UNIVERSIDAD PANAMERICANA

Mecatrónica y Control de Sistemas

Color Image Processing

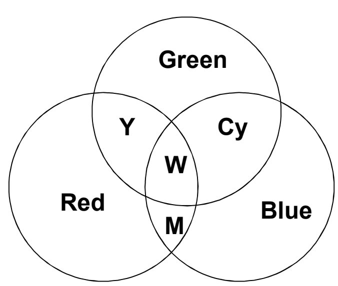
Introduction

Types of Color Processing:

Full-color: Images acquired with a full-color sensor such as a camera or color scanner.

Pseudo color (false color): Images acquired with a monochromatic sensor such as a thermal camera, ultrasound sensor, etc. The problem is to assign a color to a gray level.

Primary & Secondary colors



Y: yellow
Cy: Cyan
M: Magenta
W: White

Color Models

Color models are specifications of colors in some standard generally in hardware (monitors, printers etc.) and applications (graphics and animations).

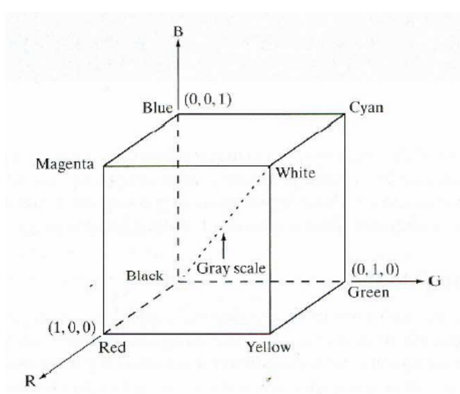
3 main models:

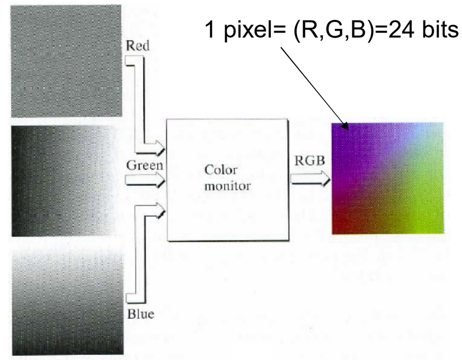
- **RGB** (Red-Green-Blue) for monitors and video cameras
- **CMY** (Cyan-Magenta-Yellow) and **CMYK** (Cyan-Magenta-Yellow-Black) for printers
- **HSI** (Hue-Saturation-Intensity) it is the model of human vision

CMY model

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

RGB model





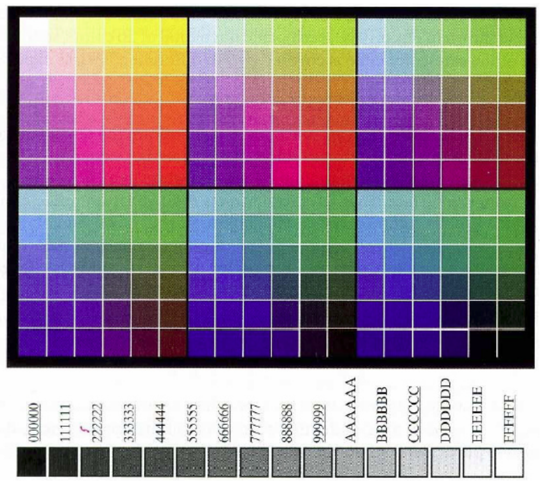
The term *full-color* is used to describe a 24-bit RGB image.

Problem: Even though the total number of colors is $(2^8)^3$, most devices are limited to 256 colors: there are applications that use just a few colors.

RGB model

There is a subgroup of 216 colors that represent the whole $(2^8)^3$ colors and are known as *safe colors.*

Each *safe color* takes values in multiples of 51:
0, 51, 102, 153, 204, 255 and they are usually expressed in hexadecimal:



Example: Pure red
R=255 (FF) G=0 (00) B=0 (00)

$\longrightarrow$ Pure red: FF0000

Black: 000000

White: FFFFFF

HSI Model

RGB and **CMY** models suit hardware perfectly but not so human vision
(You never say: “my car is 45% red, 25% green and 30% blue”).

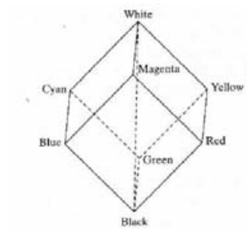
Human vision is based on 3 parameters:

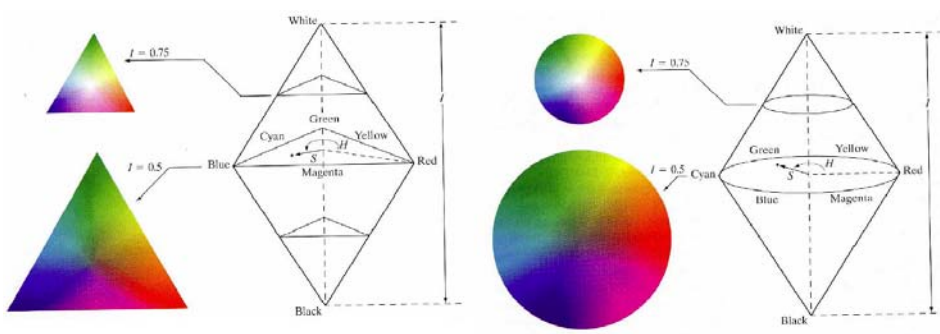
Hue (Matiz): It represents the dominant color perceived by an observer. When we say that an object is red, orange, yellow we are specifying its hue.

Saturation: It refers to the way we perceive an object’s hue through white light.
Example: pink=red+white

Intensity: A subjective notion of brightness (can’t be measured)

HSI Model





A little history: old models…

NTSC (National Television System Committee) model

TV in America and Japan. Three components (YIQ):

luma (Y), in-phase (I), quadrature (Q)

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

PAL (Phase Alternating Line) model

TV in Europe, Africa, Australia, and Asia. Three components (YUV):

luma (Y), blue tones (U), red tones (V)

Y = 0.299 R + 0.587 G + 0.114 B

U= 0.493 (B − Y)

V = 0.887 (R − Y)

Fundamentals

CMY <--> RGB

cmy_image=imcomplement(rgb_image)

RGB --> HSI

hsi_image=rgb2hsi(rgb_image)

HSI --> RGB

rgb_image=hsi2rgb(hsi_image)

Color reduction

[f1 map1]=rgb2ind(f, n, option)

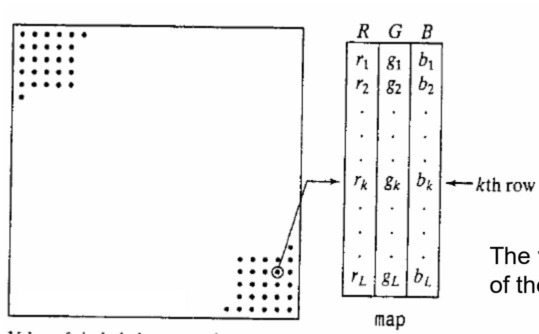
'dither' - Achieves better color resolution at the expense of spatial resolution (default)

'nodither' - Maps each color in the original image to the closest color in the new map. No dithering is performed.

Pseudo color

Objective: To assign a color to a gray-level (gray2rgb).

Matlab works with colormaps. Example::



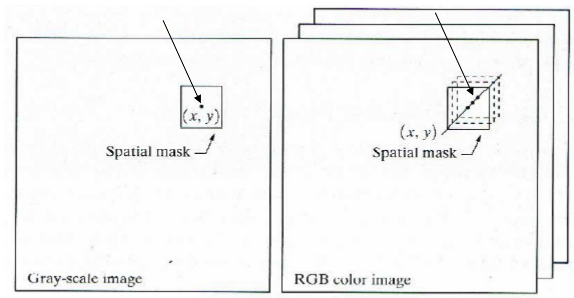
Value of circled element = k

The value in the matrix is the number of the row in the colormap

See pre-established colormaps in Matlab

Full color

$k \rightarrow 1$ *value* $\qquad k_{(R,G,B)} \rightarrow 3$ *values*



The simplest way is the individual processing of each layer in gray-level

Color Filters

Averaging filter:

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

Laplacian filter:

$$\nabla^2[\mathbf{c}(x, y)] = \begin{bmatrix} \nabla^2 R(x, y) \\ \nabla^2 G(x, y) \\ \nabla^2 B(x, y) \end{bmatrix}$$

etc…..