UNIT-4

Color Image Processing

Introduction

The use of color in image processing is motivated by two principal factors. They are Color is a powerful descriptor that often simplifies object identification and extraction from a scene. Humans can discern thousands of color shades and intensities, compared to about only two dozen shades of gray. Color in image processing is divided into two major areas,

***Full-color processing*:** Images acquired with a full-color sensor, such as color TV camera or Color scanner.

Pseudo-color processing: Assigning a color to a particular monochrome intensity or range of Intensities.

4.1. Color Fundamentals

Color of an object is determined by the nature of the light reflected from it. In 1666, Sir Isaac Newton discovered that when a beam of sunlight passes through a glass prism, the emerging beam of light is not white but consists instead of a continuous spectrum of colors ranging from violet at one end to red at the other. As the following Fig. shows that the color spectrum may be divided into six broad regions: violet, blue, green, yellow, orange, and red.

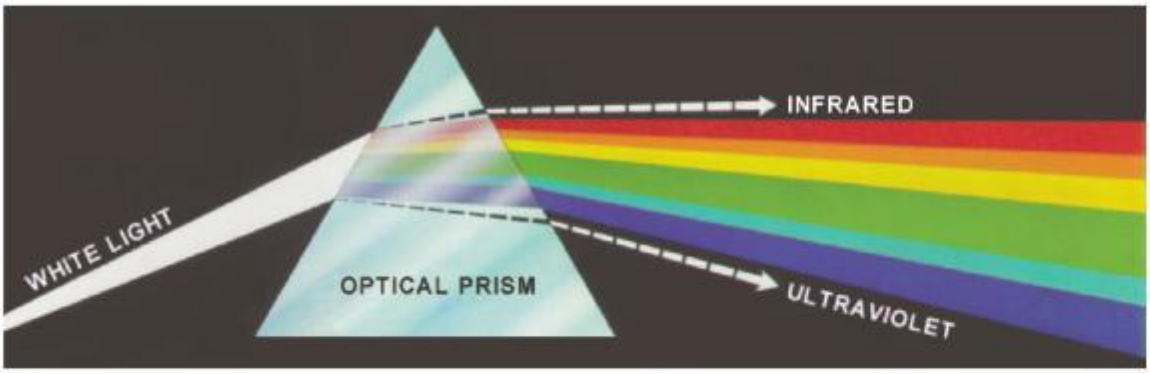


Fig. Color spectrum seen by passing white light through a prism

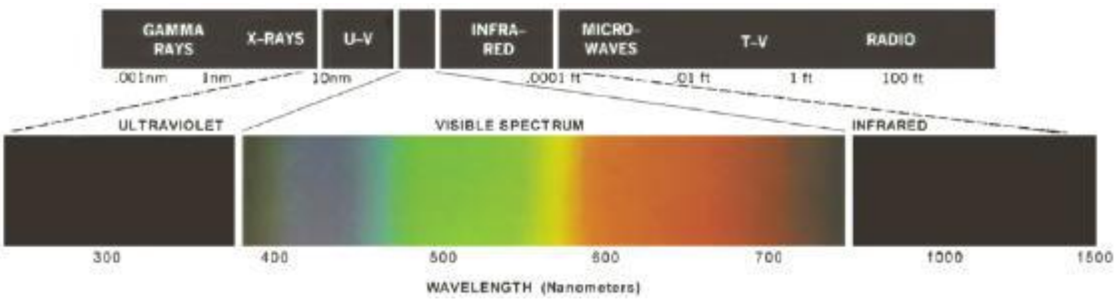


Fig. Wavelengths comprising the visible range of the electromagnetic spectrum

Visible light is composed of a relatively narrow band of frequencies in the electromagnetic spectrum. A body that reflects light that is balanced in all visible wavelengths appears white to the observer. However, a body that favors reflectance in a limited range of the visible spectrum exhibits some shades of color. For example, green objects reflect light with wavelengths primarily in the 500 to 570 nm range while absorbing most of the energy at other wavelengths. Characterization of light is central to the science of color. If the light is achromatic (void of color), its only attribute is its intensity, or amount. Achromatic light is what viewers see on a black and white television set.

Chromatic light spans the electromagnetic spectrum approximately from 400 to 700nm. Three basic quantities are used to describe the quality of a chromatic light source: radiance, luminance, and brightness.

Radiance: Radiance is the total amount of energy that flows from the light source, and it is usually measured in watts (W).

Luminance: Luminance, measured in lumens (lm), gives a measure of the amount of energy an observer perceives from a light source.

Brightness: Brightness is a subjective descriptor that is practically impossible to measure.

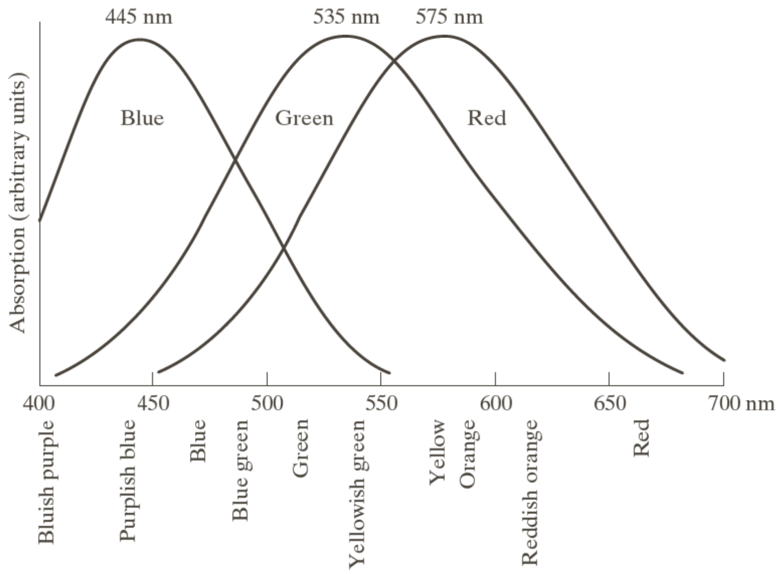
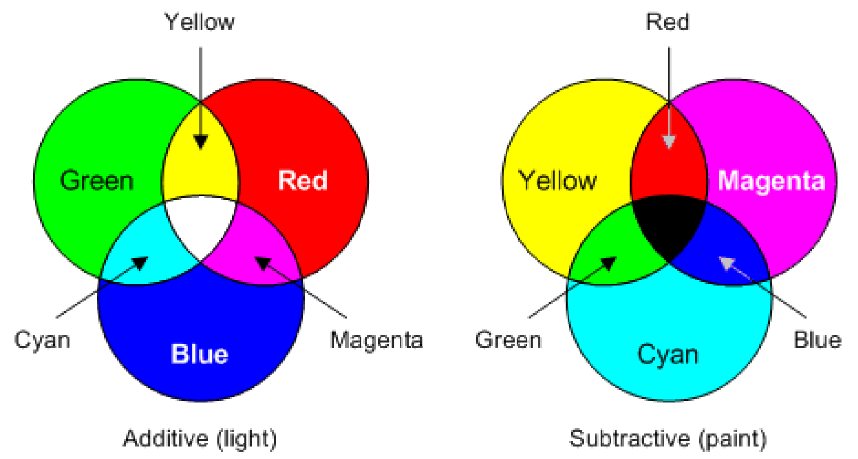


Fig. Absorption of light by the red, green and blue cones in the human eye as a function of wavelength

Cones are the sensors in the eye responsible for color vision. Detailed experimental evidence has established that the 6 to 7 million cones in the human eye can be divided into three principal sensing categories, corresponding roughly to red, green, and blue.

Approximately 65% of all cones are sensitive to red light, 33% are sensitive to green light, and only about 2% are sensitive to blue (but the blue cones are the most sensitive). The above figure shows average experimental curves detailing the absorption of light by the red, green, and blue cones in the eye. Due to these absorption characteristics of the human eye, colors are seen as variable combinations of the so- called primary colors red (R), green (G), and blue (B).

The primary colors can be added to produce the secondary colors of light --magenta (red plus blue), cyan (green plus blue), and yellow (red plus green). Mixing the three primaries or a secondary with its opposite primary color, in the right intensities produces white light.



Additive and subtractive color combinations

Fig. Primary and Secondary Colors of light and pigments

The characteristics generally used to distinguish one color from another are brightness, hue, and saturation. Brightness embodies the chromatic notion of intensity. Hue is an attribute associated with the dominant wavelength in a mixture of light waves. Hue represents dominant color as perceived by an observer. Saturation refers to the relative purity or the amount of white light mixed with a hue. The pure spectrum colors are fully saturated. Colors such as pink (red and white) and lavender (violet and white) are less saturated, with the degree of saturation being inversely proportional to the amount of white light-added. Hue and saturation taken together are called chromaticity, and. therefore, a color may be characterized by its brightness and chromaticity. The amounts of red, green and blue needed to form any particular color are called the tristimulus values and are denoted by red (X), green (Y) and blue (Z) needed to form a particular color. A color can be specified by its trichromatic coefficients and defined as

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

$$y = \frac{Y}{X+Y+Z} \qquad\qquad x+y+z=1$$

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

4.2. Color Models

The purpose of a color model (also called color space or color system) is to facilitate the specification of colors in some standard, generally accepted way. In essence, a color model is a specification of a coordinate system and a subspace within that system where each color is represented by a single point.

Most color models used are oriented either toward hardware or toward applications where color manipulation is goal. The most commonly used hardware-oriented models are RGB (Red, Green, Blue) for color monitors and video cameras, **CMY** (Cyan, Magenta, Yellow) and **CMYK** (CMY + Black) for color printing and HSI (Hue, Saturation, Intensity) which corresponds closely with the way humans describe and interpret color.

4.2.1. The RGB Color Model:

In the RGB model, each color appears in its primary spectral components of red, green, and blue. This model is based on a Cartesian coordinate system. The color subspace of interest is the cube shown in the following figure. In which RGB values are at three corners; cyan, magenta, and yellow are at three other corners; black is at the origin; and white is at the corner farthest from the origin. In this model, the gray scale (points of equal RGB values) extends from black to white along the line joining these two points. The different colors in this model arc points on or inside the cube, and are defined by vectors extending from the origin. For convenience, the assumption is that all color values have been normalized so that the cube shown in the figure is the unit cube. That is, all values of R, G. and B are assumed to be in the range [0, 1].

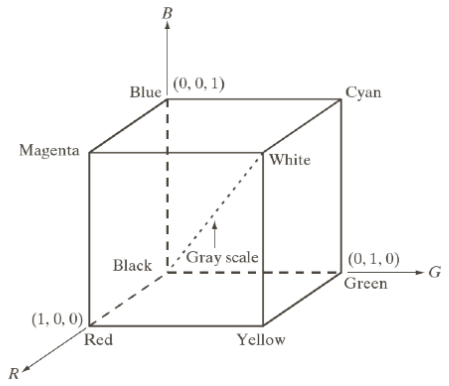


Fig. Schematic of the RGB color cube

Images represented in the RGB color model consist of three component images, one for each primary color. When fed into an RGB monitor, these three images combine on the phosphor screen to produce a composite color image.

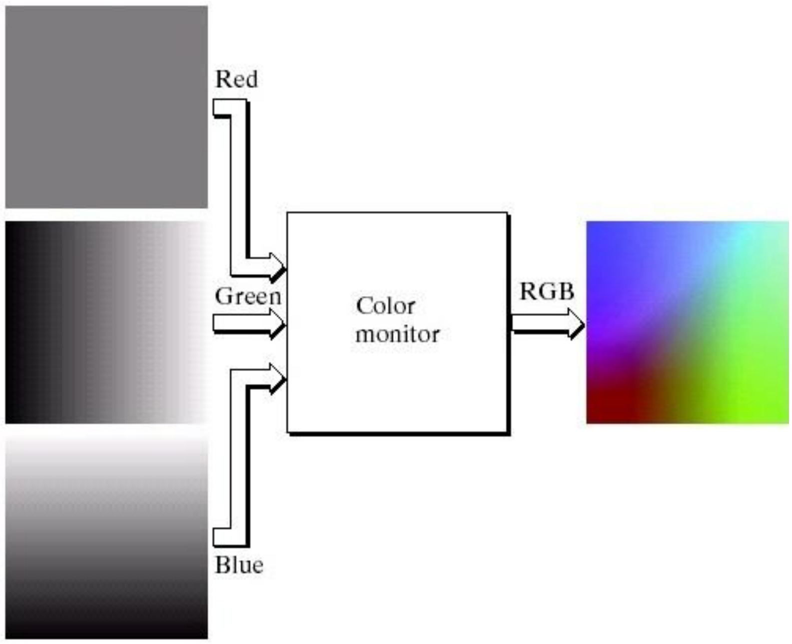


Fig. Generating the RGB image of the cross Sectional color plane

The number of bits used to represent each pixel in RGB space is called the pixel depth. Consider an RGB image in which each of the red, green, and blue images is an 8-bit image. Under these conditions each RGB color pixel [that is, a triplet of values (R, G, B)] is said to have a depth of 24 bits C image planes times the number of bits per plane). The term full-color image is used often to denote a 24-bit RGB color image. The total number of colors in a 24-bit RGB image is $(28)^3 = 16,777,216$.

4.2.2. The CMY and CMYK Color models

Cyan, magenta, and yellow are the secondary colors of light or, alternatively, the primary colors of pigments. For example, when a surface coated with cyan pigment is illuminated with white light, no red light is reflected from the surface. That is, cyan subtracts red light from reflected white light, which itself is composed of equal amounts of red, green, and blue light. Most devices that deposit colored pigments on paper, such as color printers and copiers, require CMY data input or perform an RGB to CMY conversion internally. This conversion is performed using

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

Where, again, the assumption is that all color values have been normalized to the range [0, 1]. The above equation demonstrates that light reflected from a surface coated with pure cyan does not contain red (that is, $C = 1 — R$ in the equation). Similarly, pure magenta does not reflect green, and pure yellow does not reflect blue. So, the RGB values can be obtained easily from a set of CMY values by subtracting the individual CMY values from 1. Equal amounts of the pigment primaries, cyan, magenta, and yellow should produce black. In practice, combining these colors for printing produces a muddy-looking black. So, in order to produce true black, a fourth color, black is added, giving rise to the CMYK color model.

4.2.3. HSI color model

When humans view a color object, we describe it by its hue, saturation, and brightness. Hue is a color attribute that describes a pure color (pure yellow, orange, or red), whereas saturation gives a measure of the degree to which a pure color is diluted by white light. Brightness is a subjective descriptor that is practically impossible to measure. It embodies the achromatic notion of intensity and is one of the key factors in describing color sensation.

Intensity (gray level) is a most useful descriptor of monochromatic images. This quantity definitely is measurable and easily interpretable. The HSI (hue, saturation, intensity) color model, decouples the intensity component from the color-carrying information (hue and Saturation) in a color image. As a result, the HSI model is an ideal tool for developing image processing algorithms based on color descriptions that are natural and intuitive to humans.

In the following figure the primary colors are separated by 120° and the secondary colors are 60° from the primaries, which means that the angle between secondaries is also 120°.

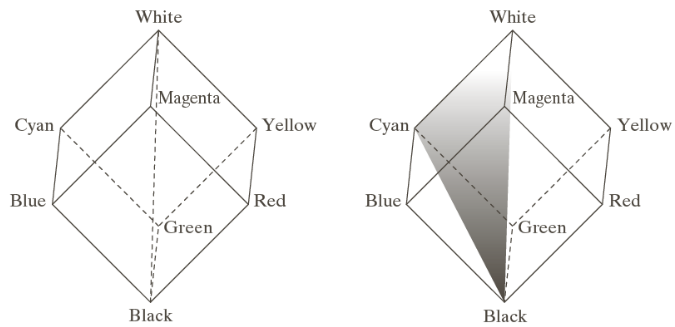


Fig. The relation between RGB and HSI color model

The hue of the point is determined by an angle from some reference point. Usually (but not always) an angle of 0° from the red axis designates 0 hue, and the hue increases counter clockwise from there. The saturation (distance from the vertical axis) is the length of the vector from the origin to the point. The origin is defined by the intersection of the color plane with the vertical intensity axis. The important components of the HSI color space are the vertical intensity axis, the length of the vector to a color point, and the angle this vector makes with the red axis.

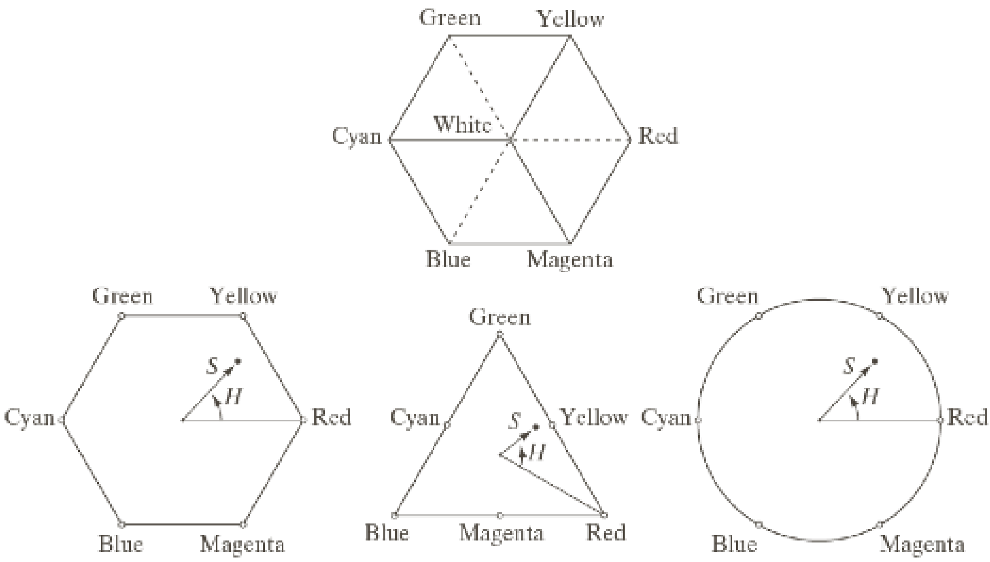


Fig. Hue and saturation in the HSI color model

4.2.4. Conversion from RGB color model to HSI color model

Given an image in RGB color format, the H component of each RGB pixel is obtained using the equation,

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

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

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

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

It is assumed that the RGB values have been normalized to the range [0, 1] and that angle θ is measured with respect to the red axis of the HST space. The SI values are in [0,1] and the H value can be divided by 360o to be in the same range.

4.2.5. Conversion from HSI color model to RGB color model

Given values of HSI in the interval [0,1], one can find the corresponding RGB values in the same range. The applicable equations depend on the values of H. There are three sectors of interest, corresponding to the 120° intervals in the separation of primaries.

RG sector ($0^o \le H < 120°$):

When H is in this sector, the RGB components are given by the equations

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

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

$$G = 3I - (R + B)$$

GB sector ($120^o \le H < 240^o$):

If the given value of H is in this sector, first subtract 120° from it.

$$\mathbf{H = H - 120^0}$$

Then the RGB components are

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

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

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

$$B = 3I - (R + G)$$

BR sector ($240^o \leq H \leq 360^o$):

If H is in this range, subtract 240^o from it

$H = H - 240^0$

Then the RGB components are

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

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

$$R = 3I - (G + B)$$

4.3. Pseudo color image processing

Pseudo color (also called false color) image processing consists of assigning colors to gray values based on a specified criterion. The term pseudo or false color is used to differentiate the process of assigning colors to monochrome images from the processes associated with true color images. The process of gray level to color transformations is known as pseudo color image processing. The two techniques used for pseudo color image processing are,

- Intensity Slicing
- Gray Level to Color Transformation

4.3.1. Intensity Slicing:

The technique of intensity (sometimes called density) slicing and color coding is one of the simplest examples of pseudo color image processing. If an image is interpreted as a 3-D function (intensity versus spatial coordinates), the method can be viewed as one of placing planes parallel to the coordinate plane of the image; each plane then "slices" the function in the area of intersection. The following figure shows an example of using a plane at f(x, y) = li to slice the image function into two levels.

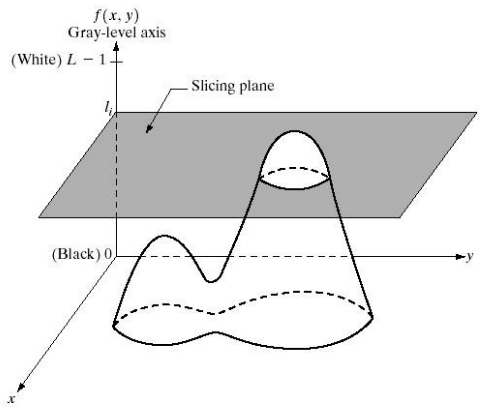


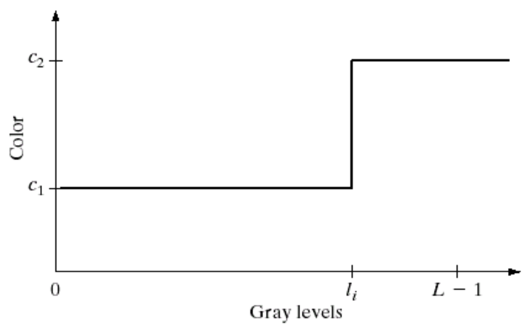
Fig. Geometric interpretation of the intensity slicing technique

If a different color is assigned to each side of the plane shown in the above figure any pixel whose gray level is above the plane will be coded with one color and any pixel below the plane will be coded with the other. Levels that lie on the plane itself may be arbitrarily assigned one of the two colors. The result is a two-color image whose relative appearance can be controlled by moving the slicing plane up and down the gray-level axis.

In general, the technique may be summarized as follows. Let [0, L - 1] represent the gray scale, level l_o represent black [f(x, y) = 0], and level l_{L-1} represent white [f(x, y) = L - 1]. Suppose that P planes perpendicular to the intensity axis are defined at levels $l_1, l_2, \ldots, l_p$.. Then, assuming that $0 < P < L - 1$, the P planes partition the gray scale into P + 1 intervals, V1, V2,..., Vp + 1. Gray-level to color assignments are made according to the relation

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

Where c_k is the color associated with the k^{th} intensity interval V_k defined by the partitioning planes at $l = k - 1$ and $l = k$. An alternative representation defines the same mapping according to the mapping function shown in the following figure. Any input gray level is assigned one of two colors, depending on whether it is above or below the value of *li*. When more levels are used, the mapping function takes on a staircase form.



An alternative representation of the intensity-slicing technique

4.3.2. Gray Level to Color Transformation:

This approach is to perform three independent transformations on the gray level of any input pixel. The three results are then fed separately into the red, green, and blue channels of a color television monitor. This method produces a composite image whose color content is modulated by the nature of the transformation functions. These are transformations on the gray-level values of an image and are not functions of position. In intensity slicing, piecewise linear functions of the gray levels are used to generate colors. On the other hand, this method can be based on smooth, nonlinear functions, which, as might be expected, gives the technique considerable flexibility. The output of each transformation is a composite image.

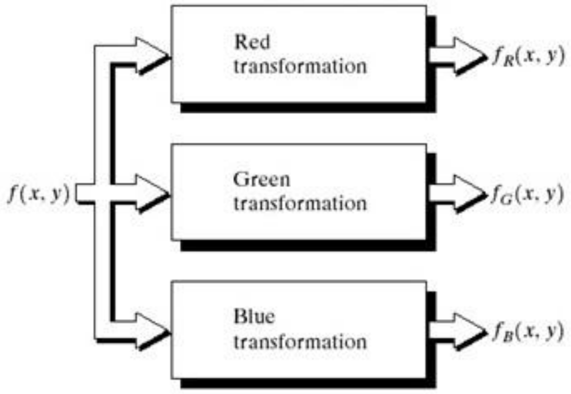


Fig. Functional block diagram for pseudo color image processing

4.4. Full color image processing

Full-color image processing approaches fall into two major categories. In the first category, each component image is processed individually and then forms a composite processed color image from the individually processed components. In the second category, one works with color pixels directly. Because full-color images have at least three components, color pixels really are vectors. For example, in the RGB system, each color point can be interpreted as a vector extending from the origin to that point in the RGB coordinate system.

Let c represent an arbitrary vector in RGB color space:

$$\mathbf{c} = \begin{bmatrix} c_R \\ c_G \\ c_B \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

It indicates that the components of c are simply the RGB components of a color image at a point. If the color components are a function of coordinates (x, y) by using the notation

$$\mathbf{c}(x,y) = \begin{bmatrix} c_R(x,y) \\ c_G(x,y) \\ c_B(x,y) \end{bmatrix} = \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix}$$

For an image of size M × N, there are MN such vectors, c(x, y), for x = 0,1, 2,...,M- l; y = 0,1,2,...,N- 1. In order for per-color-component and vector-based processing to be equivalent, two conditions have to be satisfied: First, the process has to be applicable to both vectors and scalars. Second, the operation on each component of a vector must be independent of the other components.

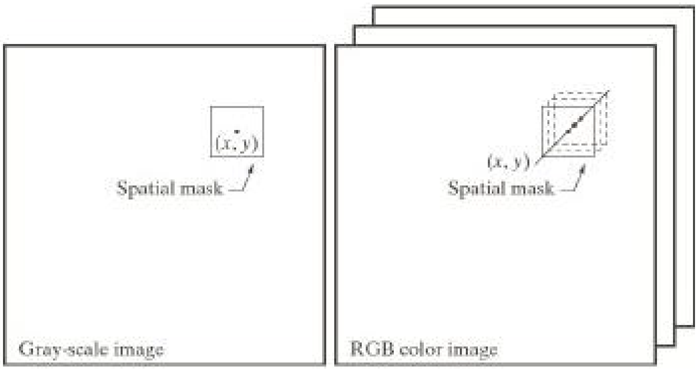


Fig. Spatial masks for (a)gray-scale and (b) RGB color images.

The above figure shows neighborhood spatial processing of gray-scale and full-color images. Suppose that the process is neighborhood averaging. In Fig. (a), averaging would be accomplished by summing the gray levels of all the pixels in the neighborhood and dividing by the total number of pixels in the neighborhood. In Fig. (b), averaging would be done by summing all the vectors in the neighborhood and dividing each component by the total number of vectors in the neighborhood. But each component of the average vector is the sum of the pixels in the image corresponding to that component, which is the same as the result that would be obtained if the averaging were done on a per-color-component basis and then the vector was formed.

4.5. Color Transformations

Color transformations deal with processing the components of a color image within the context of a single color model, without converting components to different color space.

4.5.1. Formulation

We can model color transformations using the expression

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

Where f(x, y) is color input image, g(x, y) is the transformed color output image and T is the operator over a spatial neighborhood of (x, y). Each f(x, y) component is a triplet in the chosen color space. For a given transformation the cost of converting from one color space to another is also a factor to implement it. Hence, we wish to modifying intensity of an image in different color spaces, using the transform

$$g(x, y) = k\, f(x, y)$$

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

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

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

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

In RGB color space,

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

In HSI color space,

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

In CMY color space,

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

4.5.2. 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. Color complements are used to enhance the details in dark regions of a color image.

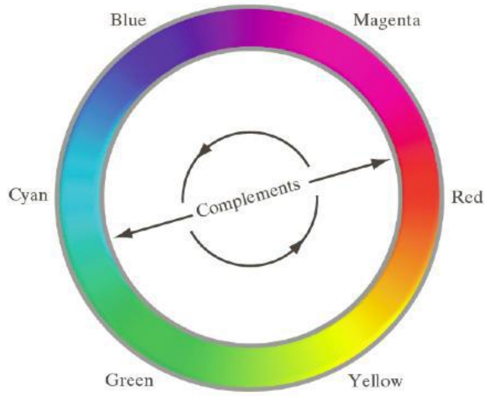


Fig. Complements on the Circle

4.5.3. Color Slicing

Color slicing is the process of highlighting a specific range of colors in an image is useful for separating object from their surroundings. It is more complex than gray-level slicing, due to multiple dimensions for each pixel. This can be done by selecting the region that needs to be high spotted in a cube of width '*w*'. The outside region must be mapped with a neutral color. Then the transformation is given by

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

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

or

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

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

4.5.4. Tone and Color Corrections

Effectiveness of these transformations judged ultimately in print. But developed, refined and evaluated on monitors. Need to maintain a high degree of color consistency between monitors used and eventual output devices. *Device-independent* color model, relating the color gamut's of the monitors and output devices. The success of this approach is a function of the quality of the color profiles used to map each device to the model and the model itself. The model of choice for many color management system (CMS) is the CIE *L*a*b* model.

$$L^* = 116 \cdot h\left(\frac{Y}{Y_w}\right) - 16$$

$$a^* = 500\left[h\left(\frac{X}{X_w}\right) - h\left(\frac{Y}{Y_w}\right)\right]$$

$$b^* = 200\left[h\left(\frac{Y}{Y_w}\right) - h\left(\frac{Z}{Z_w}\right)\right]$$

where

$$h(q) = \begin{cases} \sqrt[3]{q} & q > 0.008856 \\ 7.787q + 16/116 & q \le 0.008856 \end{cases}$$

$X_w, Y_w,$ and Z_w are reference white tristimulus values

Like the HIS system, the *L*a*b* system is an excellent decoupler of intensity (represented by lightness L^*) and color (represent by a^* for red minus green and b^* for green minus blue). The tonal range of an image, also called its *key type*, refer to its general distribution of color intensities. Most of the information in *high-key* images are located predominantly at low intensities; *middle-key* images lie in between.

4.5.5. Histogram Processing

Histogram processing transformations can be applied to color images in an automated way. As might be expected, it is generally unwise to histogram equalize the component of a color image independently. This results in erroneous color. A more logical approach is to spread the color intensities uniformly, leaving the colors themselves (e.g., hues) unchanged. The HSI color space is ideally suited to this type of approach.

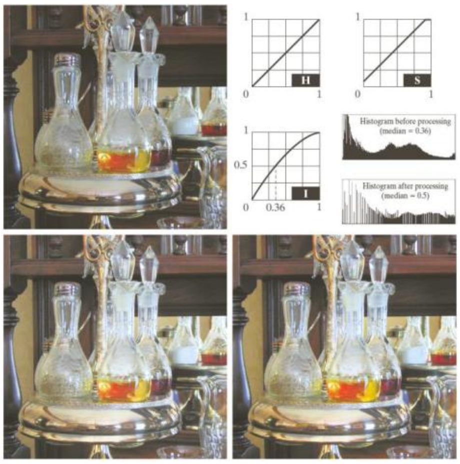


Fig. Histogram Equalization in the HSI Color Space

4.6. Color segmentation process

Segmentation is a process that partitions an image into regions and partitioning an image into regions based on color is known as color segmentation.

Segmentation in HSI Color Space:

If anybody wants to segment an image based on color, and in addition, to carry out the process on individual planes. It is natural to think first of the HSI space because color is conveniently represented in the hue image. Typically, saturation is used as a masking image in order to isolate further regions of interest in the hue image. The intensity image is used less frequently for segmentation of color images because it carries no color information.

Segmentation in RGB Vector Space:

Although, working in HSI space is more intuitive, segmentation is one area in which better results generally are obtained by using RGB color vectors. The approach is straightforward. Suppose that the objective is to segment objects of a specified color range in an RGB image. Given a set of sample color point's representative of the colors of interest, we obtain an estimate of the "average" color that we wish to segment. Let this average color be denoted by the RGB vector **a**. The objective of segmentation is to classify each RGB pixel in a given image as having a color in the specified range or not. In order to perform this comparison, it is necessary to have a measure of similarity. One of the simplest measures is the Euclidean distance. Let **z** denote an arbitrary point in RGB space. **z** is similar to **a** if the distance between them is less than a specified threshold, Do. The Euclidean distance between **z** and **a** is given by

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

Where the subscripts R, G, and B, denote the RGB components of vectors **a** and **z.** The locus of points such that $D(z, a) \leq D_o$ is a solid sphere of radius D_o. Points contained within or on the surface of the sphere satisfy the specified color criterion; points outside the sphere do not. Coding these two sets of points in the image with, say, black and white, produces a binary segmented image.

A useful generalization of previous equation is a distance measure of the form

$$D(z, a) = [(z-a)^{T} C^{-1} (z-a)]^{1/2}$$

Where C is the covariance matrix1 of the samples representative of the color to be segmented and the above equation represents an ellipse with color points such that $D(z, a) \leq D_o$.

PREVIOUS QUESTIONS

1. Explain about RGB, CMY and CMYK color models?
2. What is Pseudocolor image processing? Explain.
3. Explain about color image smoothing and sharpening.
4. Explain about histogram processing of color images.
5. Explain the procedure of converting colors from RGB to HSI and HSI to RGB.
6. Discuss about noise in color images.
7. Explain about HSI colour model.
8. Consider the following RGB triplets. Convert each triplet to CMY and YIQ
   i) (1 1 0) ii) (1 1 1) iii). (1 0 1)
9. Explain in detail about how the color models are converted to each other.
10. Discuss about color quantization and explain about its various types.
11. What are color complements? How they are useful in image processing.
12. What is meant by Luminance, brightness, Radiance & trichromatic Coefficients.