Image Types

The image types we will consider are:

1. Binary Images

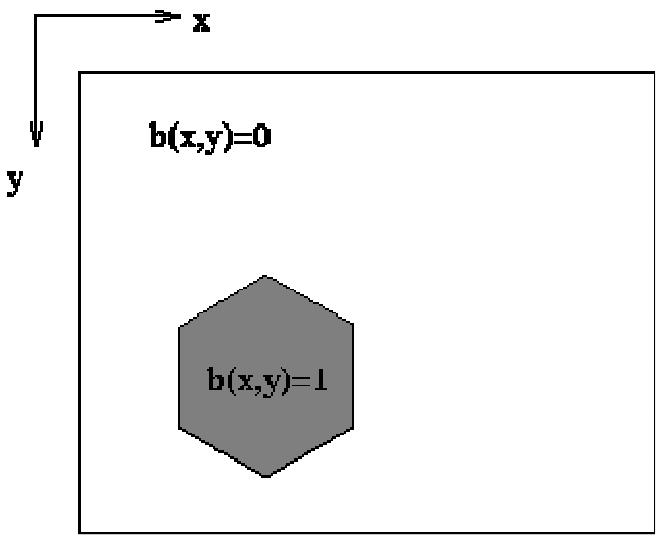
Binary images are the simplest type of images and can take on two values, typically black and white, or '0' and '1'. A binary image is referred to as a 1 bit/pixel image because it takes only 1 binary digit to represent each pixel.

These types of images are most frequently in computer vision application where the only information required for the task is general shapes, or outlines information. For example, to position a robotics gripper to grasp an object or in optical character recognition (OCR).

Binary images are often created from gray-scale images via a threshold value is turned white ('1'), and those below it are turned black ('0').

We define the characteristic function of an object in an image to be

$$b(x,y)\begin{cases} = 1 & \text{for points on the object} \\ = 0 & \text{for background points.} \end{cases}$$



(a)

(b)

Figure(1) (a) binary image representation (b) binary Lenna image

- Each pixel is stored as a single bit (0 or 1)
- A 640 x 480 monochrome image requires 37.5 KB of storage.

2-Gray Scale Images

Gray _scale images are referred to as monochrome, or one-color image. They contain brightness information only brightness information only, no color information. **The number of different brightness level available. The typical image contains 8 bit/ pixel (data, which allows us to have (0-255) different brightness (gray) levels**. The 8 bit representation is typically due to the fact that the byte, which corresponds to 8-bit of data, is the standard small unit in the world of digital computer.

Figure Examples of gray-scale images

- Each pixel is usually stored as a byte (value between 0 to 255)

- A 640 x 480 greyscale image requires over 300 KB of storage.

Figure 2.3 shows a grayscale image and a 6 × 6 detailed region, where brighter pixels correspond to larger values.

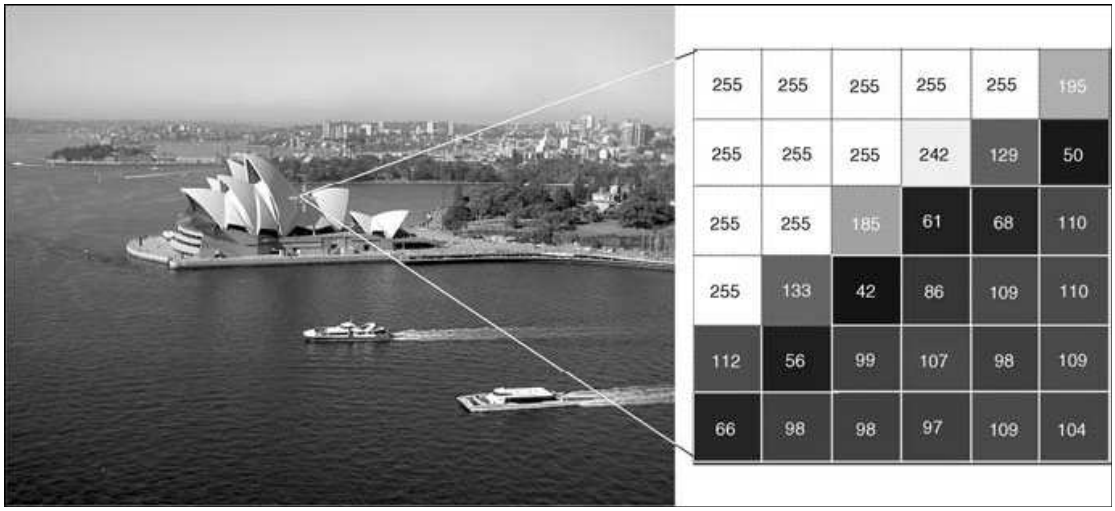


FIGURE 2.3 A grayscale image and the pixel values in a 6 × 6 neighborhood.

3. COLOR IMAGES

Representation of color images is more complex and varied. The two most common ways of storing color image contents are
1)RGB representation—in which each pixel is usually represented by a 24-bit number containing the amount of its red (R), green (G), and blue (B) components.
2) **indexed representation**—where a 2D array contains indices to a color palette (or lookup table - (LUT)).

24-Bit (RGB) Color Images Color images can be represented using three 2D arrays of same size, one for each color channel: red (R), green (G), and blue (B) (Figure 2.4).1 Each array element contains an 8-bit value, indicating the amount of red, green, or blue at that point in a [0, 255] scale. The combination of the three 8-bit values into a 24-bit number allows 224 (16,777,216, usually referred to as 16 million or 16 M) color combinations. An alternative representation uses 32 bits per pixel and includes a fourth channel, called the alpha channel, that provides a measure of transparency for each pixel and is widely used in image editing effects.

The following figure we see a representation of a typical RGB color image.

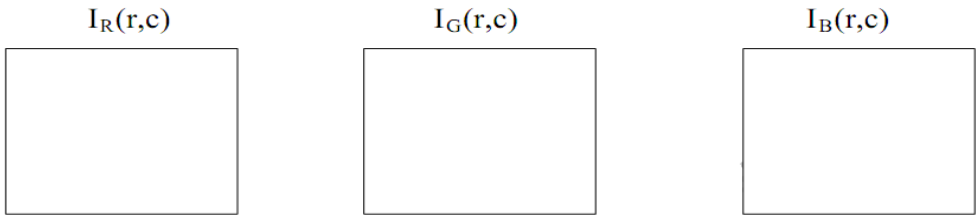


Figure ( ) Typical RGB color image can be thought as three separate images $I_R(r,c)$,$I_G(r,c)$,$I_B(r,c)$ [1]

The following figure illustrate that in addition to referring to arrow or column as a vector, we can refer to a single pixel red ,green, and blue values as a color pixel vector –(R,G,B).

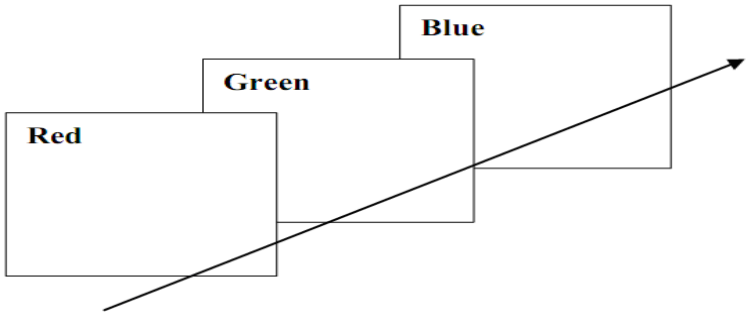
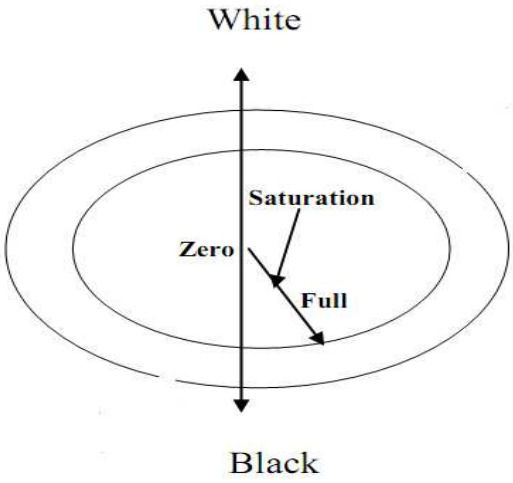


Figure ( ) A color pixel vector consists of the red, green and blue pixel values (R, G, B) at one given row/column pixel

For many applications, RGB color information is transformed into mathematical space that that decouples the brightness information from the color information.

The hue/saturation /lightness (HSL) color transform allows us to describe colors in terms that we can more readily understand.



The lightness is the brightness of the color, and the hue is what we normally think of as “color” and the hue (ex: green, blue, red, and orange).

The saturation is a measure of how much white is in the color (ex: Pink is red with more white, so it is less saturated than a pure red). [Most people relate to this method for describing color}.

FIGURE 2.4 Color image (a) and its R (b), G (c), and B (d) components.

Example of 24-Bit Colors Image

- Each pixel is represented by three bytes (e.g., RGB)
- Supports 256 x 256 x 256 possible combined colors (16,777,216)
- A 640 x 480 24-bit color image would require 921.6 KB of storage

Indexed Color Images: A problem with 24-bit color representations is backward compatibility with older hardware that may not be able to display the 16 million colors simultaneously. A solution—devised before 24-bit color displays and video cards were widely available—consisted of an indexed representation, in which a 2D array of the same size as the image contains indices (pointers) to a *color palette* (or *color map*) of fixed maximum size (usually 256 colors). The color map is simply a list of colors used in that image. Figure 2.5 shows an indexed color image and a 4 × 4 detailed region, where each pixel shows the index and the values of R, G, and B at the color palette entry that the index points to.

Example of 8-Bit Color Image

- One byte for each pixel
- Supports 256 out of the millions s possible, acceptable color quality
- Requires Color Look-Up Tables (LUTs)
- A 640 x 480 8-bit color image requires 307.2 KB of storage (the same as 8-bit grayscale)

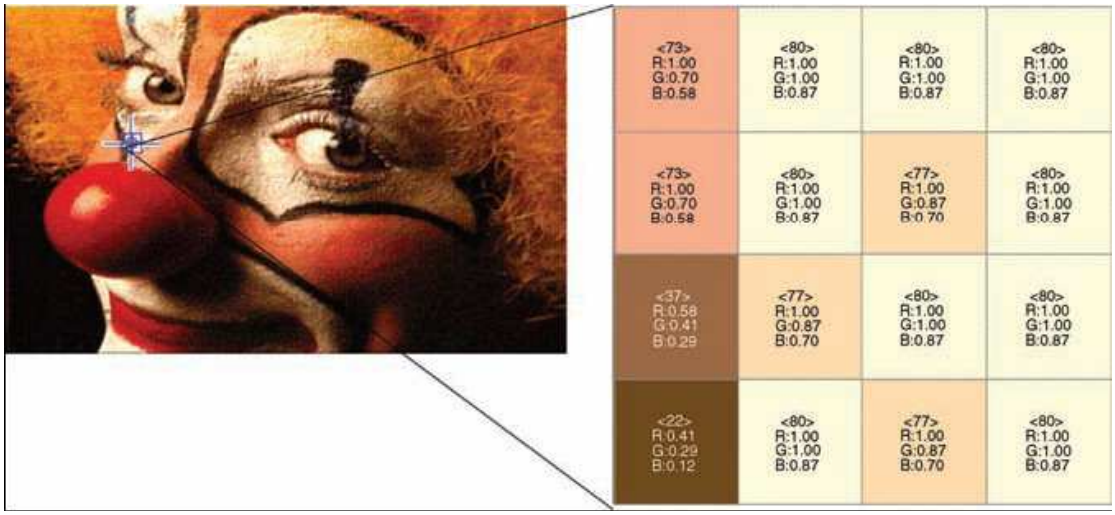


FIGURE 2.5 An indexed color image and the indices in a 4 $\times$ 4 neighborhood. Original image