

Color Image Processing

Color Fundamentals

- *Visible light can be represented by the combination of different colors with changing electromagnetic wavelengths from violet to red in the visible spectrum.*

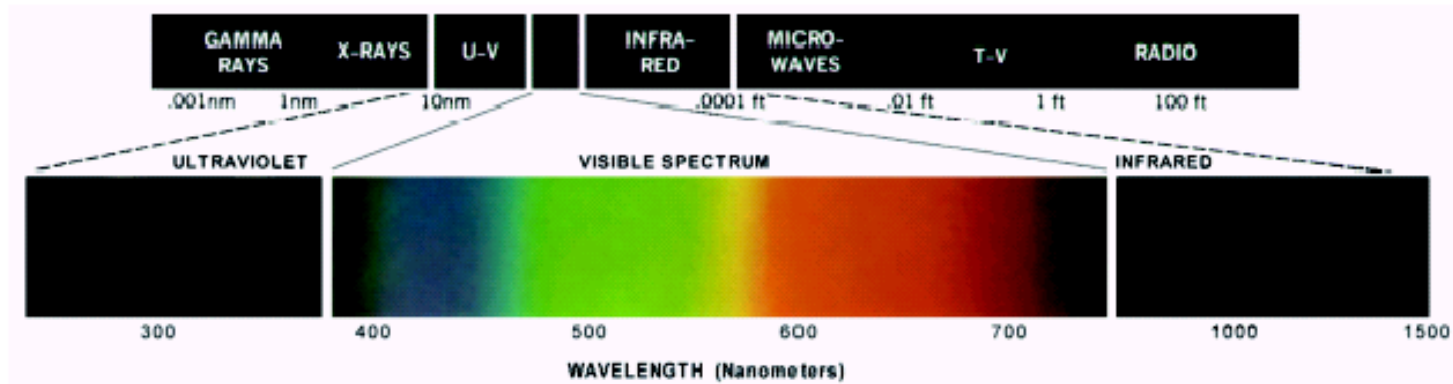


FIGURE 6.2 Wavelengths comprising the visible range of the electromagnetic spectrum. (Courtesy of the General Electric Co., Lamp Business Division.)

Color Image Processing

Color Fundamentals

- Chromatic (colored) light spans to electromagnetic waves between approximately 400 nm and 700 nm which corresponds to colors from violet to red. These colors are perceived by the human eye in the following way.*

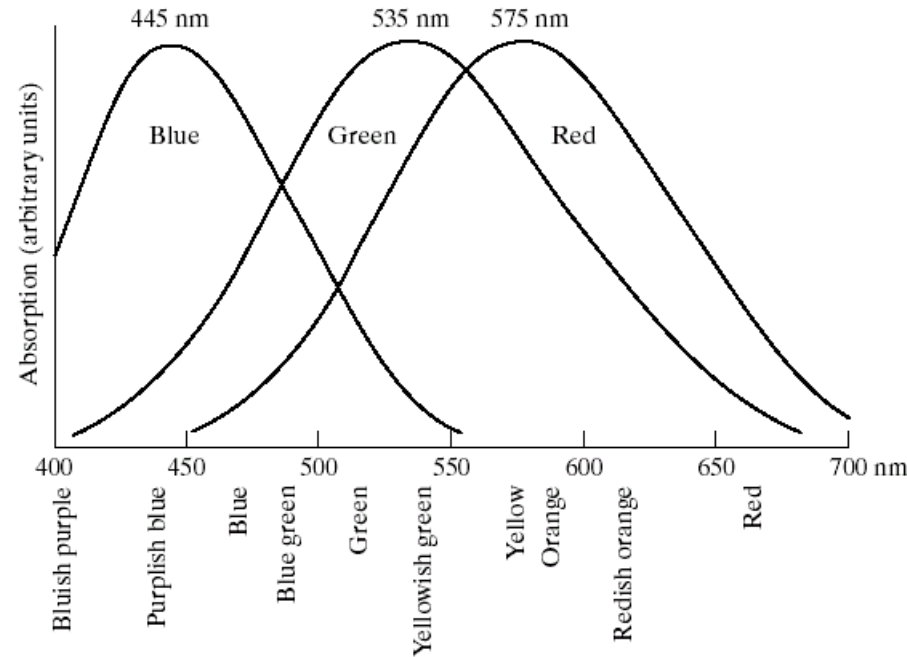
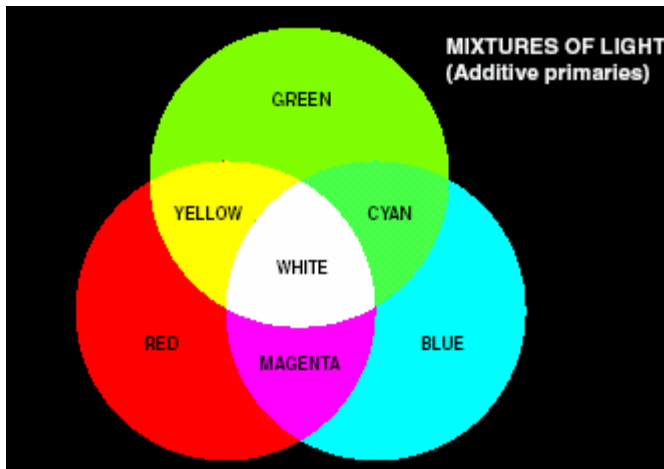


FIGURE 6.3 Absorption of light by the red, green, and blue cones in the human eye as a function of wavelength.

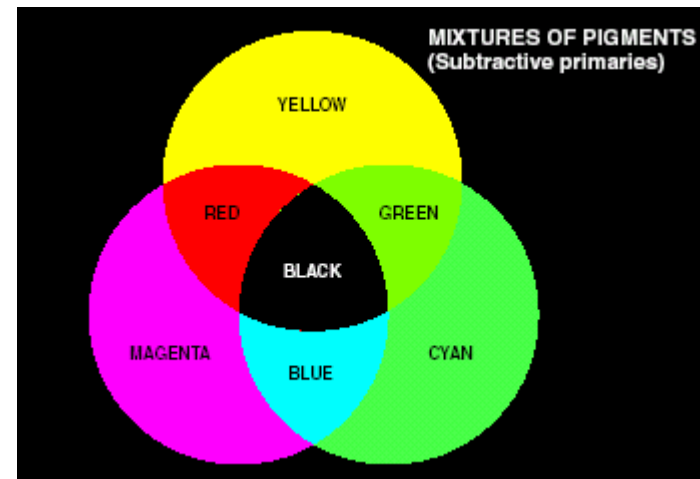
Color Image Processing

Color Fundamentals

- **Primary Colors:** *Red(R), Green(G) and Blue(B)* are referred as the primary colors and when mixed with various intensity proportions, can produce all visible colors.
- The primary colors can be mixed to generate **secondary colors** such as magenta (red+blue), cyan (green + blue) and yellow (red+green).



Primary Colors -
RGB

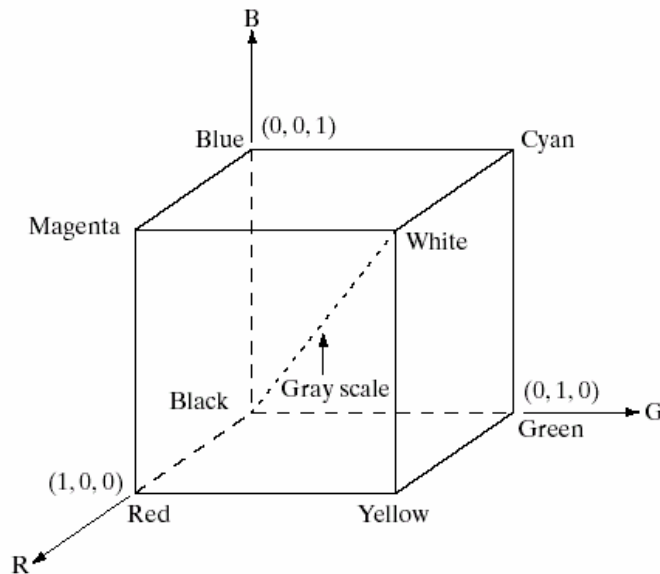


Secondary Colors -
CMYK

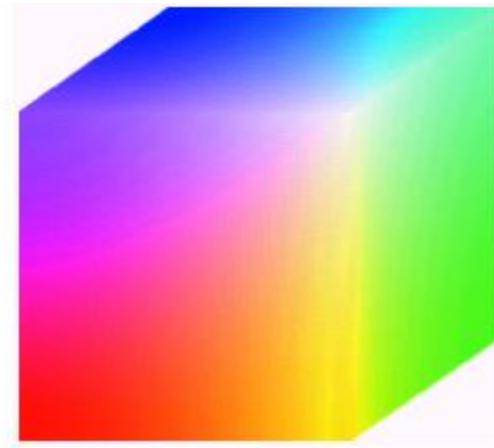
Color Image Processing

Color Models

- **RGB Color Model:** *In this color model each color appears in its primary spectral components of Red, Green and Blue.*
- *The model is based on the 3D Cartesian coordinate system, where the color subspace of interest is the color cube shown below.*



Color Cube
0,0,0=Black
1,1,1=White

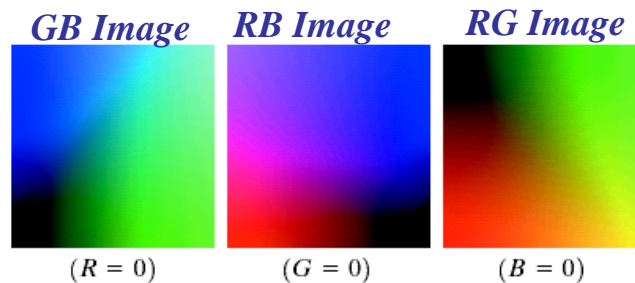
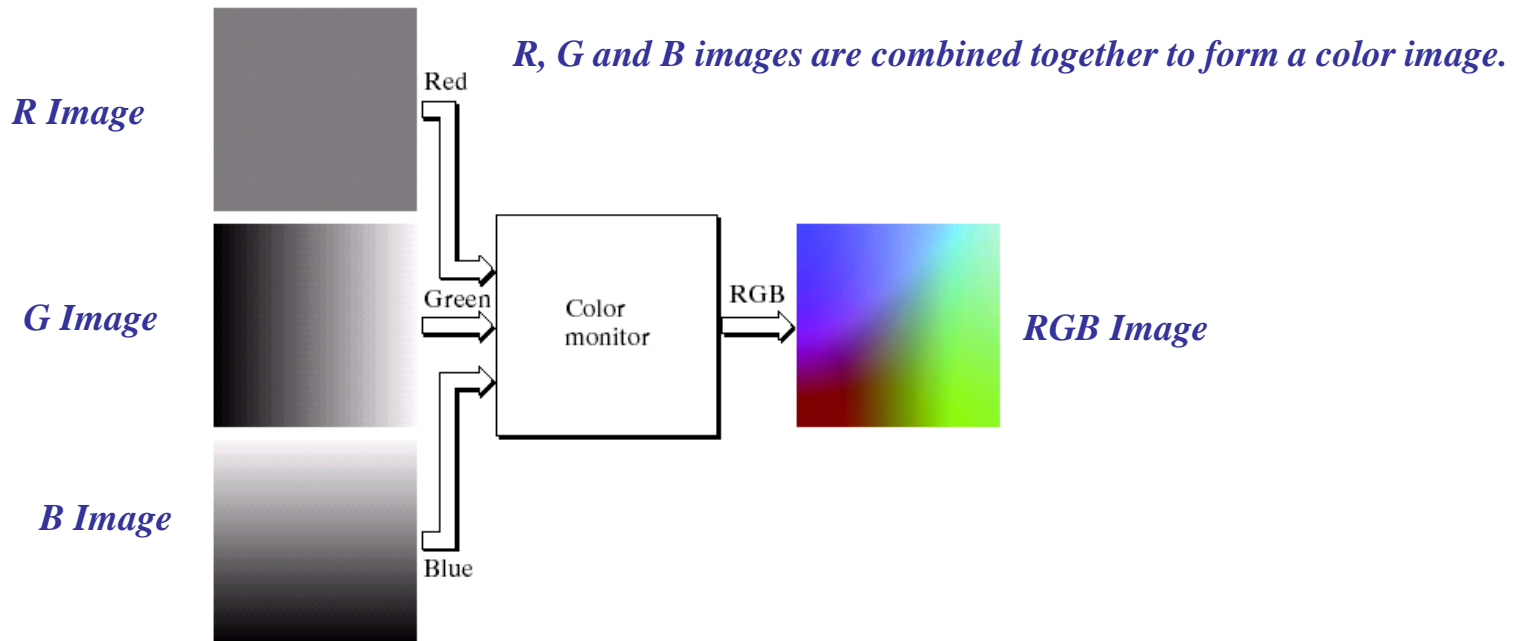


24-bit Color Cube

Color Image Processing

Color Models

- **RGB Color Model:**



Color Image Processing

Color Models

- CMY Color Model: Cyan, Magenta and Yellow color model is made of secondary colors.
- Some printers and devices use secondary colors instead of the primary colors.
- The conversion from RGB to CMY can be performed as follows:

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

- HSI Color Model: Hue, Saturation and Intensity are three important descriptors used by human being in describing colors.
- Hue represents the purity of the color. (i.e. pure red, yellow, green).
- Saturation represents the measure of the degree to which a pure color is diluted by white light.
- Intensity is the gray level value of the color.
- Hue and Saturation represents the color carrying Chrominance (Chromatic) information.
- Intensity represents the gray-level Luminance (achromatic) information.

Color Image Processing

Color Models

- **HSI Color Model:** *Hue, Saturation and Intensity can be represented by:*

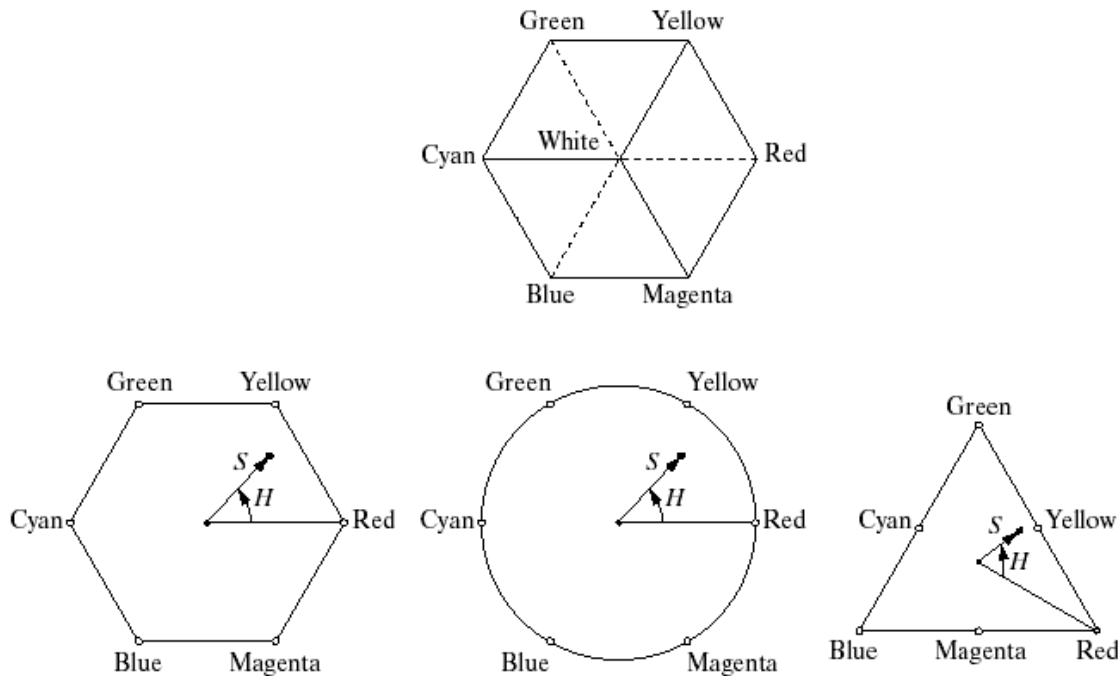
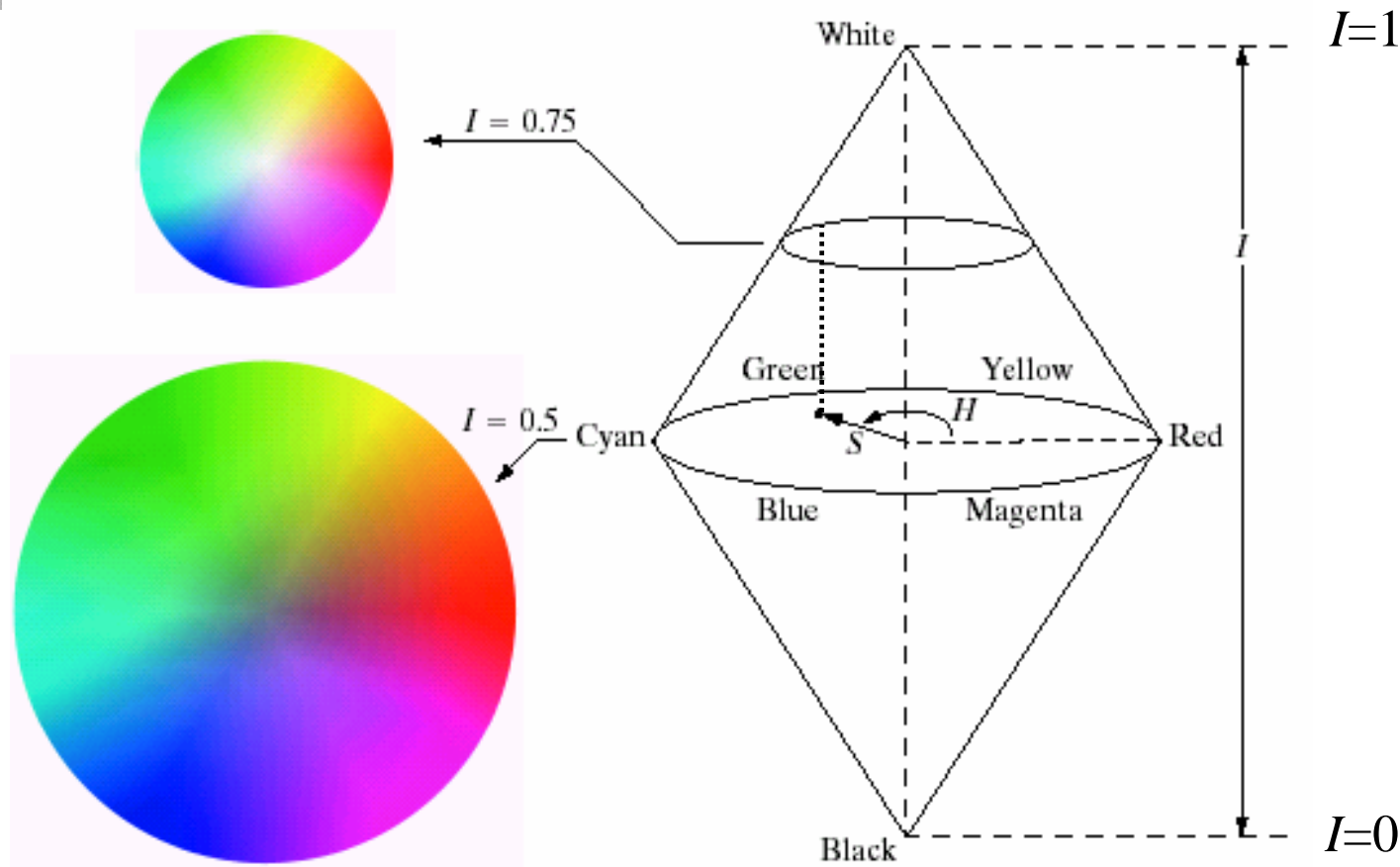


FIGURE 6.13 Hue and saturation in the HSI color model. The dot is an arbitrary color point. The angle from the red axis gives the hue, and the length of the vector is the saturation. The intensity of all colors in any of these planes is given by the position of the plane on the vertical intensity axis.

Color Image Processing

Color Models

- *HSI Color Model: Hue, Saturation and Intensity*



3D Circular HSI color plane

Color Image Processing

Color Models

- **Converting Color from RGB to HSI:** Given an image in RGB color format, the *H* component of each RGB pixel is obtained using:

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

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

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

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

• The RGB values are normalized to the range [0,1].

• The SI values are in [0,1] and the H value can be divided by 360° to be in the same range.

Color Image Processing

Color Models

- *Converting from HSI to RGB : Given the HSI values in the interval of [0,1].*
- *Multiply the H by 360° so that it is in the range of [0, 360°].*
- *If $(0 \leq H < 120^\circ)$, then color is in RG Sector and then:*

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

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

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

Color Image Processing

Color Models

- *Converting from HSI to RGB : Given the HSI values in the interval of [0,1].*
- *Multiply the H by 360° so that it is in the range of [0, 360°].*
- *If ($120^\circ \leq H < 240^\circ$), then color is in GB Sector and then. Firstly H in this sector is:*

$$H = H - 120^\circ$$

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

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

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

Color Image Processing

Color Models

- *Converting from HSI to RGB : Given the HSI values in the interval of [0,1].*
- *Multiply the H by 360° so that it is in the range of [0, 360°].*
- *If ($240^\circ \leq H < 360^\circ$), then color is in BR Sector and then. Firstly H in this sector is:*

$$H = H - 240^\circ$$

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

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

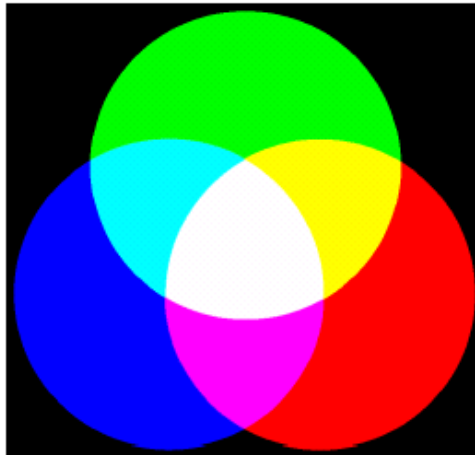
$$R = 1 - (G + B)$$

Color Image Processing

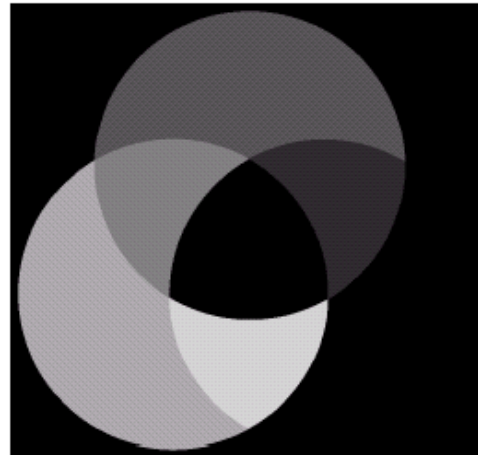
Color Models

- *Converting from HSI to RGB :*

RGB Image



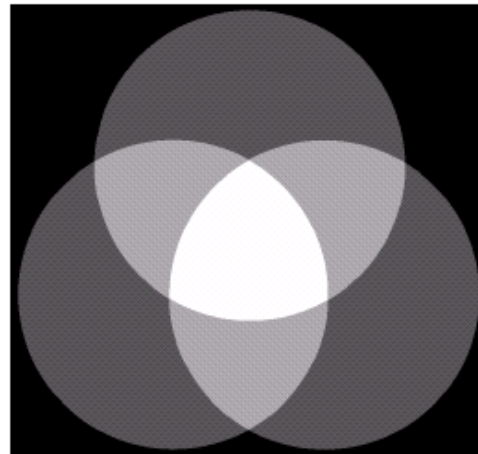
H Image



S Image



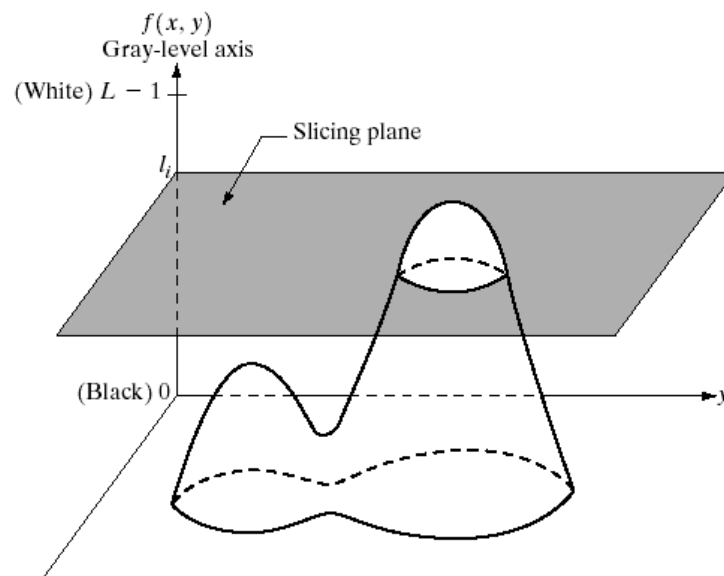
I Image



Color Image Processing

Pseudocolor Image Processing

- *This technique is based on assigning color (false/pseudo) values to different gray levels. By converting monochrome images to color images human visualization and interpretation of the gray level images can be improved.*
- **Intensity/density Slicing:** *The image is interpreted as 3D function (intensity versus spatial coordinates), where, planes which are parallel to the coordinate planes called “slices” are considered to slice the image function into two color levels.*



Let $[0, L-1]$ represent the gray level image, and l_0 represents black $[f(x,y)=0]$, l_{L-1} represents white $[f(x,y)=L-1]$,

Consider P planes defined at l_1, l_2, \dots, l_P intensity levels. Then P planes partition the gray scale into $P+1$ intervals, V_1, V_2, \dots, V_{P+1} . Each interval is assigned to a different color and hence:

$$f(x, y) = c_k \quad , \quad \text{if } f(x, y) \in V_k$$

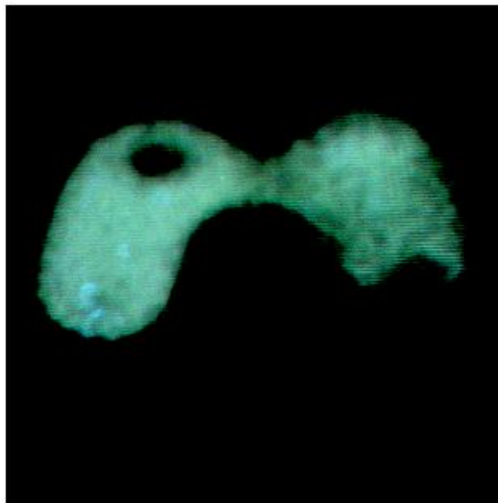
c_k is the color with k^{th} intensity interval V_k .

Intensity slicing

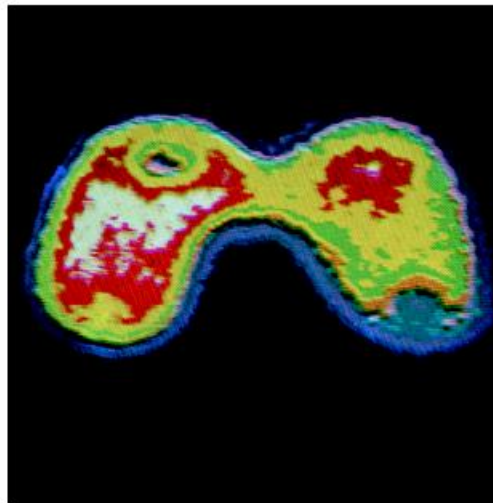
Color Image Processing

Pseudocolor Image Processing

- Intensity/density Slicing:



Monochrome Image



Intensity slicing into 8 colors

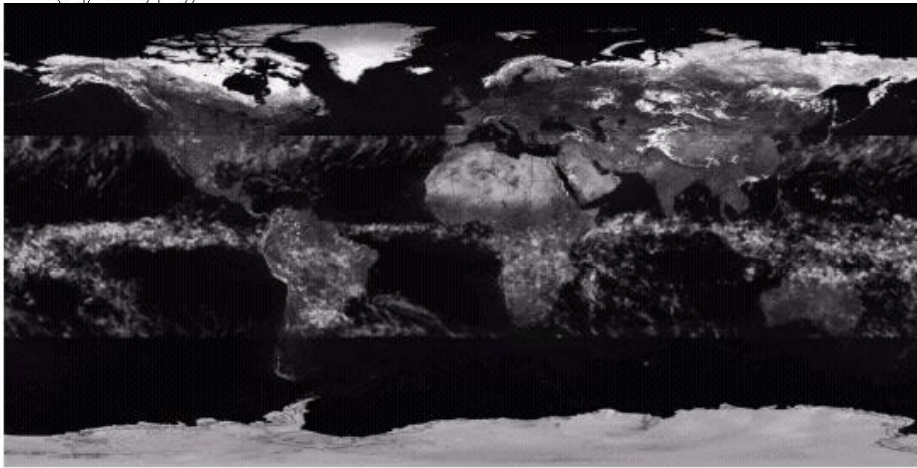
Color Image Processing

Pseudocolor Image Processing

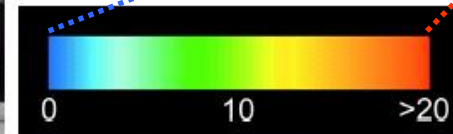
- Intensity/density Slicing:



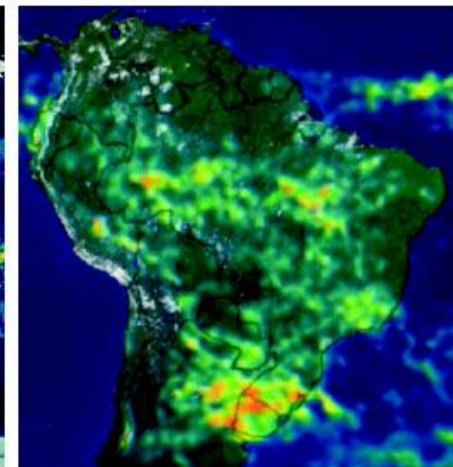
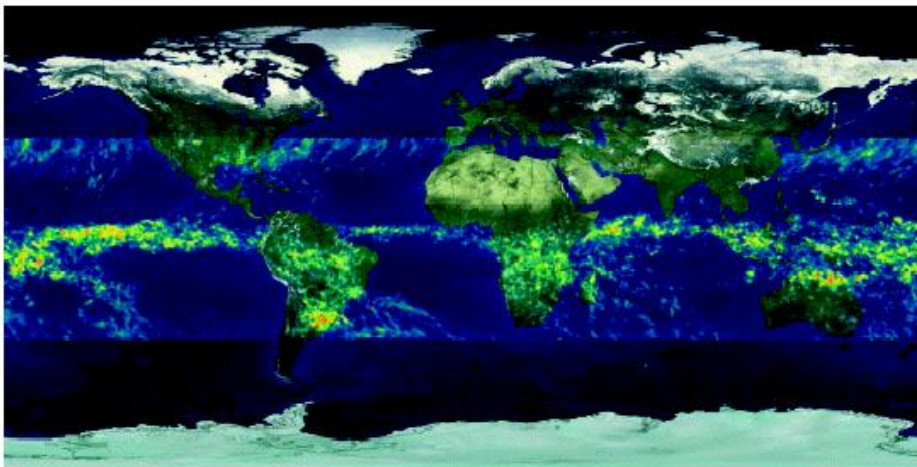
Monochrome
Image



intensity slicing into 256 colors
Intensity values from 0 to 255



After intensity
slicing applied
Tropical regions

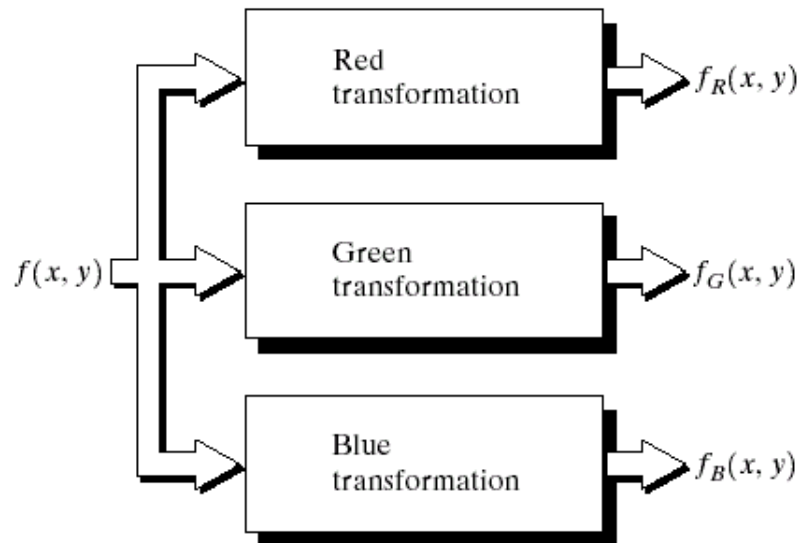


Zoom of south
America region

Color Image Processing

Pseudocolor Image Processing

- Gray Level to Color Transformations : In this method, a given gray level image is processed by 3 different transformation functions producing 3 enhanced images in Red, Green and Blue channels respectively.
- By combining the 3 channel images we get a colored image.

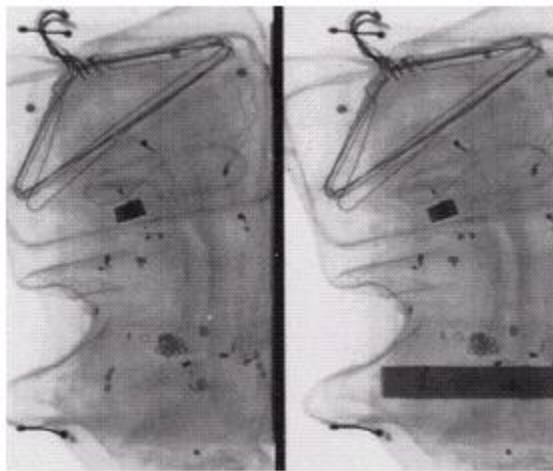
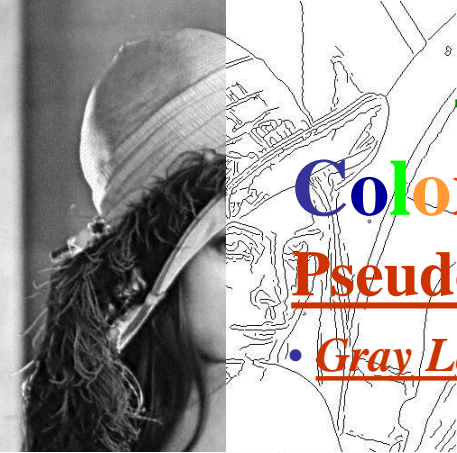


f_R , f_G and f_B are used to be inputs to an RGB monitor, producing a colored image.

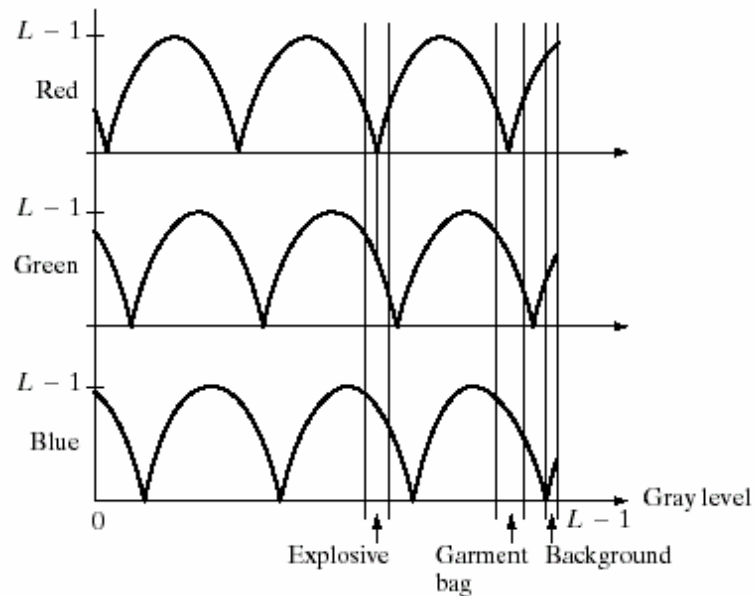
Color Image Processing

Pseudocolor Image Processing

- Gray Level to Color Transformations :

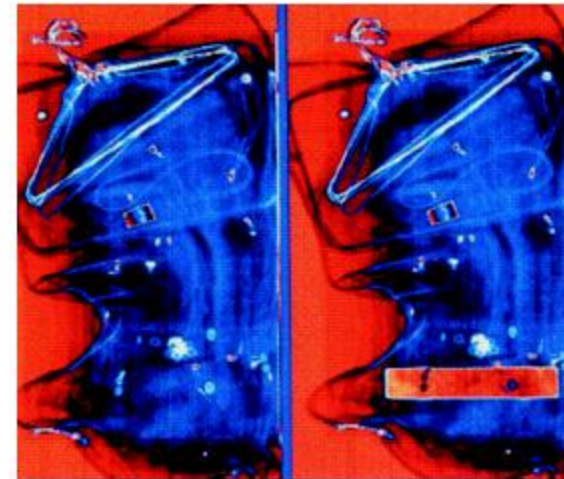


input image.



Transformation functions for R, G and B channels.

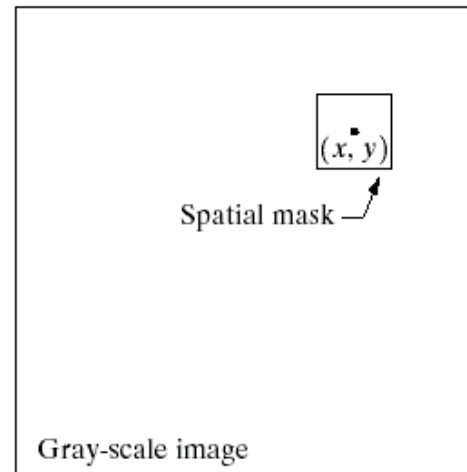
Output Pseudocolor image.



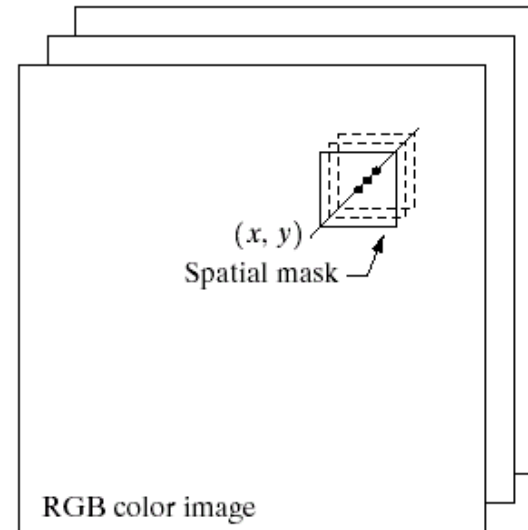
Color Image Processing

Full-Color Image Processing

- *There are two main methods in using full color images.*
 - *In the first method each color component is processed separately to form a composite color image.*
 - *In the second approach we consider each pixel as a vector of 3 values and process each pixel.*



Each component is processed like a gray-level image



Each pixel is considered as a vector of RGB components.

Color Image Processing

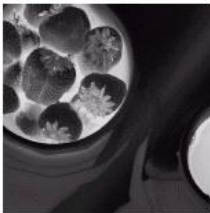
Full-Color Image Processing

- Various Color space components: Consider each color component as a gray level image.



Full color

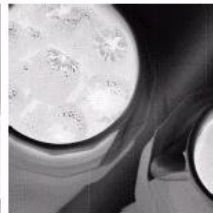
Full color image



Cyan



Magenta



Yellow

C,M,Y components.



Red

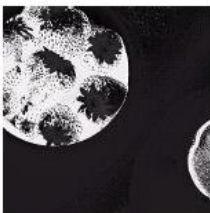


Green



Blue

R,G,B components.



Hue



Saturation



Intensity

H,S,I components.

Color Image Processing

Full-Color Image Processing

- Color Complements (inverse colors). Color complementing a color image is identical to gray scale negatives in monochrome images.
- Color complement transformations are performed according to the following color circle.

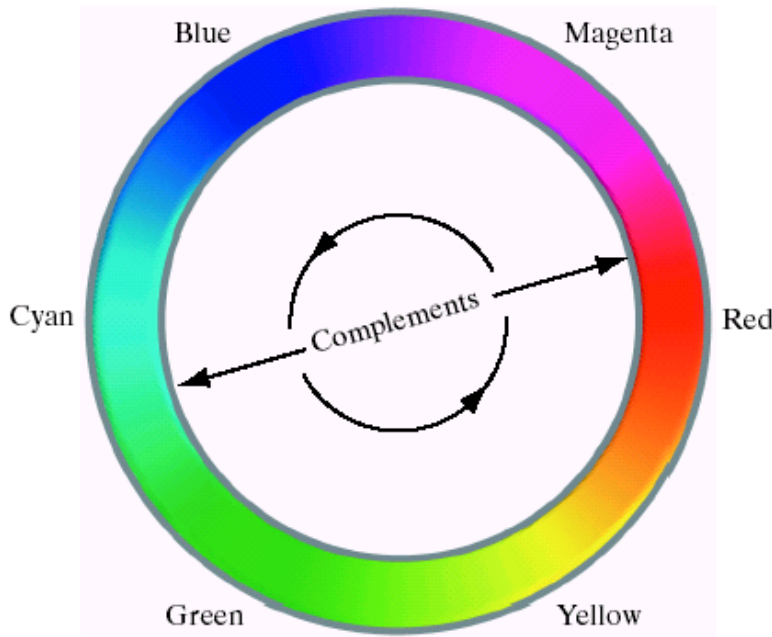
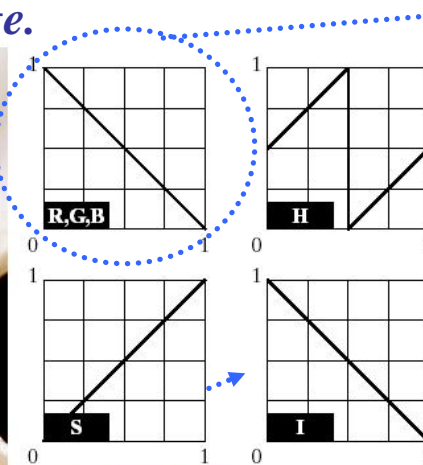


FIGURE 6.32
Complements on
the color circle.

Color Image Processing

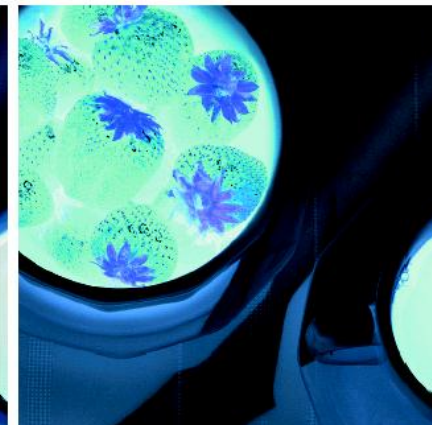
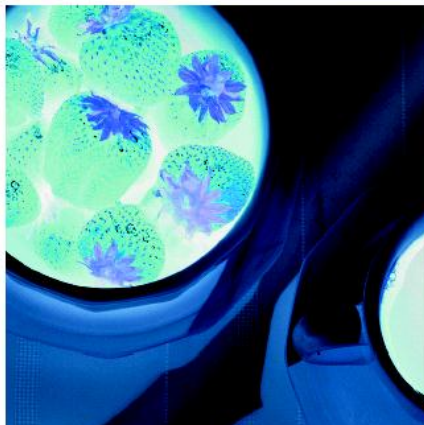
Full-Color Image Processing

- Color Complements (inverse colors). If you apply respective color complement transformation to each color component, you can obtain the complement of a given color image.



Transformation functions for RGB

Transformation functions for HSI



Complementing RGB components

Complementing HSI components



Color Image Processing

Full-Color Image Processing

- Color Image Smoothing and Sharpening : The idea of gray-scale image smoothing can be extended into processing of full color images.
- Let S_{xy} denote the set of coordinates defining a neighborhood centered at (x,y) in RGB color image. Averaging (smoothing) and Sharpening using Laplacian operator of the RGB component vectors in this neighborhood is:

Smoothing using Averaging

$$\bar{\mathbf{c}}(x, y) = \frac{1}{K} \sum_{(x,y) \in S_{xy}} \mathbf{c}(x, y)$$

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

K is the number of pixels within the neighborhood of the averaging mask.

Sharpening using Laplacian

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

Color Image Processing

Full-Color Image Processing

- Color Image Smoothing: Given a full color image with the following color components,



Full color RGB image



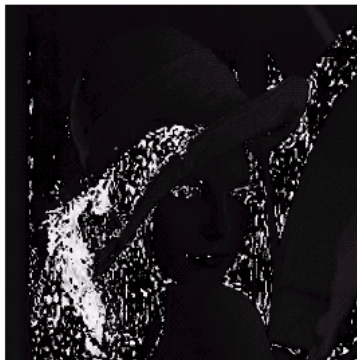
R Image



G Image



B Image



Hue image



Saturation Image



Intensity Image

Color Image Processing

Full-Color Image Processing

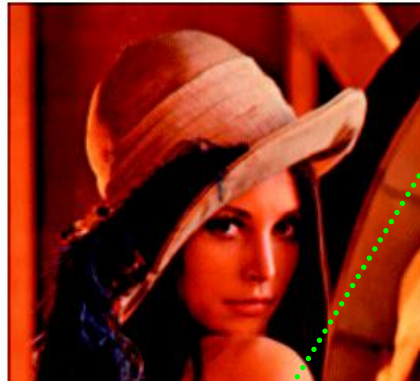
- Color Image Smoothing & Sharpening:

* Original Hue and Saturation are maintained. Working with HSI image is a better idea.

Smoothing by
5x5 averaging mask



Each RGB component filtered



Only I of the HSI filtered



Difference of the 2 images

Sharpened by
3x3 Laplacian mask



Each RGB component filtered



Only I of the HSI filtered



Difference of the 2 images

Color Image Processing

Full-Color Image Processing

- Segmentation in RGB vector space : Although working with HSI space is more intuitive in most applications, in segmentation working with RGB color vectors is generally more advantages.
- Suppose that an object within a specified RGB color range is to be segmented. Assume that \mathbf{a} is the average RGB vector. Each RGB pixel is classified to have color in the specified range/distance from the average color vector.
- Let \mathbf{z} denote an arbitrary point in RGB space. Then the Euclidean Distance between \mathbf{z} and \mathbf{a} is given by:

$$\begin{aligned} D(\mathbf{z}, \mathbf{a}) &= \|\mathbf{z} - \mathbf{a}\| = \left[(\mathbf{z} - \mathbf{a})^T (\mathbf{z} - \mathbf{a}) \right]^{1/2} \\ &= \left[(z_R - a_R)^2 + (z_G - a_G)^2 + (z_B - a_B)^2 \right]^{1/2} \end{aligned}$$

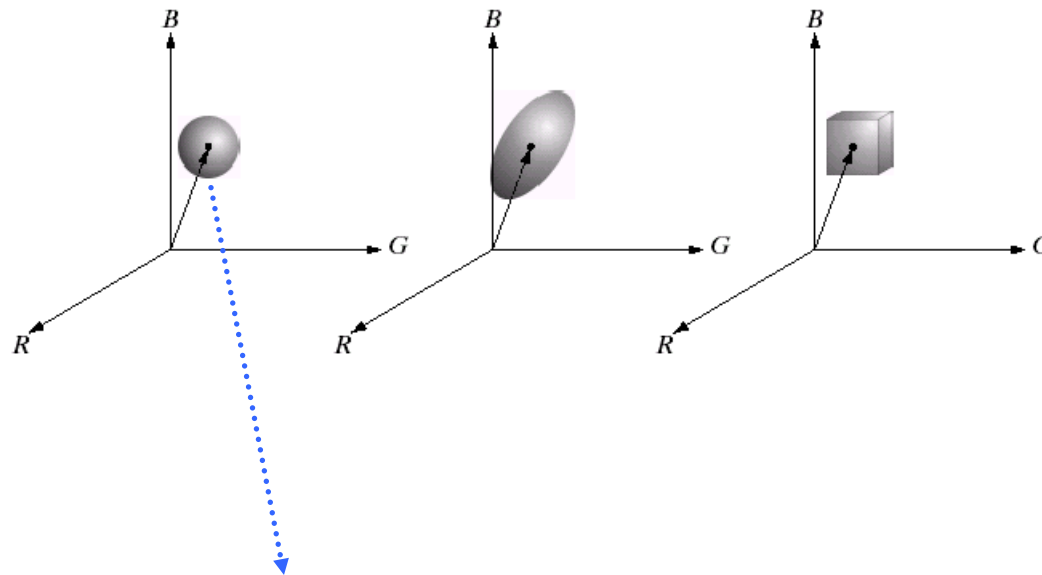
- The segmented pixels fall into the solid sphere of radius D_o where,

$$D(\mathbf{z}, \mathbf{a}) \leq D_o$$

Color Image Processing

Full-Color Image Processing

- Segmentation in RGB vector space :



$$D(\mathbf{z}, \mathbf{a}) \leq D_o, \text{ radius} = D_o$$

a b c

FIGURE 6.43

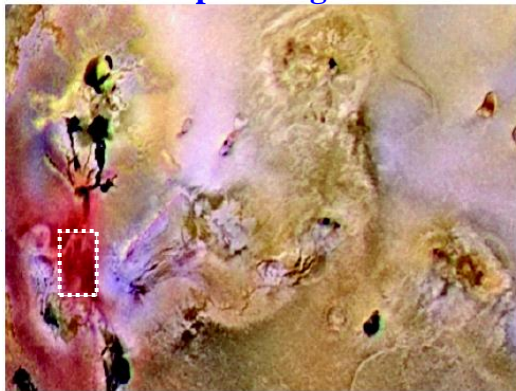
Three approaches for enclosing data regions for RGB vector segmentation.

Color Image Processing

Full-Color Image Processing

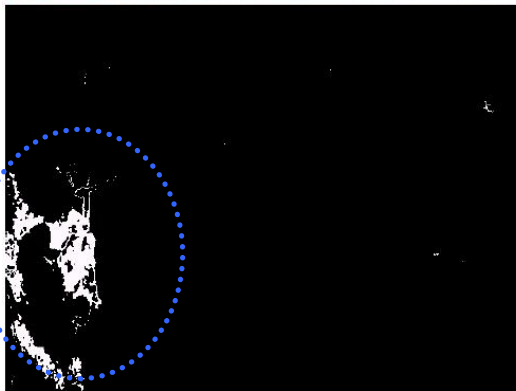
Segmentation in RGB vector space :

Input image



Area of interest

Segmented region



Segmented Output image

Segmentation Procedure:

1. Sample data is taken from the area of interest.

2. Mean RGB components and their standard deviation values are calculated.

3. D_o is determined by using the standard deviation σ .

For example in this example:

$$D_{oR} = 1.25 \sigma_R$$

$$D_{oG} = 1.25 \sigma_G$$

$$D_{oB} = 1.25 \sigma_B$$

4. Any pixel within D_o distance from the mean is set to white color and all the other pixels are set to black color.