

Color Image Processing

The use of color is important in image processing because:

- Color is a powerful descriptor that simplifies object identification and extraction.
- Humans can discern thousands of color shades and intensities, compared to about only two dozen shades of gray.

Color image processing is divided into two major areas:

- Full-color processing: images are acquired with a full-color sensor, such as a color TV camera or color scanner.
- Pseudocolor processing: The problem is one of assigning a color to a particular monochrome intensity or range of intensities.

Color Fundamentals

Colors are seen as variable combinations of the *primary colors* of light: red (R), green (G), and blue (B). The primary colors can be mixed to produce the *secondary colors*: magenta (red+blue), cyan (green+blue), and yellow (red+green). Mixing the three primaries, or a secondary with its opposite primary color, produces white light.

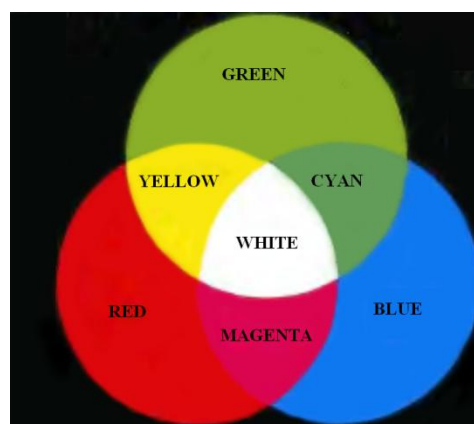


Figure 15.1 Primary and secondary colors of light

RGB colors are used for color TV, monitors, and video cameras.

However, the primary colors of pigments are cyan (C), magenta (M), and yellow (Y), and the secondary colors are red, green, and blue. A proper combination of the three pigment primaries, or a secondary with its opposite primary, produces black.

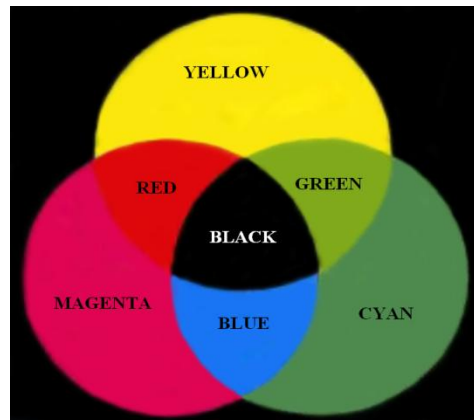


Figure 15.2 Primary and secondary colors of pigments

CMY colors are used for color printing.

Color characteristics

The characteristics used to distinguish one color from another are:

- Brightness: means the amount of intensity (i.e. color level).
- Hue: represents dominant color as perceived by an observer.
- Saturation: refers to the amount of white light mixed with a hue.

Color Models

The purpose of a color model is to facilitate the specification of colors in some standard way. 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. Color models most commonly used in image processing are:

- RGB model for color monitors and video cameras
- CMY and CMYK (cyan, magenta, yellow, black) models for color printing
- HSI (hue, saturation, intensity) model

The RGB color model

In this model, each color appears in its primary colors red, green, and blue. This model is based on a Cartesian coordinate system. The color subspace is the cube shown in the figure below. The different colors in this model are points on or inside the cube, and are defined by vectors extending from the origin.

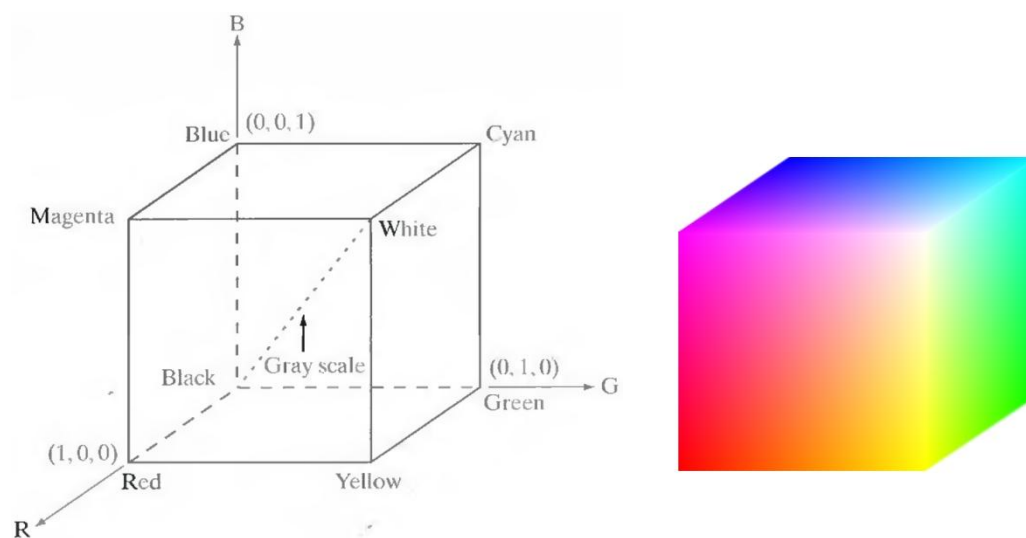


Figure 15.3 RGB color model

All color values R, G, and B have been normalized in the range $[0, 1]$.

However, we can represent each of R, G, and B from 0 to 255.

Each RGB color image consists of three component images, one for each primary color as shown in the figure below. These three images are combined on the screen to produce a color image.

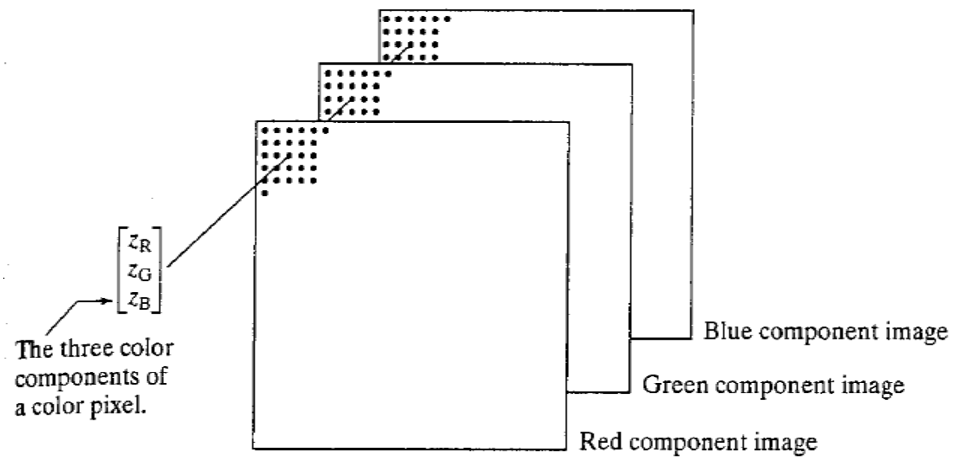


Figure 15.4 Scheme of RGB color image

The total number of bits used to represent each pixel in RGB image is called *pixel depth*. For example, in an RGB image if each of the red, green, and blue images is an 8-bit image, the pixel depth of the RGB image is 24-bits. The figure below shows the component images of an RGB image.

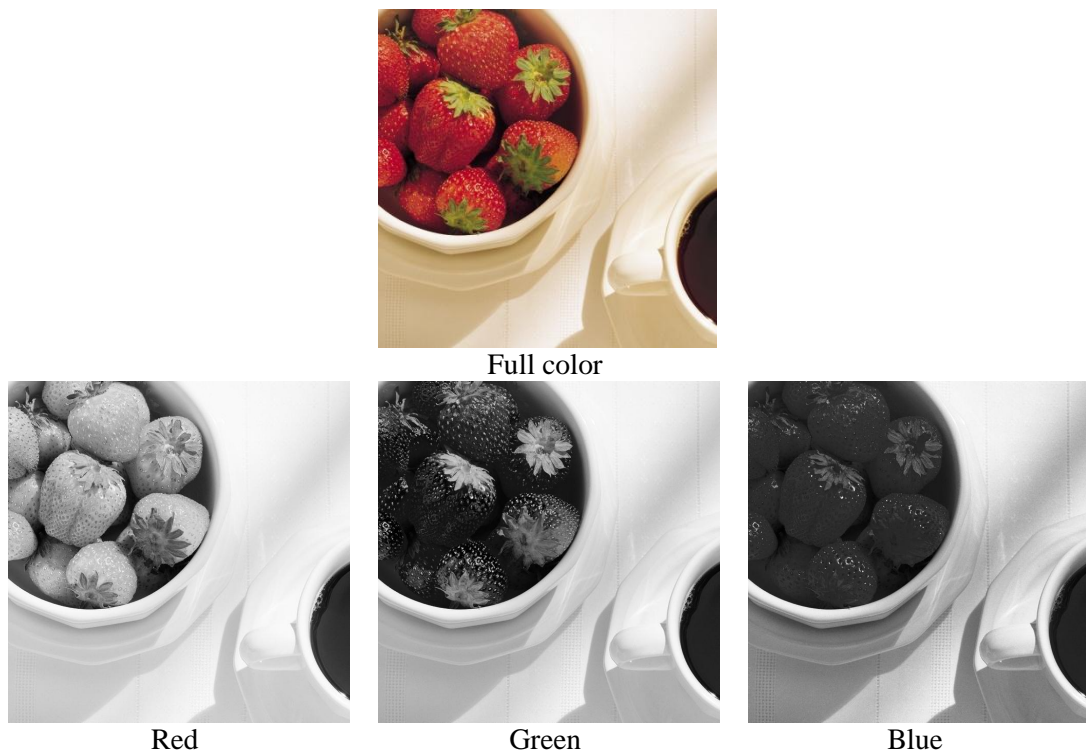


Figure 15.5 A full-color image and its RGB component images

The CMY and CMYK color model

Cyan, magenta, and yellow are the primary colors of pigments. Most printing devices such as color printers and copiers require CMY data input or perform an RGB to CMY conversion internally. This conversion is performed using the equation

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

where, all color values have been normalized to the range [0, 1].

In printing, combining equal amounts of cyan, magenta, and yellow produce muddy-looking black. In order to produce true black, a fourth color, black, is added, giving rise to the CMYK color model.

The figure below shows the CMYK component images of an RGB image.

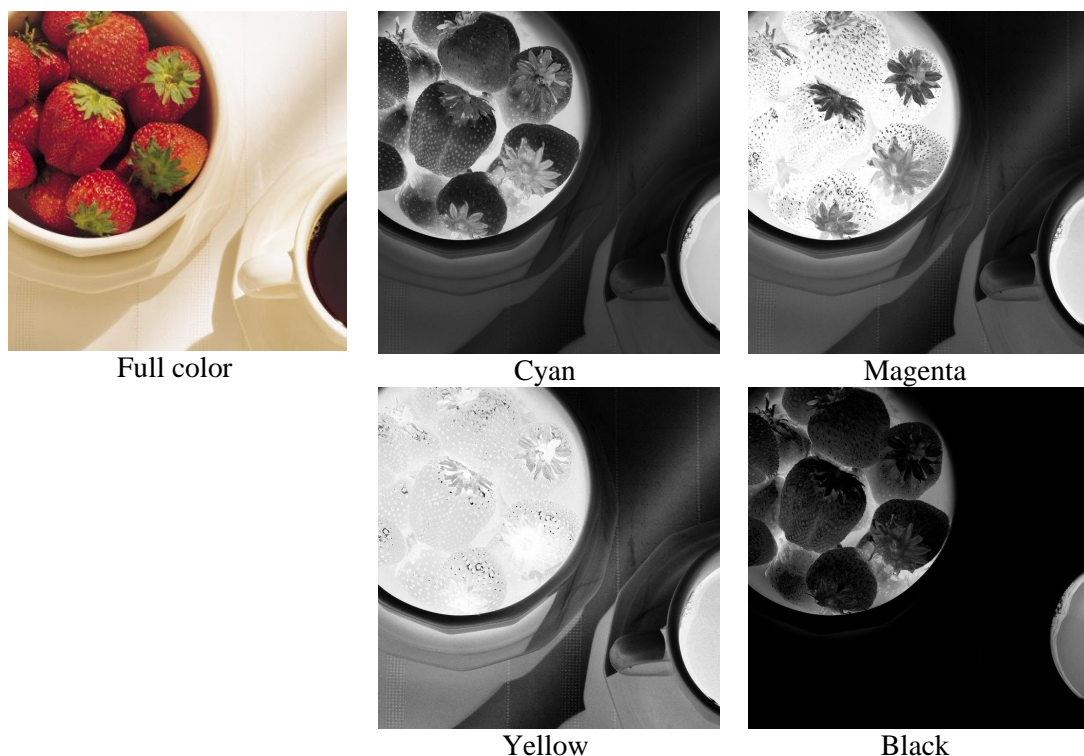


Figure 15.6 A full-color image and its CMYK component images

The HSI color model

The RGB and CMY color models are not suited for describing colors in terms of human interpretation. When we view a color object, we describe it by its hue, saturation, and brightness (intensity). Hence the HSI color model has been presented. The HSI model decouples the intensity component from the color-carrying information (hue and saturation) in a color image. As a result, this model is an ideal tool for developing color image processing algorithms.

The hue, saturation, and intensity values can be obtained from the RGB color cube. That is, we can convert any RGB point to a corresponding point in the HSI color model by working out the geometrical formulas.

Converting colors from RGB to HSI

The hue H is given by

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

Where

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

The saturation S is given by

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

The intensity I is given by

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

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

Converting colors from HSI to RGB

The applicable equations depend on the value of H :

If $0^\circ \leq H < 120^\circ$:

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

If $120^\circ \leq H < 240^\circ$:

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

If $240^\circ \leq H \leq 360^\circ$:

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

The next figure shows the HSI component images of an RGB image.

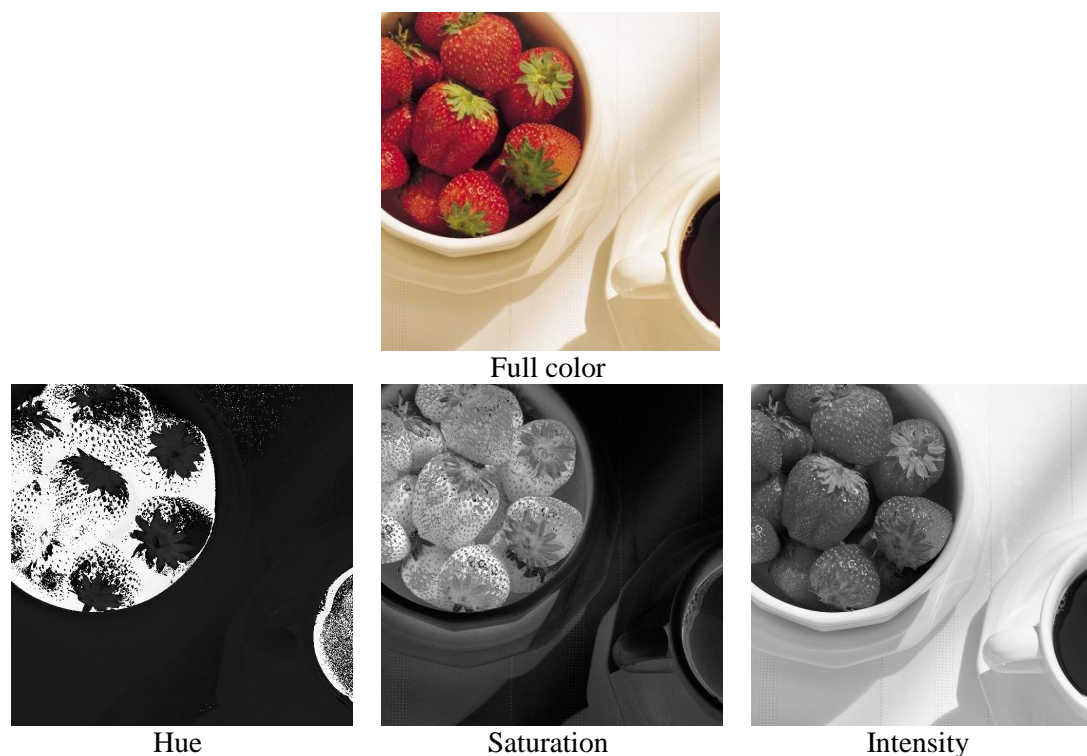


Figure 15.7 A full-color image and its HSI component images

Basics of Full-Color Image Processing

Full-color image processing approaches fall into two major categories:

- Approaches that process each component image individually and then form a composite processed color image from the individually processed components.
- Approaches that work with color pixels directly.

In full-color images, color pixels really are vectors. For example, in the RGB system, each color pixel can be expressed as

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

For an image of size $M \times N$, there are MN such vectors, $c(x, y)$, for $x = 0, 1, 2, \dots, M-1$; $y = 0, 1, 2, \dots, N-1$.

Color Transformation

As with the gray-level transformation, we model color transformations using the expression

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

where $f(x, y)$ is a color input image, $g(x, y)$ is the transformed color output image, and T is the color transform.

This color transform can also be written

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

For example, we wish to modify the intensity of the image shown in Figure 14.8(a) using

$$g(x, y) = 0.7f(x, y)$$

- In the RGB color space, three components must be transformed:

$$s_i = 0.7r_i \quad i = 1, 2, 3$$

- In CMY space, also three component images must be transformed

$$s_i = 0.7r_i + 0.3 \quad i = 1, 2, 3$$

- In HSI space, only intensity component r_3 is transformed

$$s_3 = 0.7r_3$$



(a)

(b)

Figure 15.8 (a) Original image. (b) Result of decreasing its intensity