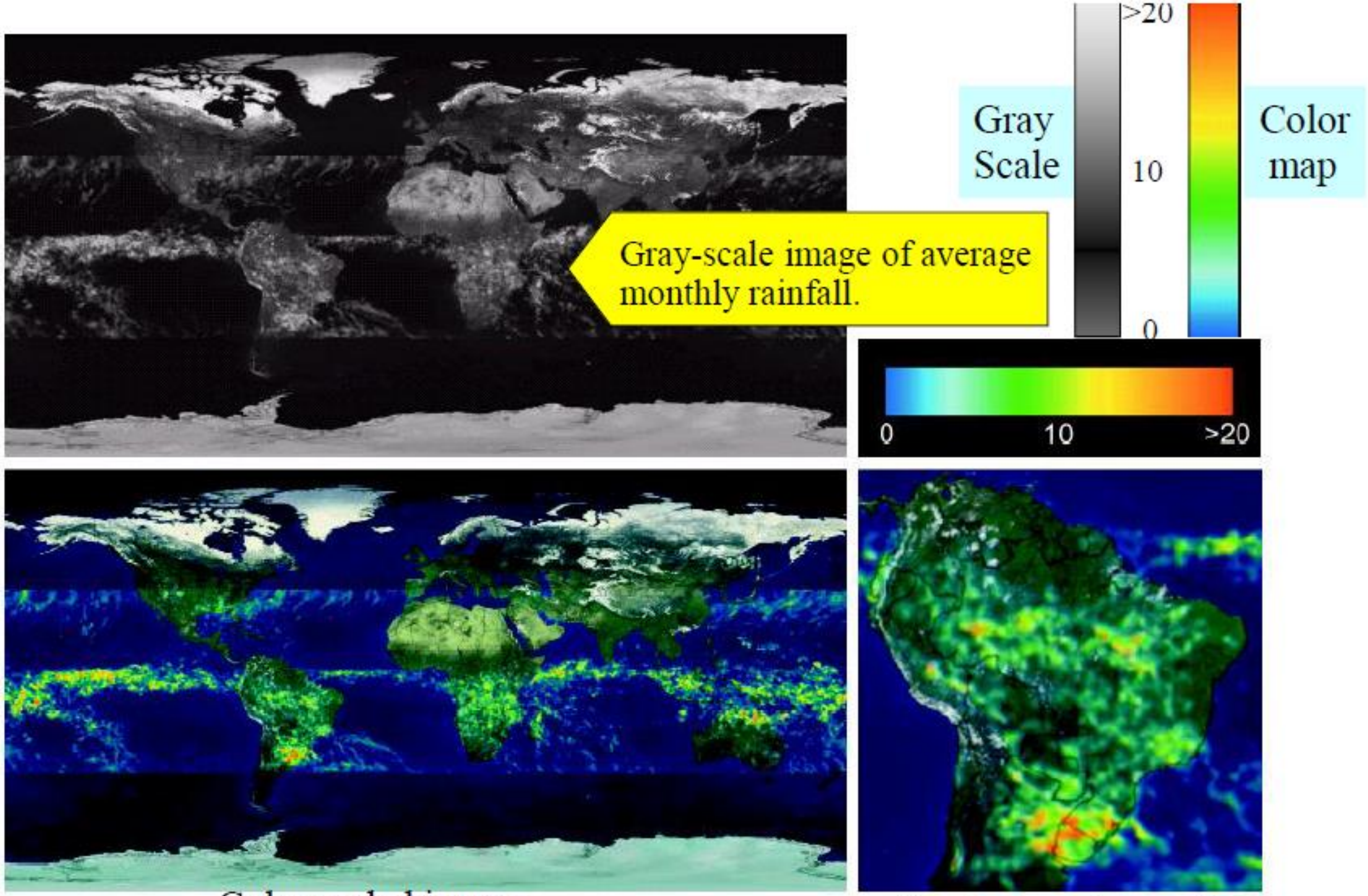


After density slicing into 8 colors

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Coding Example



Color coded image

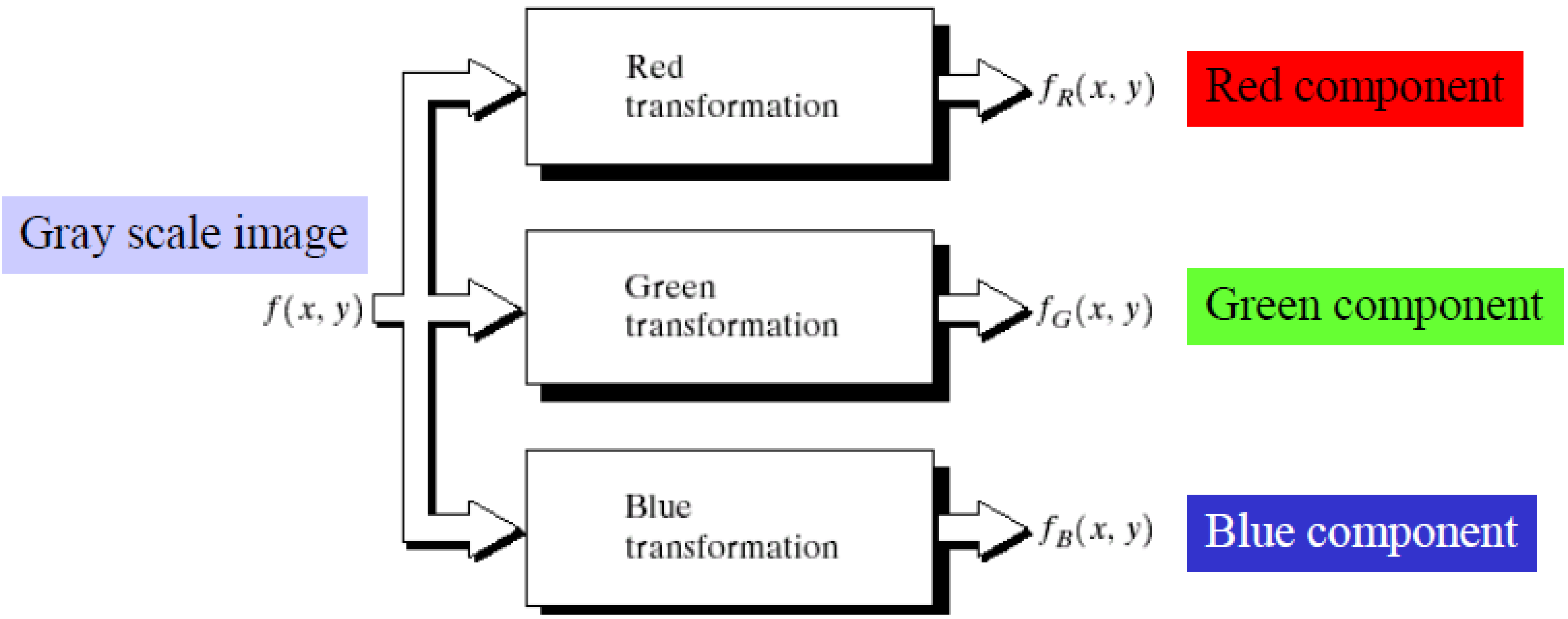
South America region

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Gray Level to Color Transformation

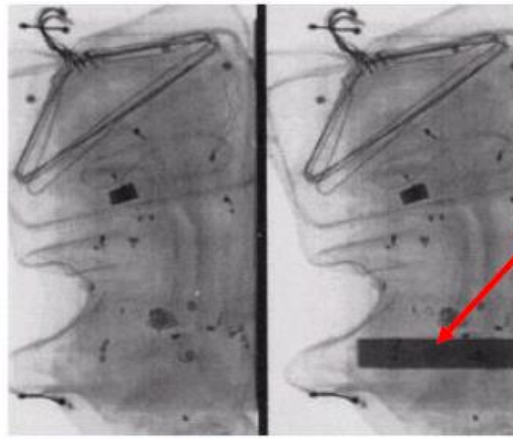
Assigning colors to gray levels based on specific mapping functions



© 2017 NUS. All Rights Reserved.

Gray Level to Color Transformation Example

An X-ray image of a garment bag

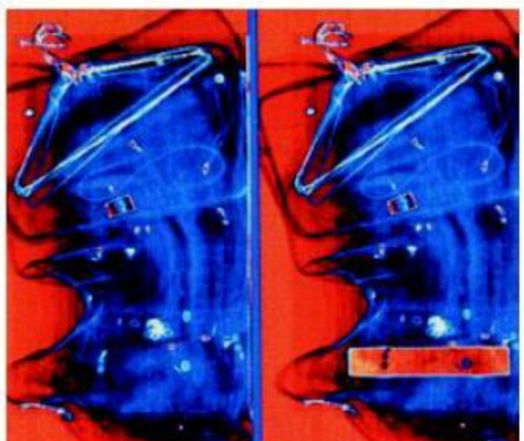


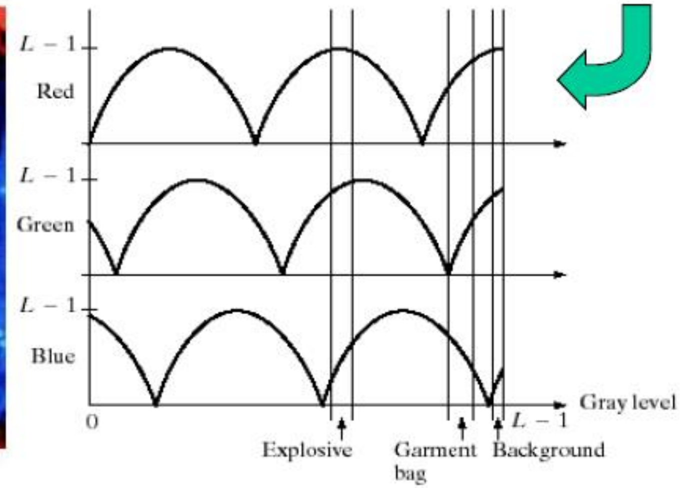
An X-ray image of a garment bag with a simulated explosive device

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Transformations

Color coded images





© 2017 NUS. All Rights Reserved.

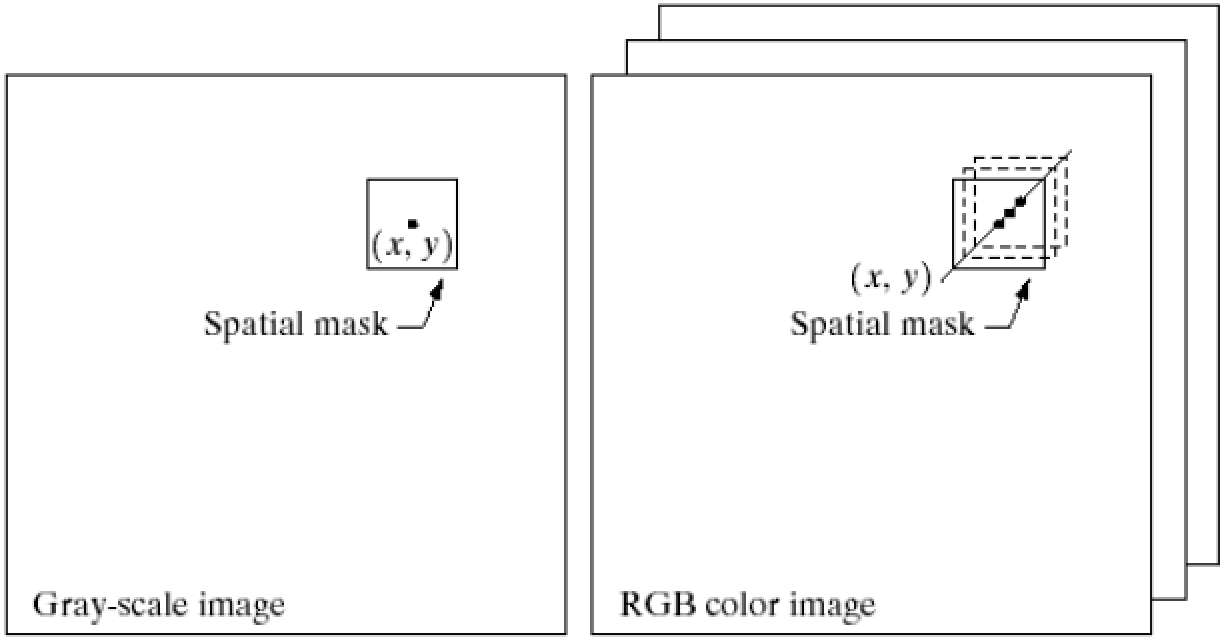
Basics of Full-Color Image Processing

2 Methods:

- Per-color-component processing: process each component separately.
- Vector processing: treat each pixel as a vector to be processed.

Example of per-color-component processing: smoothing an image

By smoothing each RGB component separately.



(Images from Rafael C. Gonzalez and Richard E.
Wood, Digital Image Processing, 2 Edition.

© 2017 NUS. All Rights Reserved.

Example: Full-Color Image and Various Color Space Components

Color image

CMYK components

RGB components

HSI components

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Transformation

Use to transform colors to colors.

Formulation:

$$g(x,y) = T\left[f(x,y)\right]$$

$f(x,y)$ = input color image, $g(x,y)$ = output color image
T = operation on f over a spatial neighborhood of (x,y)

When only data at one pixel is used in the transformation, we can express the transformation as:

$$s_i = T_i(r_1, r_2, \cdots, r_n) \qquad i = 1, 2, \ldots, n$$

Where r_i = color component of $f(x,y)$
s_i = color component of $g(x,y)$

For RGB images, $n = 3$

© 2017 NUS. All Rights Reserved.

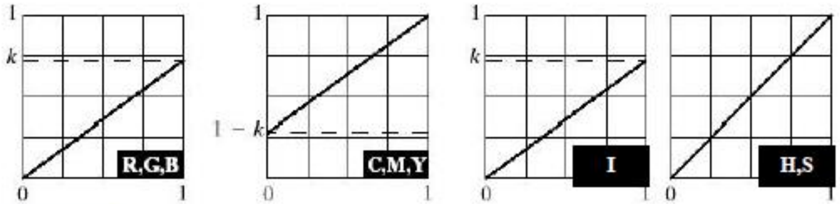
Example: Color Transformation

Formula for RGB:

$$s_R(x,y) = kr_R(x,y)$$
$$s_G(x,y) = kr_G(x,y)$$
$$s_B(x,y) = kr_B(x,y)$$

Formula for HSI:

$$s_I(x,y) = kr_I(x,y)$$



Formula for CMY:

$$s_C(x,y) = kr_C(x,y) + (1-k)$$
$$s_M(x,y) = kr_M(x,y) + (1-k)$$
$$s_Y(x,y) = kr_Y(x,y) + (1-k)$$

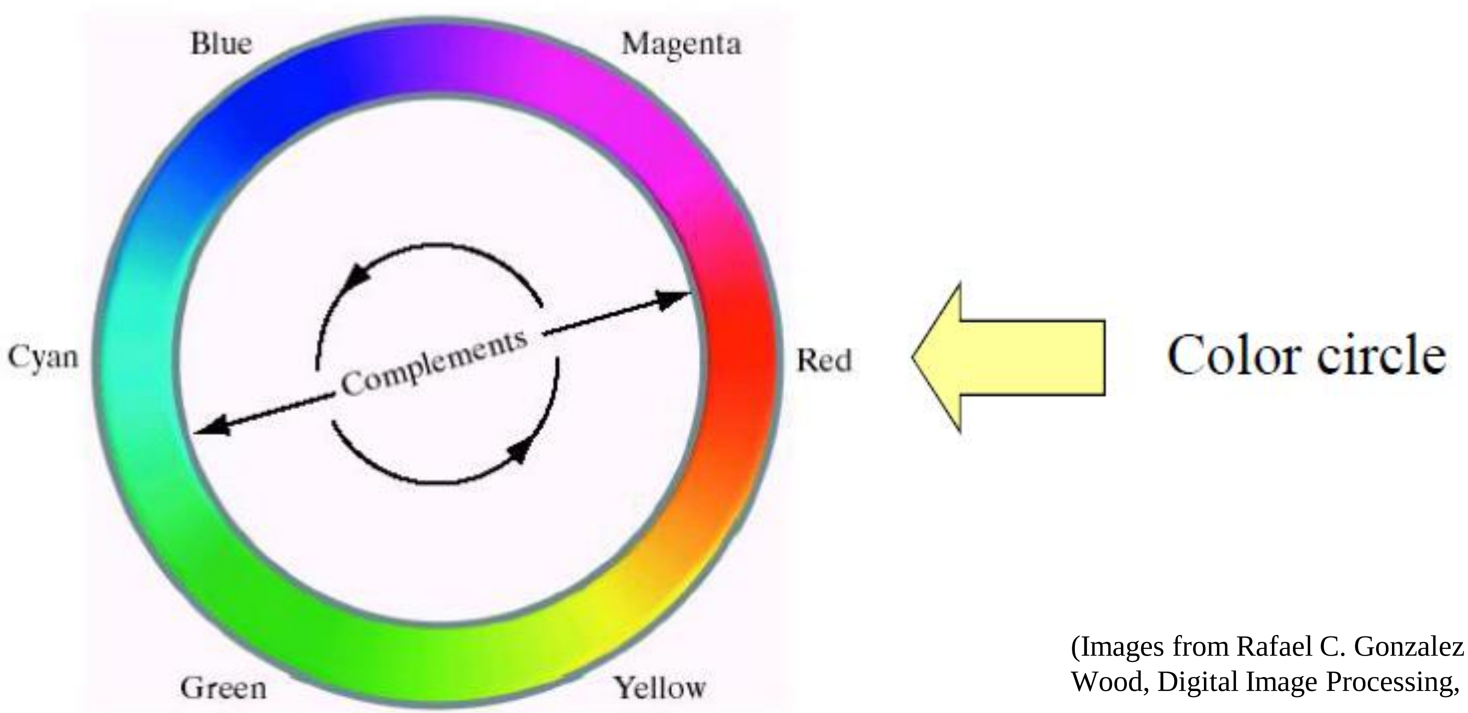
These 3 transformations give the same results.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Complements

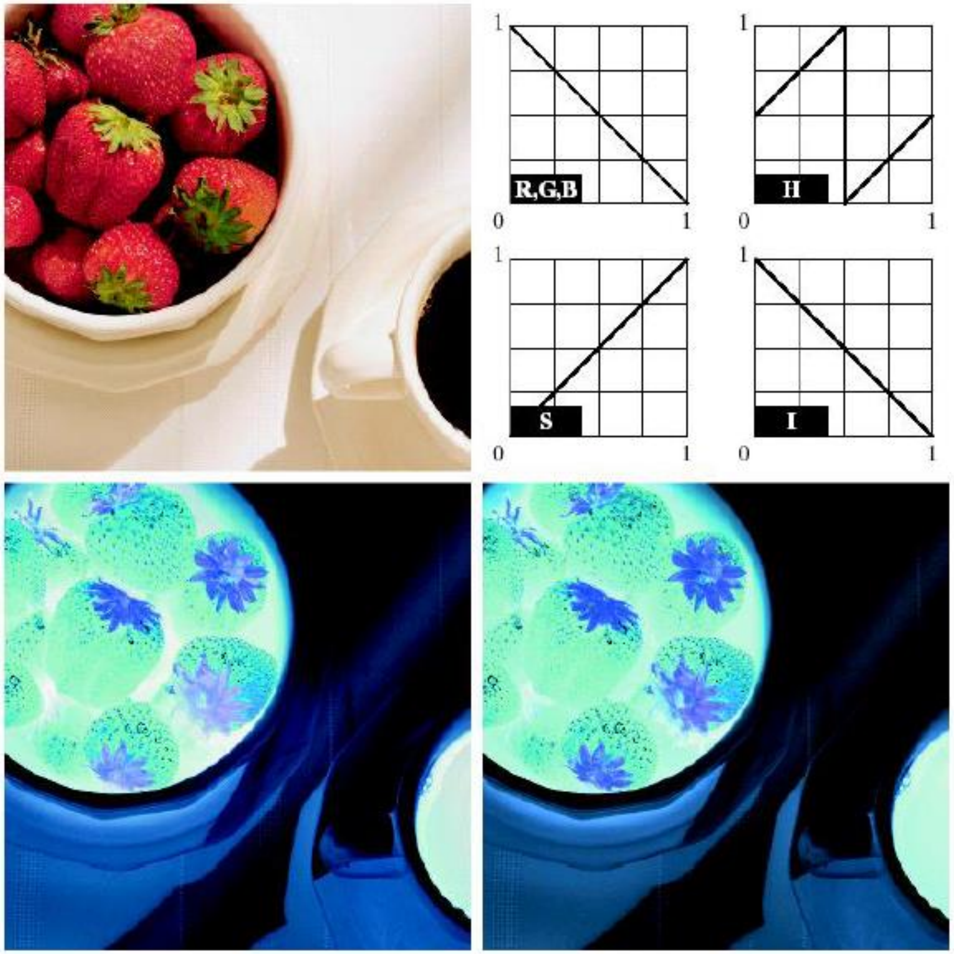
Color complement replaces each color with its opposite color in the color circle of the Hue component. This operation is analogous to image negative in a gray scale image.



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Complement Transformation Example



a b
c d

FIGURE 6.33 Color complement transformations. (a) Original image. (b) Complement transformation functions. (c) Complement of (a) based on the RGB mapping functions. (d) An approximation of the RGB complement using HSI transformations.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Slicing Transformation

We can perform "slicing" in color space: if the color of each pixel is far from a desired color more than threshold distance, we set that color to some specific color such as gray, otherwise we keep the original color unchanged.

$$s_i = \begin{cases} 0.5 & \text{if } \left[\left| r_j - a_j \right| > \dfrac{W}{2} \right]_{any\ 1 \le j \le n} \\ r_i & \text{otherwise} \end{cases}$$

Set to gray

Keep the original color

$i = 1, 2, \ldots, n$

or

$$s_i = \begin{cases} 0.5 & \text{if } \sum_{j=1}^{n} \left(r_j - a_j \right)^2 > R_0^2 \\ r_i & \text{otherwise} \end{cases}$$

Set to gray

Keep the original color

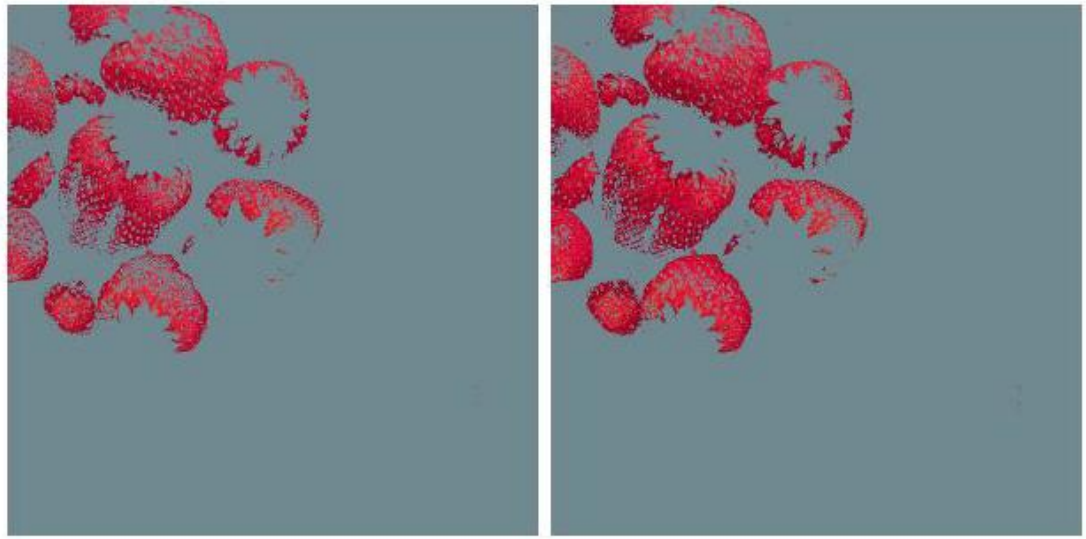
$i = 1, 2, \ldots, n$

© 2017 NUS. All Rights Reserved.

Color Slicing Transformation Example

After color slicing

Original image



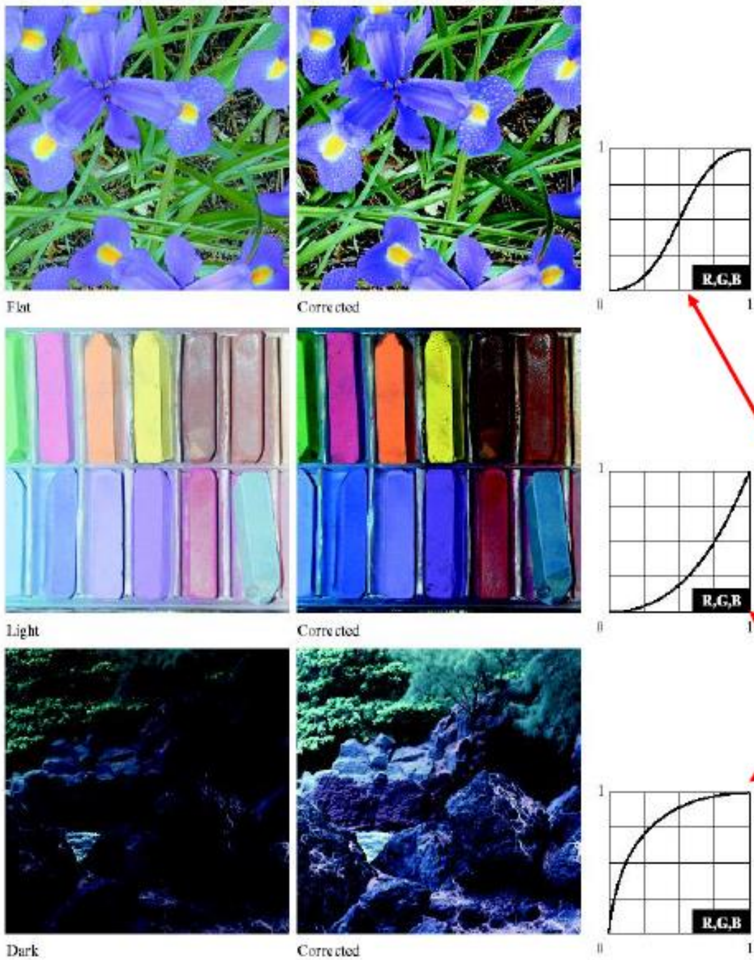
a b

FIGURE 6.34 Color slicing transformations that detect (a) reds within an RGB cube of width $W = 0.2549$ centered at (0.6863, 0.1608, 0.1922), and (b) reds within an RGB sphere of radius 0.1765 centered at the same point. Pixels outside the cube and sphere were replaced by color (0.5, 0.5, 0.5).

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Tonal Correction Examples



In these examples, only brightness and contrast are adjusted while keeping color unchanged.

This can be done by using the same transformation for all RGB components.

Contrast enhancement

Power law transformations

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Histogram Equalization of a Full-Color Image

- Histogram equalization of a color image can be performed by adjusting color intensity uniformly while leaving color unchanged.
- The HSI model is suitable for histogram equalization where only Intensity (I) component is equalized.

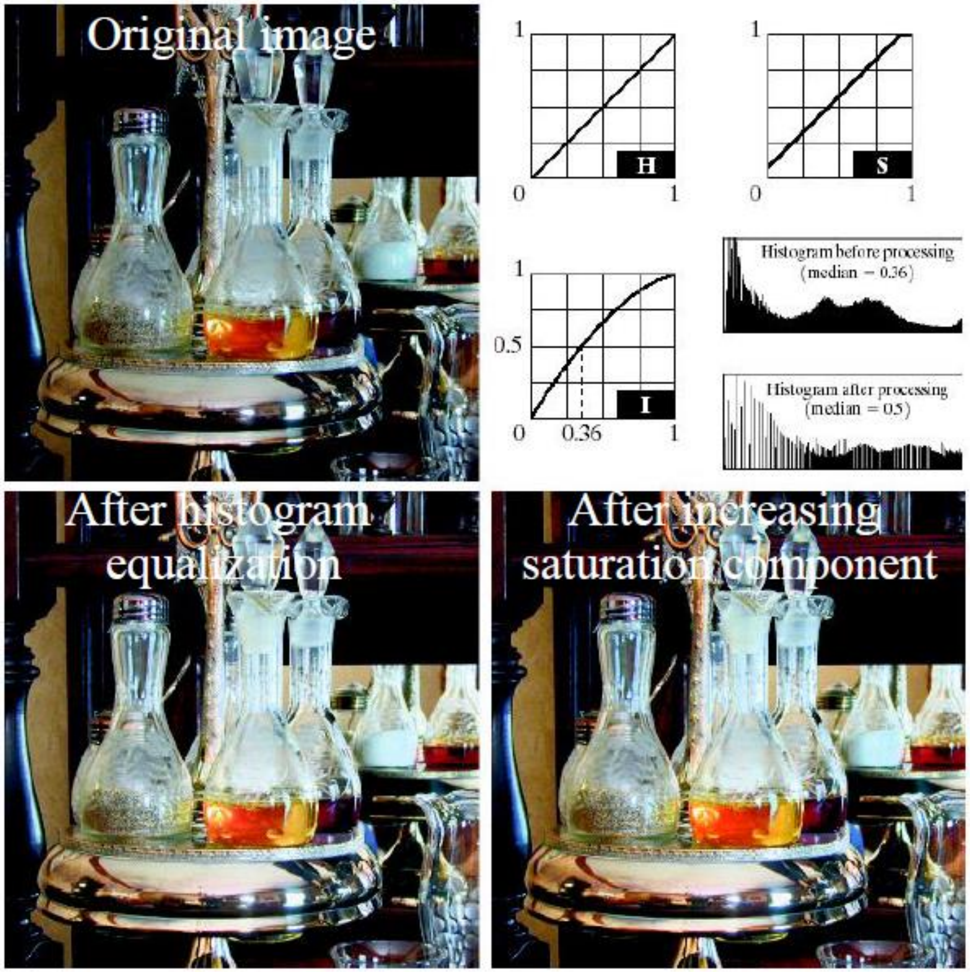
$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j)$$

$$= \sum_{j=0}^{k} \frac{n_j}{N}$$

- where r and s are intensity components of input and output color image.

© 2017 NUS. All Rights Reserved.

Histogram Equalization of a Full-Color Image



a b
c d

FIGURE 6.37 Histogram equalization (followed by saturation adjustment) in the HSI color space.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Image Smoothing

2 Methods:

- Per-color-plane method: for RGB, CMY color models

  Smooth each color plane using moving averaging and the combine back to RGB

$$\bar{\mathbf{c}}(x,y) = \frac{1}{K}\sum_{(x,y)\in S_{xy}}\mathbf{c}(x,y) = \begin{bmatrix} \frac{1}{K}\sum_{(x,y)\in S_{xy}} R(x,y) \\ \frac{1}{K}\sum_{(x,y)\in S_{xy}} G(x,y) \\ \frac{1}{K}\sum_{(x,y)\in S_{xy}} B(x,y) \end{bmatrix}$$

- Smooth only Intensity component of a HSI image while leaving H and S unmodified.

  Note: 2 methods are not equivalent.

© 2017 NUS. All Rights Reserved.

Color Image Sharpening

We can do in the same manner as color image smoothing:

- Per-color-plane method for RGB,CMY images
- Sharpening only I component of a HSI image

Sharpening all RGB components

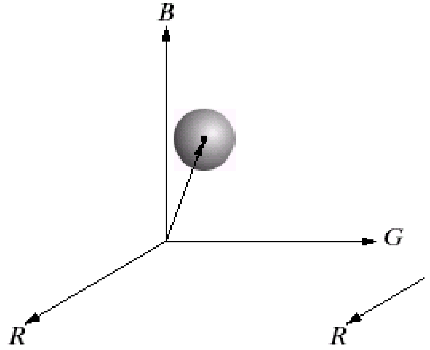
Sharpening only I component of HSI

© 2017 NUS. All Rights Reserved.

Color Segmentation

2 Methods:

- Segmented in HSI color space:
  - A thresholding function based on color information in H and S Components. We rarely use I component for color image segmentation.
- Segmentation in RGB vector space:
  - A thresholding function based on distance in a color vector space.

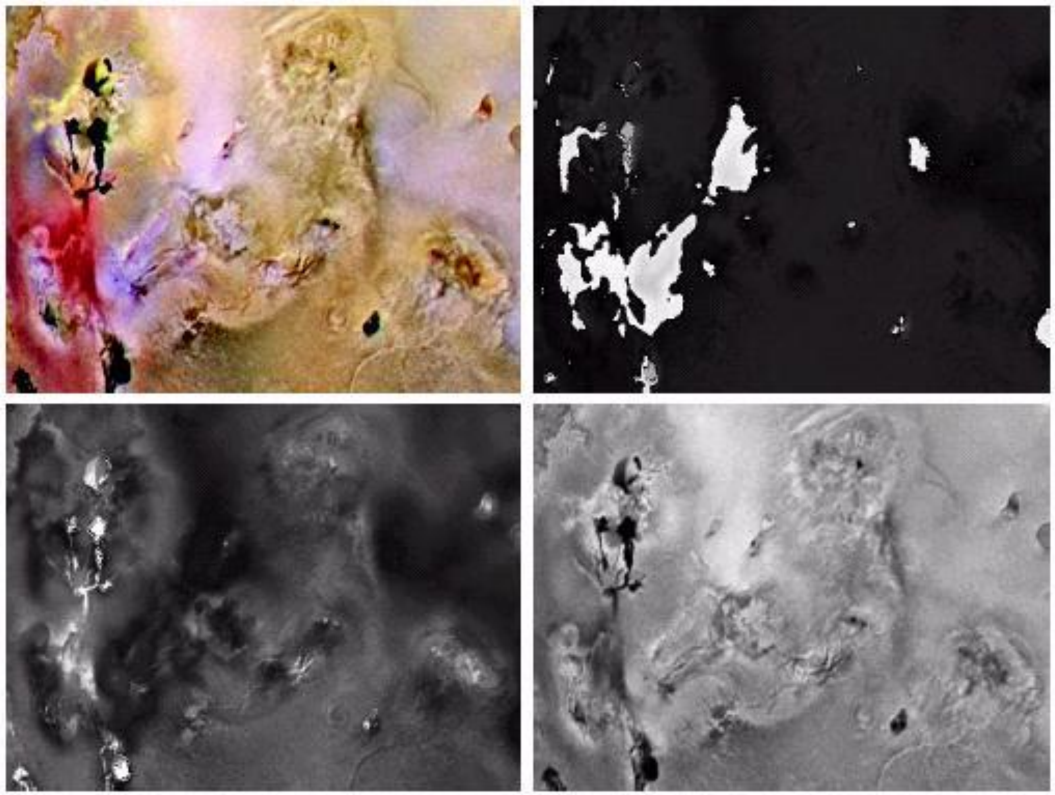


(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition

© 2017 NUS. All Rights Reserved.

Color Segmentation in HSI Color Space

Color image Hue



Saturation Intensity

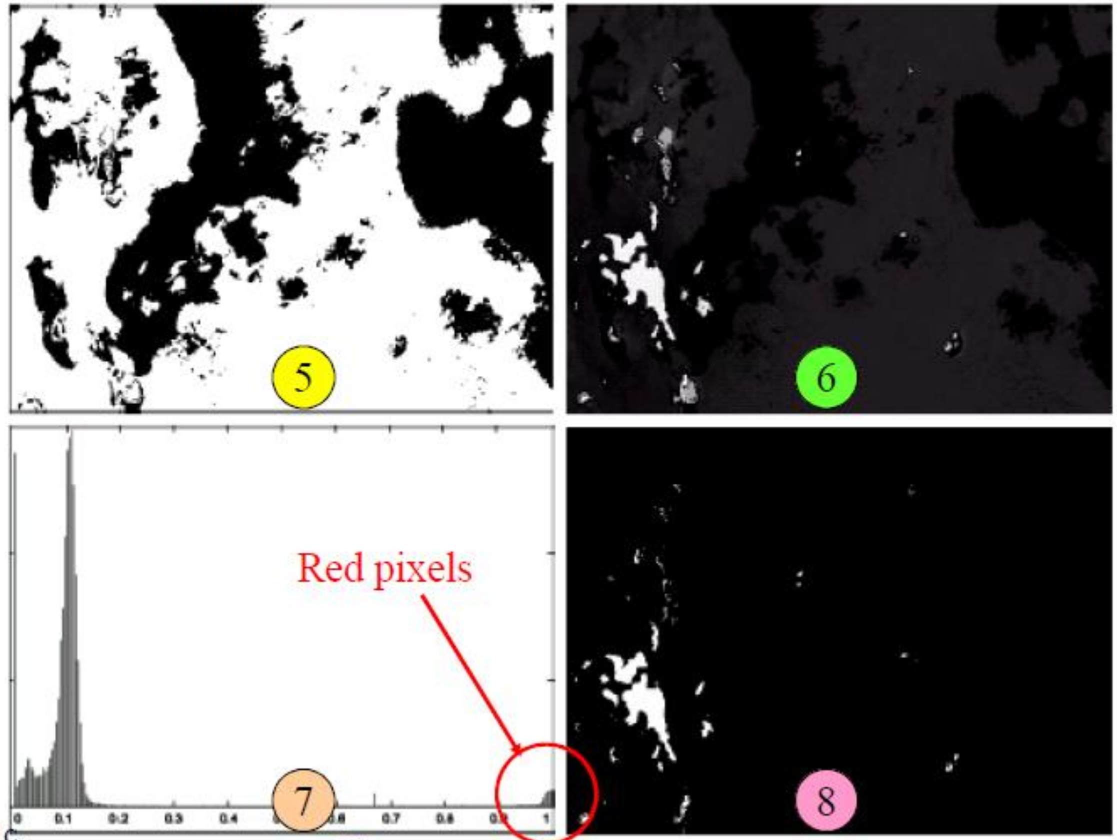
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2 Edition.

© 2017 NUS. All Rights Reserved.

Color Segmentation in HSI Color Space

Binary thresholding of S component with T = 10%

Product of 2 and 5



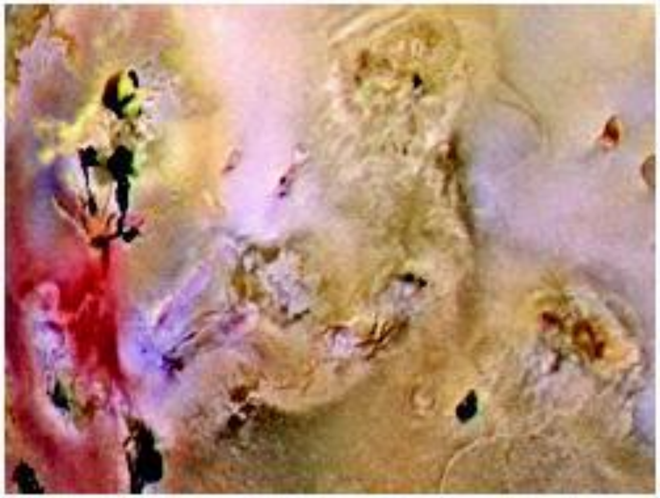
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.)

Histogram of 6

Segmentation of red color pixels

© 2017 NUS. All Rights Reserved.

Color Segmentation in HSI Color Space



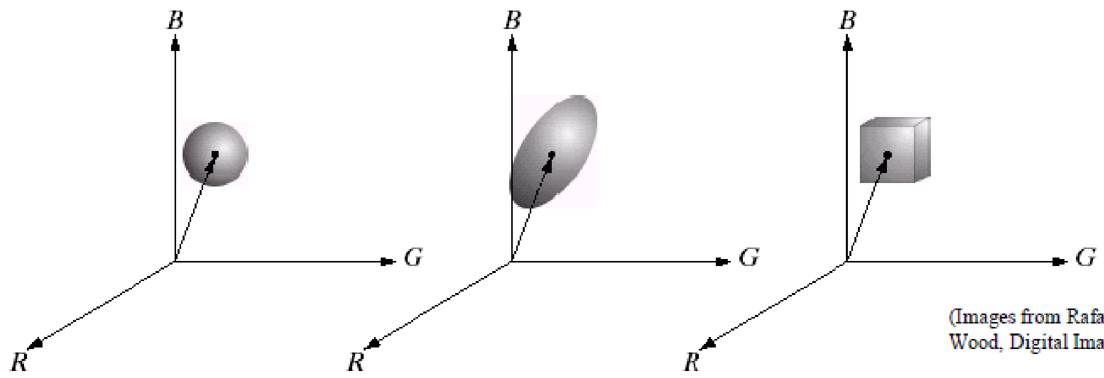
Color image

Segmented results of red pixels

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Color Segmentation in RGB Vector Space



a b c

FIGURE 6.43 Three approaches for enclosing data regions for RGB vector segmentation.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

1. Each point with (R,G,B) coordinate in the vector space represents one color.
2. Segmentation is based on distance thresholding in a vector space

$$g(x,y) = \begin{cases} 1 & \text{if } D(\mathbf{c}(x,y),\mathbf{c}_T) \le T \\ 0 & \text{if } D(\mathbf{c}(x,y),\mathbf{c}_T) > T \end{cases}$$

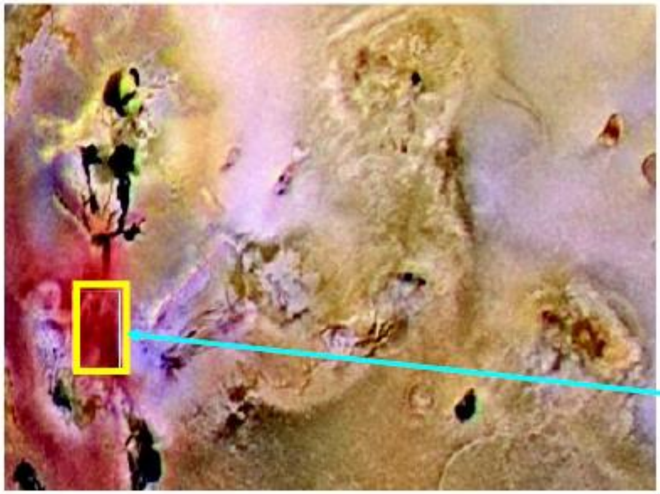
$D(\mathbf{u},\mathbf{v})$ = distance function

$\mathbf{c}_T$ = color to be segmented.

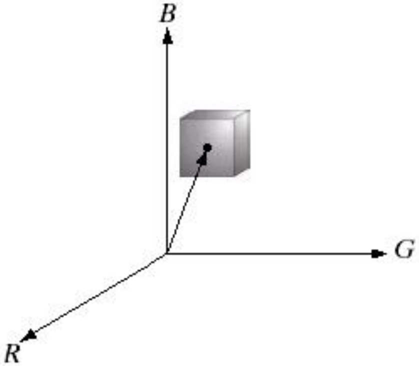
$\mathbf{c}(x,y)$ = RGB vector at pixel (x,y).

© 2017 NUS. All Rights Reserved.

Example: Segmentation in RGB Vector Space



Color image



Reference color $\mathbf{c}_T$ to be segmented

$\mathbf{c}_T$ = average color of pixel in the box

Results of segmentation in RGB vector space with Threshold value

T = 1.25 times the SD of R,G,B values In the box

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

© 2017 NUS. All Rights Reserved.

Robotic Vision & Navigation

© 2017 NUS. All Rights Reserved.

INTRODUCTION

- Navigation is plotting an efficient route from point A to point B.
- Robot navigation includes two things:
  - the ability to move
  - the means to determine whether or not the goal has been reached

© 2017 NUS. All Rights Reserved.

INTRODUCTION

- Navigation is the process of moving the robot to a goal location (or series of locations) in the environment, steering around obstacles along the way.
- This is accomplished via operations called Direct Motion commands, such as moving forward a given distance, turning to a given angle, or changing (accelerating or decelerating) to a given velocity.

© 2017 NUS. All Rights Reserved.

INTRODUCTION

- Robot navigation can be defined as the combination of three basic activities:

  - Map building

  - Localization

  - Path planning

© 2017 NUS. All Rights Reserved.

MAP BUILDING

- The process of constructing a map from sensor readings taken at different robot locations. The correct treatment of sensor data and the reliable localization of the robot are fundamental in the map-building process.

© 2017 NUS. All Rights Reserved.

LOCALIZATION

- The process of getting the actual robot's location from sensor readings and the most recent map. An accurate map and reliable sensors are crucial to achieving good localization.

© 2017 NUS. All Rights Reserved.

PATH PLANNING

- The process of generating a feasible and safe trajectory from the current robot location to a goal based on the current map. In this case, it is also very important to have an accurate map and a reliable localization.

© 2017 NUS. All Rights Reserved.

NAVIGATION

- Divided into 2 types:
- - Natural landmark navigation
- - Artificial landmark navigation
- Landmarks are distinct features that a robot can recognize from its sensory input. Landmarks can be geometric shapes such as rectangles, lines, circles, etc.

© 2017 NUS. All Rights Reserved.

NAVIGATION

- Landmarks are carefully chosen to be easily identifiable; for example, there must be sufficient contrast to the background.
- Before a robot can use landmarks for navigation, the characteristics of the landmark must be known and stored in the robot's memory.

© 2017 NUS. All Rights Reserved.

NATURAL LANDMARK NAVIGATION

- A natural landmark positioning system has the following basic components:
- - A sensor for detecting landmarks and contrasting them against their background .
- - A method for matching observed features with a map of known landmarks.
- - A method of computing location and localization errors from the matches.

© 2017 NUS. All Rights Reserved.

NATURAL LANDMARKS NAVIGATION

- Natural landmarks work best in highly structured environments such as corridors, manufacturing floors, or hospitals.
- *natural landmarks define* as those objects or features that are already in the environment and have a function other than robot navigation.
- Natural landmarks require no modifications to the environment.

© 2017 NUS. All Rights Reserved.

ARTIFICIAL LANDMARK NAVIGATION

- artificial landmarks define as specially designed objects or markers that need to be placed in the environment with the sole purpose of enabling robot navigation.
- Detection is much easier with artificial landmarks, which are designed for optimal contrast. In addition, the exact size and shape of artificial landmarks are known in advance.
- Artificial landmarks are inexpensive and can have additional information encoded on them.

© 2017 NUS. All Rights Reserved.

MAP BASED NAVIGATION

- Map-based positioning (also known as "map matching"), is a technique in which the robot uses its sensors to create a map of its local environment.
- This local map is then compared to the global map previously stored in memory. If a match is found then the robot can compute its actual position and orientation in the environment.
- The pre stored map can be a CAD model of the environment, or it can be constructed from prior sensor data.

© 2017 NUS. All Rights Reserved.

MAP BASED NAVIGATION

- The main advantages of map-based positioning are given below:
- It uses naturally the naturally occurring structure of typical indoor environments to derive position information without modifying the environment.
- It can be used to generate an updated map of the environment. Environment maps are important for other mobile robot tasks, such as global path planning.
- It allows a robot to learn about a new environment and to improve positioning accuracy through exploration.

© 2017 NUS. All Rights Reserved.

MAP BASED NAVIGATION

- Disadvantages of map-based positioning arise because it requires that:
- There be enough stationary, easily distinguishable features that can be used for matching.
- The sensor map be accurate enough (depending on the task at hand) to be useful.
- A significant amount of sensing and processing power is available.

© 2017 NUS. All Rights Reserved.

VISION BASED NAVIGATION

- Vision based positioning or localization uses the same basic principles of landmark-based and map-based positioning but relies on optical sensors rather than ultrasound, dead-reckoning and inertial sensors.

© 2017 NUS. All Rights Reserved.

VISION BASED NAVIGATION

- The advantage of these type of sensors lies in their ability to directly provide distance information needed for collision avoidance. They have an important drawback in that only vertical structures (ie. mainly the shape of the free space surrounding the robot) can be recognised.

© 2017 NUS. All Rights Reserved.

VISION BASED NAVIGATION

- Visual sensing can provide the robot with an incredible amount of information about its environment.
- The most common optical sensors include laser-based range finders and photometric cameras using CCD arrays.

© 2017 NUS. All Rights Reserved.

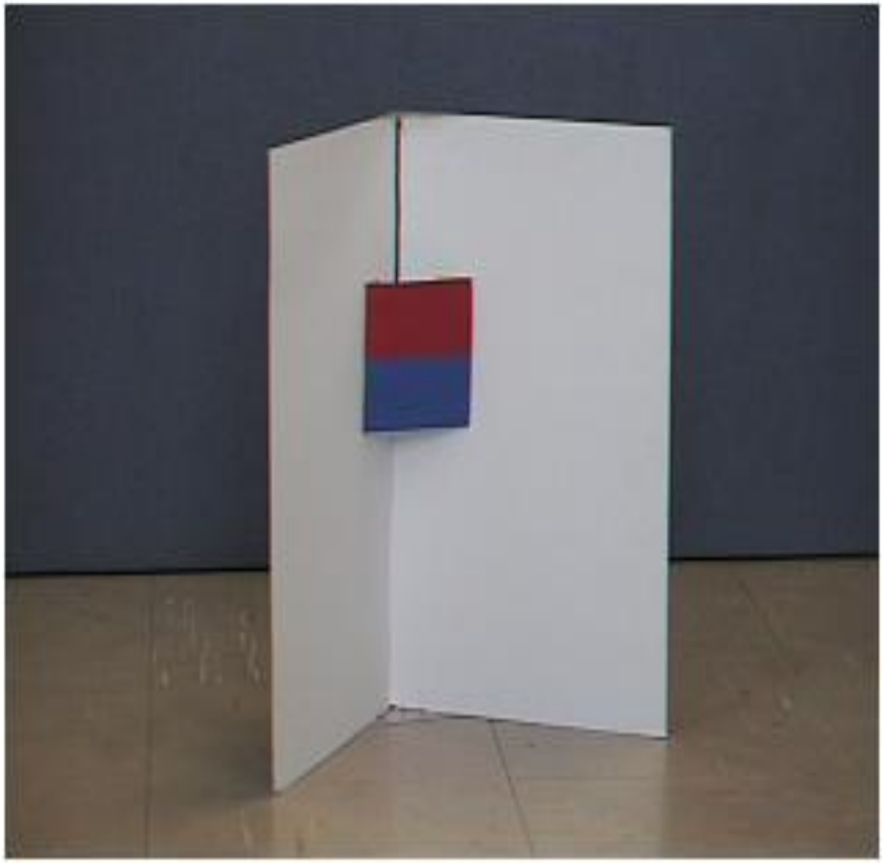
EXAMPLE: Navigation of indoor mobile robots

- The landmark model is designed to have a three-dimensional structure consisting of a multi-colored planar pattern.
- Experimental results show that the proposed landmark model is effective.

© 2017 NUS. All Rights Reserved.

EXAMPLE: Navigation of indoor mobile robots

- Figure shows the appearance of the proposed landmark. The color pattern is composed of two vertically neighboring color patches. The color pattern makes the angle of 45o with respect to two supporting planes at the right angle.



© 2017 NUS. All Rights Reserved.

EXAMPLE: Navigation of indoor mobile robots

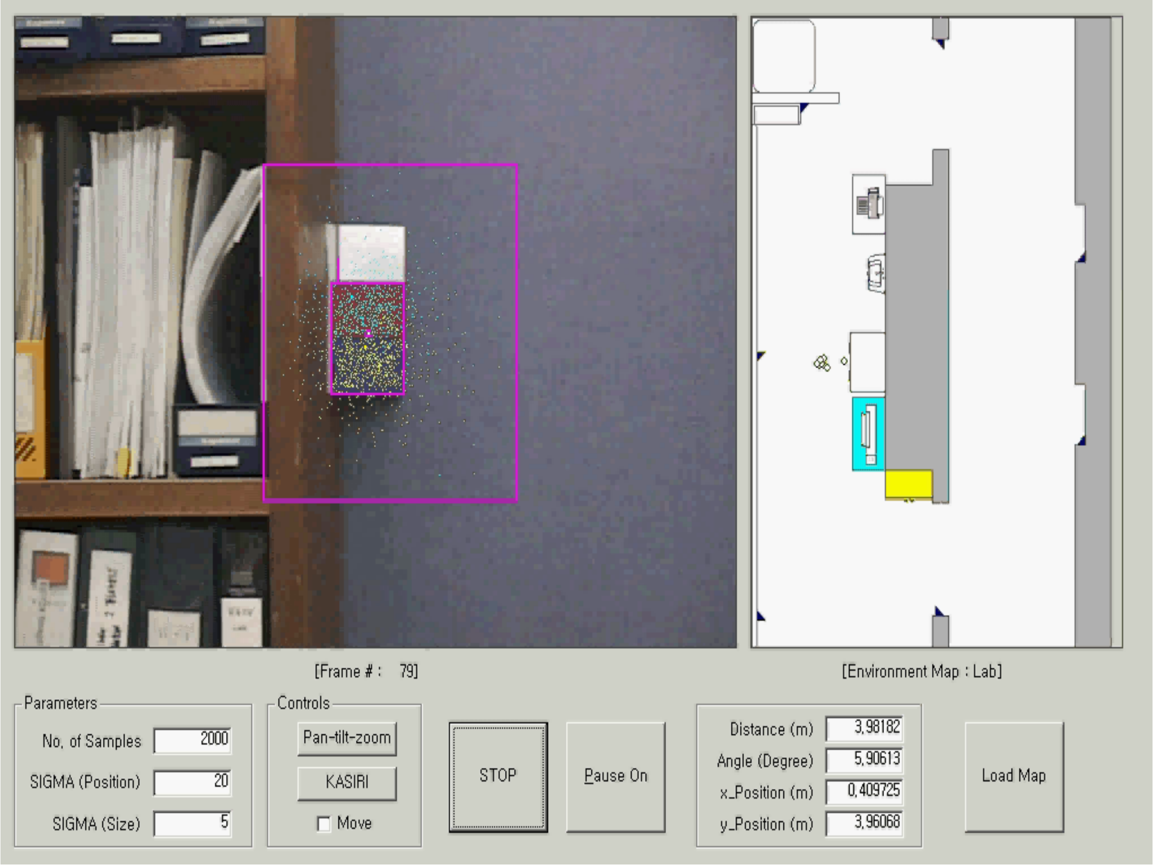
- Seven landmarks is set at the predetermined positions in laboratory and corridor then the robot estimated its position and corrected the direction of movement along the round path.

© 2017 NUS. All Rights Reserved.

EXAMPLE: Navigation of indoor mobile robots

© 2017 NUS. All Rights Reserved.

EXAMPLE: Navigation of indoor mobile robots



© 2017 NUS. All Rights Reserved.

End of Lecture 3

© 2017 NUS. All Rights Reserved.