Digital Image Representation

1. Digital Images

The result of these three stages is a description of the image in the form of a 2D, f(x,y) is approximated by equally spaced samples in the form of an NxM array where each element is a discrete quantity ordered matrix of integers (Fig. 1).

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ f(1,0) & f(1,1) & \cdots & f(1,M-1) \\ . & & & \\ f(N-1,0) & f(N-1,1) & \cdots & f(N-1,M-1) \end{bmatrix}$$

Fig.1: Image as discrete function

A digital image I (Fig.2) is a 2D function that maps from the domain of integer coordinates N × N to a range of possible pixel values P such that $I(u, v) \in$ P and $u, v \in$ N.

$F(x, y)$

148	123	52	107	123	162	172	123	64	89	···
147	130	92	95	98	130	171	155	169	163	···
141	118	121	148	117	107	144	137	136	134	···
82	106	93	172	149	131	138	114	113	129	···
57	101	72	54	109	111	104	135	106	125	···
138	135	114	82	121	110	34	76	101	111	···
138	102	128	159	168	147	116	129	124	117	···
113	89	89	109	106	126	114	150	164	145	···
120	121	123	87	85	70	119	64	79	127	···
145	141	143	134	111	124	117	113	64	112	···
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

$I(u, v)$

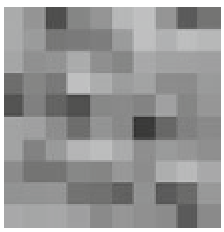


Fig. 2: The transformation of a continuous grayscale image f(x, y) to a discrete digital image

Now we are ready to transfer the image to our computer so that we can save, compress, and otherwise manipulate it into the file format of our choice. At this point, it is no longer important to us how the image originated since it is now a simple 2D array of numerical data.

2. Image Size and Resolution

The *size* of an image is determined directly from the *width M* (number of columns) and the *height N* (number of rows) of the image matrix *I*.

$Size = MxN$ if we assume rectangular image, and

if N=M

$$Size = NxN$$

The resolution of an image specifies the spatial dimensions of the image in the real world and is given *as the number of image elements per measurement;* for example, *dots per inch* (dpi) or *lines per inch* (lpi) for print production, or in *pixels per kilometer* for satellite images.

In most cases, the resolution of an image is the same in the horizontal and vertical directions, which means that the image elements are square (Fig.3)

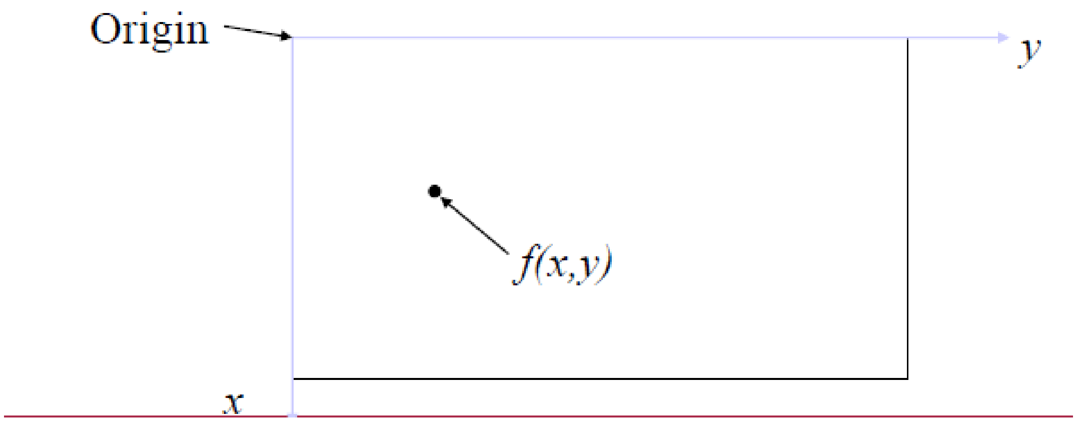
Pixel replication occurs as resolution is decreased

Fig.3: Image resolution

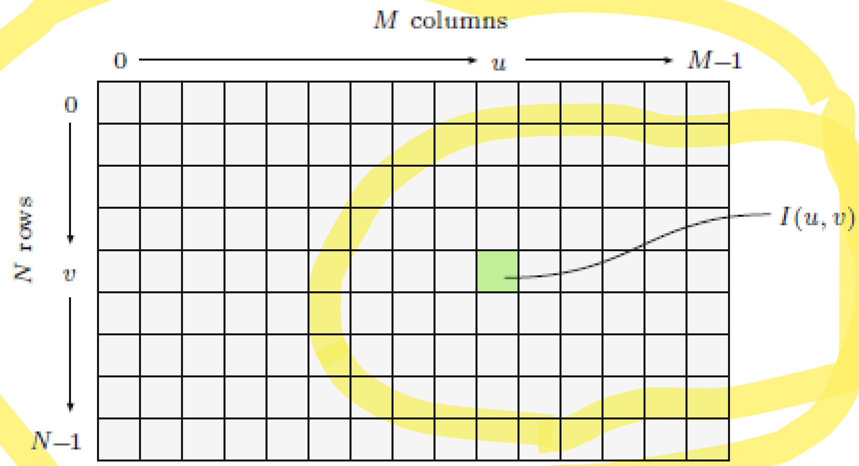
The spatial resolution of an image may not be relevant in many basic image processing steps, such as point operations or filters. Precise resolution information is, however, important in cases where geometrical elements such as circles need to be drawn on an image or when distances within an image need to be measured. For these reasons, most image formats and software systems designed for professional applications rely on precise information about image resolution.

3. Image Coordinate System

In order to know which position on the image corresponds to which image element, we need to impose a coordinate system. Contrary to normal mathematical conventions, in image processing the coordinate system is usually flipped in the vertical direction; that is, the *y*-coordinate runs from top to bottom and the origin lies in the upper left corner (Fig. 4a, and 4b).



(a)



(b)

Fig.4: Image coordinate

We start the numbering of rows and columns at zero for practical reasons.

4. Image Types

The information within an image element (pixel) depends on the data type used to represent it. Pixel values are practically always binary words of length *b* so that a pixel can represent any of 2^b different values.

The value b is called the bit depth (or just "depth") of the image. The exact bit-level layout of an individual pixel depends on the kind of image; for example, binary, grayscale, or RGB color.

a. Binary Image

Binary images are a special type of intensity image(Fig.5), a binary image is referred to as a **1 bit/pixel** image because it takes only 1 binary digit to represent each pixel. where pixels can only take on one of two values, black or white (i.e.,b=1).

$$N_g = 2^b$$

b=1, therefore : $N_g = 2^1$ two gray levels (0,1)

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').

$$1 \leq Thr. < 0$$

Example:

$$I(r,c) = \begin{bmatrix} 122 & 128 & 145 & 133 & 132 \\ 129 & 134 & 144 & 139 & 140 \\ 120 & 122 & 139 & 136 & 132 \\ 119 & 123 & 122 & 131 & 137 \end{bmatrix}$$

Thr.Value= 134

$$I(r,c) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Thr.Value=136

$$I(r,c) = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Thr.Value=123

$$I(r,c) = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Thr.Value=129

$$I(r,c) = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Binary images are often used for representing line graphics, archiving documents, encoding fax transmissions, and of course in electronic printing.

Fig.5: Binary image

b. Grayscale Images (Intensity Images)

Gray _scale images are referred to as monochrome, or one-color image. The image data in a grayscale image consist of a single channel that represents the intensity, brightness, or density of the image. In most cases, only positive values make sense, as the numbers represent the intensity of light energy or density of film and thus cannot be negative, so typically whole integers in the range 0, . . . , 2^{b-1} are used.

For example, a typical grayscale image uses *b* = 8 bits (1 byte) per pixel and intensity values in the range 0, . . . , 255, where the value 0 represents the minimum brightness (black) and 255 the maximum brightness (white).

For many professional photography and print applications, as well as in medicine and astronomy, 8 bits per pixel is not sufficient. Image depths of 12, 14, and even 16 bits are often encountered in these domains (Fig.6).

Fig.6: Gray Scale Image

c. Color images

Color image can be modeled as three band monochrome image data, where each band of the data corresponds to a different color. Most color images are based on the primary colors red, green, and blue RGB), typically making use of 8 bits for each color component (Fig.7)

Fig.7: Color images

In these color images, each pixel requires 3 ×8 = 24 bits to encode all three components, and the range of each individual color component is [0, 255],(Fig.8).

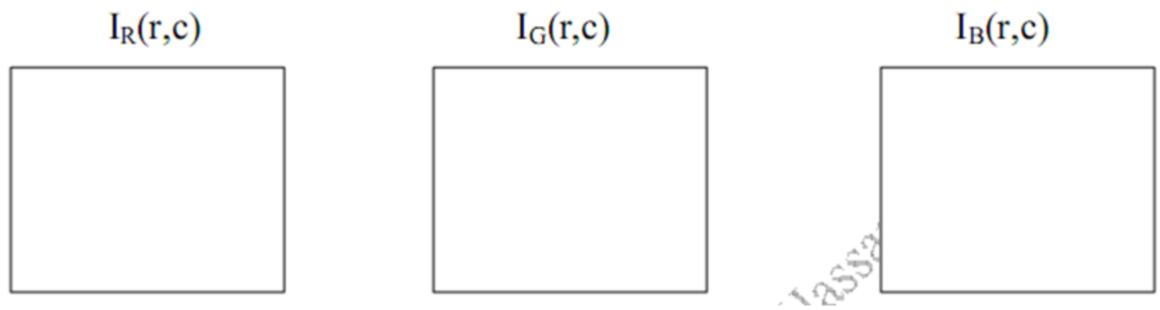


Fig.8: Typical RGB color image can be thought as three

Separate images IR(r,c),IG(r,c),IB(r,c)

Fig.9 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).

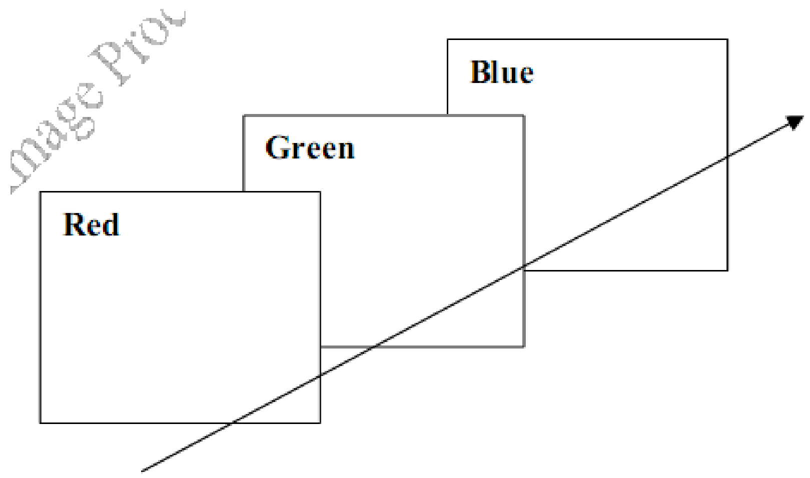


Fig.9:A color pixel vector consists of the red, green and blue pixel values (R, G, B) at one given row/column pixel coordinate(r, c)

The actual information stored in the digital image data is brightness information in each spectral band. When the image is displayed, the corresponding brightness information is displayed on the screen by picture elements that emit light energy corresponding to that particular color.

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.

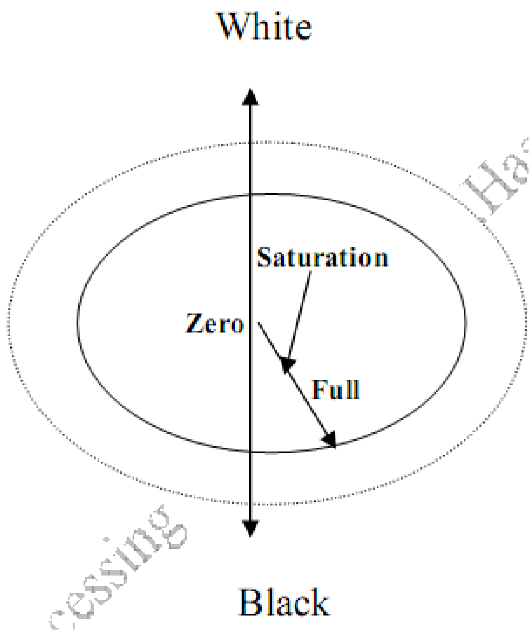


Fig19: HSL Color Space

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}.

Example: “a deep, bright orange” would have a large intensity (“bright”), a hue of “orange”, and a high value of saturation (“deep”).we can estimate this color in our minds, but if we defined this color in terms of its RGB components, R=245, G=110 and B=20. Modeling the color information creates a more people oriented way of describing the colors.

d. Multispectral Images

Multispectral images typically contain information outside the normal human perceptual range. This may include infrared, ultraviolet, X-ray, acoustic or radar data. Source of these types of image include satellite systems underwater sonar systems and medical diagnostics imaging systems. As with intensity images, color images with 30, 36, and 42 bits per pixel are commonly used in professional applications.

5. Important Concepts

1. Image size

Image size of a bitmapped image can be described by the horizontal (H) and vertical (V) pixel count. The total number of pixels in an image is found by multiplying the horizontal and vertical pixel counts:

Total pixel count=HxV

2. Color Depth

Each pixel of the image contains unique color information. The mount of color information is the color depth therefore; it is described in the unit of bits. Where

b= number of bits.

2^b =number of possible display colors.

In a 1 bit image (b=1) each pixel has either a 0 or 1 to code color so only two colors (2^1=2) black or white.

An 8-bit image uses 8 places of binary code to code for the colors. That allows a palette of 2^8=256 colors or 256 shades of gray.

A 24-bit color image works with a palette of over 16.7 million colors (i.e., 2^{24}=16700000).

3. Raw File Size

The image size combined with color depth gives the raw file size, the raw file size can be thought of as a volume. We multiply the horizontal (H) pixel count by the vertical (V) count by the color depth (D) to get the raw file size:

Raw file size=HxVxD

Because all three of these variables are multiplied, an increase in any of three adds to the file size.

<u>Problem</u> (HW)

What is the size of an 8-bit image which is 220 horizontal pixel, by 180 vertical pixel? (answer in kbyte)

Answer:

Raw file size=HxVxD

220pixelsx180pixelsx8bit/pixel=316,800 bits

316,800 bits/8 bits/byte=39,600 bytes

39600/1024=40kbytes