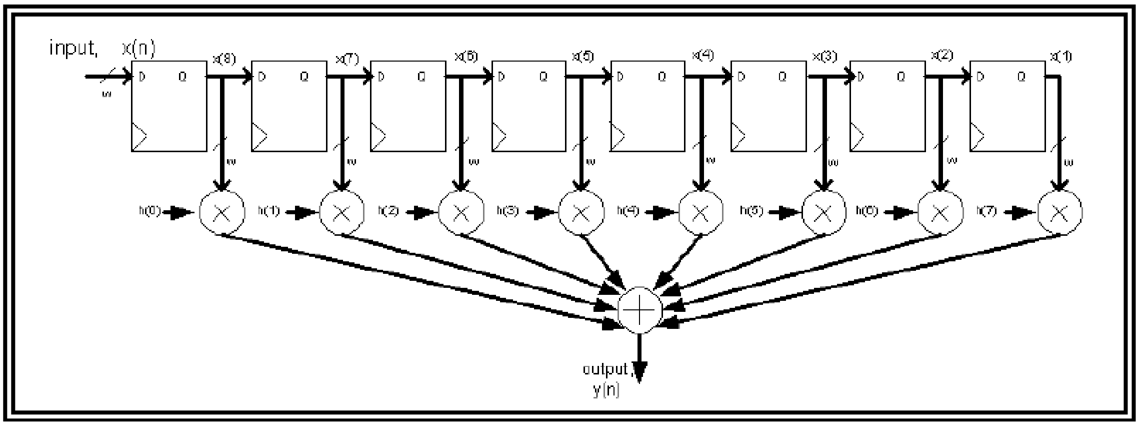
Ministry of Higher Education and Scientific Research
Middle Technical University
Electrical Engineering Technical College

Training package
in

Digital Signal Processing
(Image Processing I)

For

Students of third class
Control and Automation Engineering Techniques



By

Prof. Dr. Raaed F. Hassan
Dep. Control and Automation Eng. Tech.
September/2021

Image Processing I

1/ Overview

1 / A –Target population :-

For students of third class

Department of Medical Instrumentation Eng. Techniques

1 / B –Rationale :-

This unit introduces Image processing techniques

1 / C –Central Idea :-

In this unit, we describe principles of image processing.The major topics discussed in this unit are included in the following outline.

- **Introduction to Image Processing**
- **Image Processing Notation and Data Formats**
- **8-Bit Gray Level Images**
- **24-Bit Color Images**
- **8-Bit Color Images**
- **Intensity Images**
- **Red, Green, Blue Components and Grayscale Conversion**
- **Image Histogram and Equalization**
- **Grayscale Histogram and Equalization**

2/ Performance Objectives :-

After studying the 13^{th} modularunit, the student will be able to:-

1. Principles of image processing.

4/ the text :-

Introduction to Image Processing

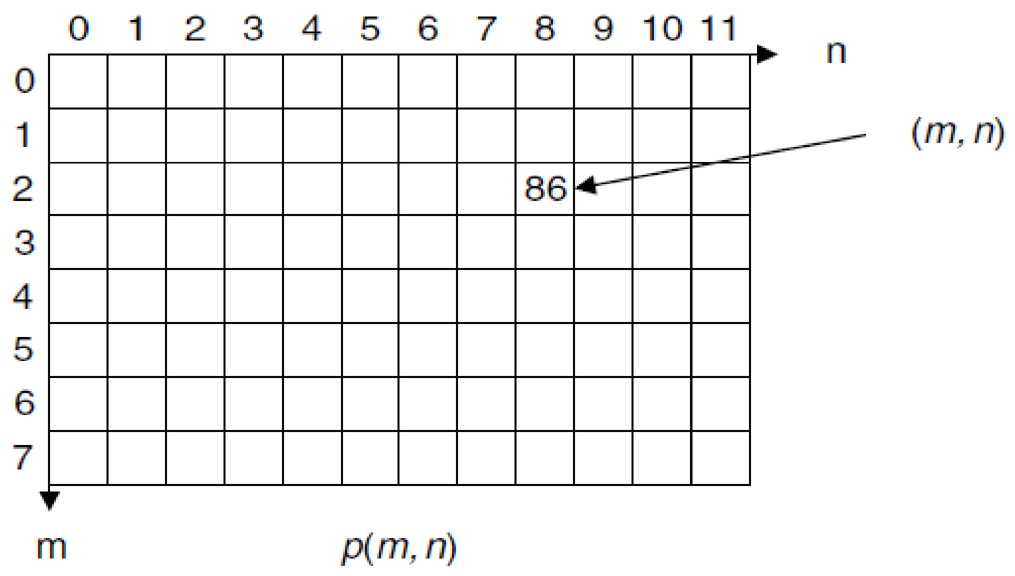
In today's modern computers, media information such as audio, images, and video have come to be necessary for daily business operations and entertainment. In this lecture, we will study the digital image and its processing techniques. This lecture introduces the basics of image processing, including image enhancement using histogram equalization and filtering methods, and proceeds to study pseudo-color generation for object detection and recognition. Finally, the lecture investigates image compression techniques and basics of video signals.

Image Processing Notation and Data Formats

The digital image is picture information in digital form. The image can be filtered to remove noise and obtain enhancement. It can also be transformed to extract features for pattern recognition. The image can be compressed for storage and retrieval, as well as transmitted via a computer network or a communication system.

The digital image consists of pixels. The position of each pixel is specified in terms of an index for the number of columns and another for the number of rows. Figure below shows that the pixel p(2, 8) has a level of 86 and is located in the second row, eighth column. We express it in notation as

$$p(2,\ 8) = 86.$$



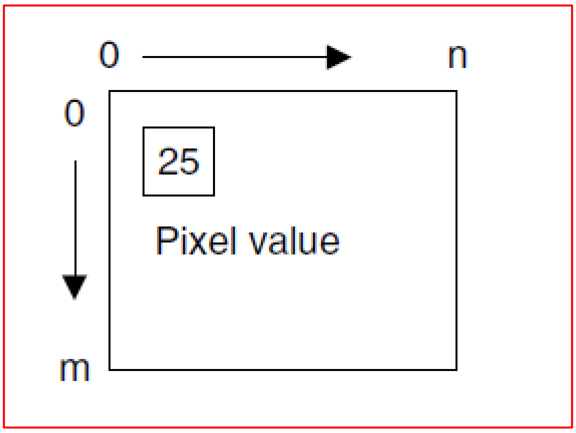
The number of pixels in the presentation of a digital image is its spacial resolution, which relates to the image quality. The higher the spacial resolution, the better quality the image has.

The spacial resolution can be fairly high, for instance, as high as 1600 x 1200 (1,920,000 pixels = 1.92 megapixels), or as low as 320 x 200 (64,000 pixels = 64 kilopixels).

In notation, the number to the left of the multiplication symbol represents the width, and that to the right of the symbol represents the height. Image quality also depends on the numbers of bits used in encoding each pixel level

8-Bit Gray Level Images

If a pixel is encoded on a gray scale from 0 to 255, where 0 = black and 255 = white, the numbers in between represent levels of gray forming a grayscale image. For a 640 x 480 8-bit image, 307.2 kilopixels (2.4 Mbyte) are required for storage. Figure below shows a grayscale image format. As shown in the figure, the pixel indicated in the box has an 8-bit value of 25.



The image of a cruise ship with spatial resolution of 320 x 240 in an 8-bit grayscale level is shown in Figure below.

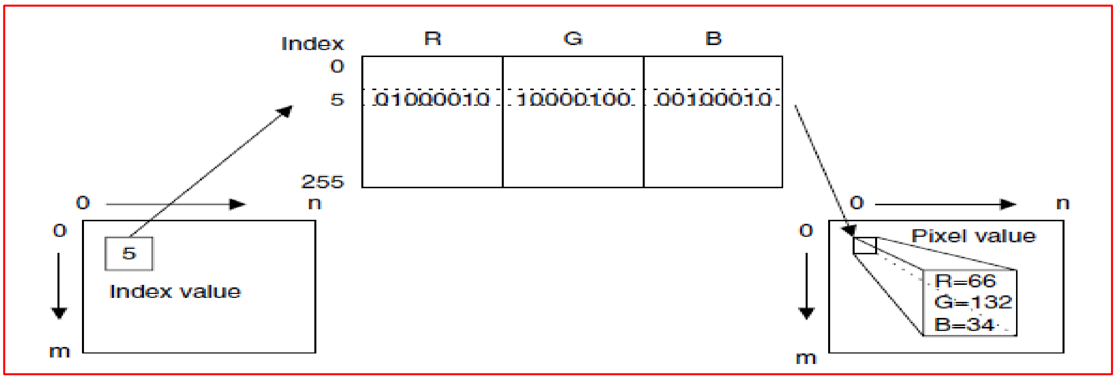
24-Bit Color Images

In a 24-bit color image representation, each pixel is recoded with red, green, and blue (RGB) components. With each component value encoded in 8 bits, resulting in 24 bits in total, we achieve a full color RGB image. With such an image, we can have $2^{24} = 16.777216 \times 10^{6}$ different colors. A 640 x 480 24-bit color image requires 7.2 Mbytes for storage.

- (0,0, 0) is black
- (255, 255, 255) is white
- (255, 0, 0) is red
- (0, 255, 0) is green
- (0, 0, 255) is blue
- (255, 255, 0) is yellow
- (0, 255, 255) is cyan
- (255, 0, 255) is magenta

8-Bit Color Images

The 8-bit color image is also a popular image format. Its pixel value is a color index that points to a color look-up table containing RGB components. We call this a color indexed image, and its format is shown in Figure below.



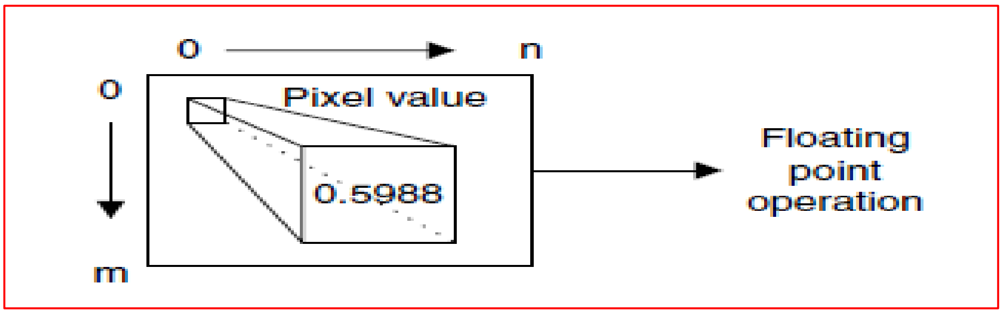
As an example in the figure, the color indexed image has a pixel index value of 5, which is the index for the entry of the color table, called the color map. At the index of location 5, there are three color components with RGB values of 66, 132, and 34, respectively.

Intensity Images

As we noted in the first section, the grayscale image uses a pixel value ranging from 0 to 255 to present luminance, or the light intensity. A pixel value of 0 designates black, and a value 255 encodes for white.

In some processing environments such as MATLAB (matrix laboratory), floating-point operations are used. The grayscale image has an intensity value that is normalized to be in the range from 0 to 1.0, where 0 represents black and 1 represents white.

We often change the pixel value to the normalized range to get the grayscale intensity image before processing it, then scale it back to the standard 8-bit range after processing for display. With the intensity image in the floating-point format, the digital filter implementation can be easily applied. Figure below shows the format of the grayscale intensity image, where the indicated pixel shows the intensity value of 0.5988.



Red, Green, Blue Components and Grayscale Conversion

In some applications, we need to convert a color image to a grayscale image so that storage space can be saved. As an example, the fingerprint image is stored in the grayscale format in the database system.

In color image compression, as another example, the transformation converts the RGB color space to YIQ color space, where Y is the luminance (Y) channel representing light intensity and the I (in-phase) and Q (quadrature) chrominance channels represent color details.

The luminance Y(m, n) carries grayscale information with most of the signal energy (as much as 93%), and the chrominance channels I(m, n) and Q(m, n) carry color information with much less energy (as little as 7%). The transformation in terms of the standard matrix notion is given by

$$\begin{bmatrix} Y(m,n) \\ I(m,n) \\ Q(m,n) \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R(m,n) \\ G(m,n) \\ B(m,n) \end{bmatrix}.$$

As an example of data compression, after transformation, we can encode *Y(m,n)* with a higher resolution using a larger number of bits, since it contains most of the signal energy, while we encode chrominance channels *I(m,n)* and *Q(m,n)* with less resolution using a smaller number of bits. Inverse transformation can be solved as

$$\begin{bmatrix} R(m,n) \\ G(m,n) \\ B(m,n) \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.621 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.106 & 1.703 \end{bmatrix} \begin{bmatrix} Y(m,n) \\ I(m,n) \\ Q(m,n) \end{bmatrix}.$$

To obtain the grayscale image, we simply convert each RGB pixel to a YIQ pixel and then keep its luminance channel and discard its IQ chrominance channels. Hence, the conversion formula is given by

$$Y(m,n) = 0.299 \cdot R(m,n) + 0.587 \cdot G(m,n) + 0.114 \cdot B(m,n).$$

Note that Y(m, n), I(m, n), and Q(m, n) can be matrices that represent the luminance image and two color component images, respectively. Similarly, R(m, n), G(m, n), and B(m, n) can be matrices for the RGB component images.

Example

Given a pixel in an RGB image as follows:

R = 200, G = 10, B = 100,

Convert the pixel values to the YIQ values.

Solution:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} 200 \\ 10 \\ 100 \end{bmatrix}.$$

Carrying out the matrix operations leads to

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 \times 200 & 0.587 \times 10 & 0.114 \times 100 \\ 0.596 \times 200 & -0.274 \times 10 & -0.322 \times 100 \\ 0.212 \times 200 & -0.523 \times 10 & 0.311 \times 100 \end{bmatrix} = \begin{bmatrix} 77.07 \\ 84.26 \\ 68.27 \end{bmatrix}.$$

Rounding the values to integers, we have

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = round \begin{bmatrix} 77.07 \\ 84.26 \\ 68.27 \end{bmatrix} = \begin{bmatrix} 77 \\ 84 \\ 68 \end{bmatrix}.$$

Example

Given a pixel of an image in the YIQ color format as follows:

Y = 77, I = 84, Q = 68,

Convert the pixel values back to the RGB values.

Solution:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.000 & 0.956 & 0.621 \\ 1.000 & -0.272 & -0.647 \\ 1.000 & -1.106 & 1.703 \end{bmatrix} \begin{bmatrix} 77 \\ 84 \\ 68 \end{bmatrix} = \begin{bmatrix} 199.53 \\ 10.16 \\ 99.90 \end{bmatrix}.$$

After rounding, it follows that

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = round \begin{bmatrix} 199.53 \\ 10.16 \\ 99.9 \end{bmatrix} = \begin{bmatrix} 200 \\ 10 \\ 100 \end{bmatrix}.$$

Example

Given the following 2 x 2 RGB image,

$$R = \begin{bmatrix} 100 & 50 \\ 200 & 150 \end{bmatrix} \quad G = \begin{bmatrix} 10 & 25 \\ 20 & 50 \end{bmatrix} \quad B = \begin{bmatrix} 10 & 5 \\ 20 & 15 \end{bmatrix},$$

Convert the RGB color image into a grayscale image.

Solution:

Since only Y components are kept in the grayscale image:

$$Y = 0.299 \times \begin{bmatrix} 100 & 50 \\ 200 & 150 \end{bmatrix} + 0.587 \times \begin{bmatrix} 10 & 25 \\ 20 & 50 \end{bmatrix} + 0.114 \times \begin{bmatrix} 10 & 5 \\ 20 & 15 \end{bmatrix} = \begin{bmatrix} 37 & 30 \\ 74 & 76 \end{bmatrix}.$$

Image Histogram and Equalization

An image histogram is a graph to show how many pixels are at each scale level, or at each index for the indexed color image. The histogram contains information needed for image equalization, where the image pixels are stretched to give a reasonable contrast.

Grayscale Histogram and Equalization

We can obtain a grayscale histogram by plotting pixel value distribution over the full grayscale range.

Example

Produce a histogram given the following image (a matrix filled with integers) with the grayscale value ranging from 0 to 7, that is, with each pixel encoded into 3 bits.

$$\begin{bmatrix} 0 & 1 & 2 & 2 & 6 \\ 2 & 1 & 1 & 2 & 1 \\ 1 & 3 & 4 & 3 & 3 \\ 0 & 2 & 5 & 1 & 1 \end{bmatrix}$$

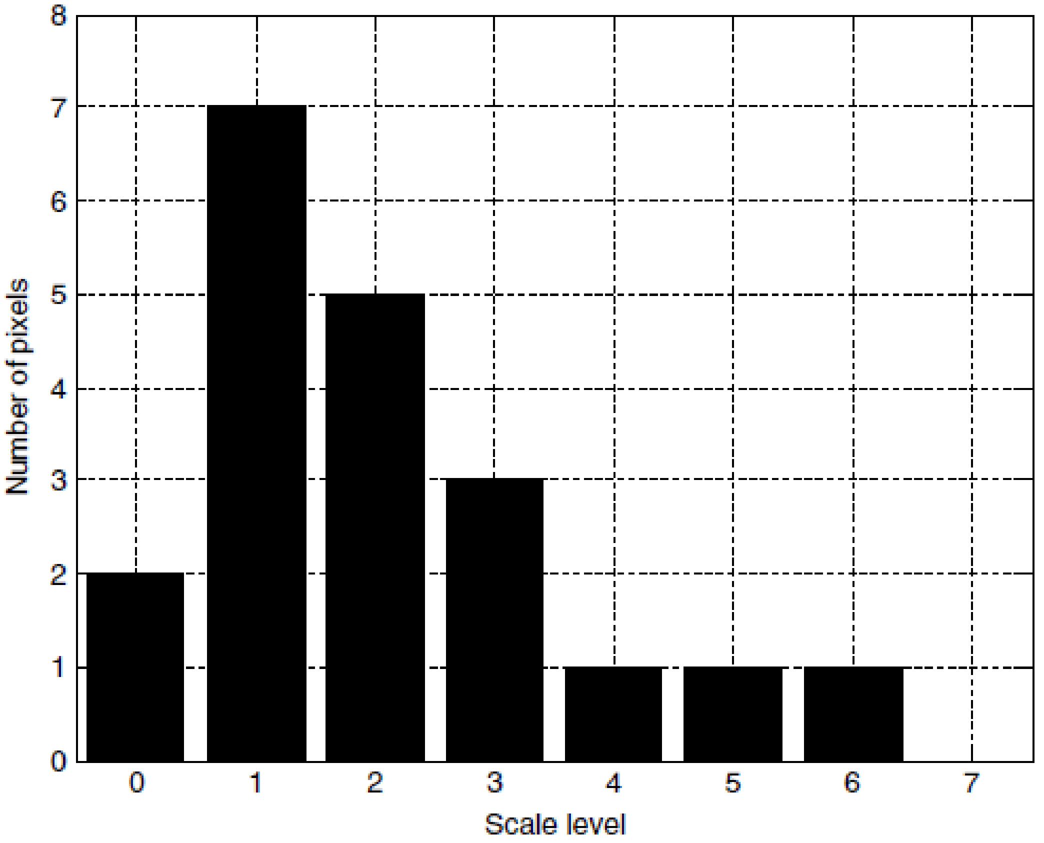
Solution:

Since the image is encoded using 3 bits for each pixel, we have the pixel value ranging from 0 to 7. The count for each grayscale is listed in Table below

Pixel counts distribution.

Pixel $p(m, n)$ Level	Number of Pixels
0	2
1	7
2	5
3	3
4	1
5	1
6	1
7	0

Based on the grayscale distribution counts, the histogram is created as shown in Figure below



With the histogram, the equalization technique can be developed. Equalization stretches the scale range of the pixel levels to the full range to give an improved contrast for the given image. To do so, the equalized new pixel value is redefined as

$$p_{eq}(m,n) = \frac{\text{Number of pixels with scale level} \leq p(m,n)}{\text{Total number of pixels}} \times (\text{maximum scale level})$$

The new pixel value is reassigned using the value obtained by multiplying the maximum scale level by the scaled ratio of the accumulative counts up to the current image pixel value over the total number of the pixels. Clearly, since the accumulate counts can range from 0 up to the total number of pixels, then the equalized pixel value can vary from 0 to the maximum scale level. It is due to the accumulation, the pixel values are spread over the whole range from 0 to the maximum scale level (255). Let us look at a simplified equalization example.

Example

Given the following image (matrix filled with integers) with a grayscale value ranging from 0 to 7, that is, with each pixel encoded in 3 bits,

$$\begin{bmatrix} 0 & 1 & 2 & 2 & 6 \\ 2 & 1 & 1 & 2 & 1 \\ 1 & 3 & 4 & 3 & 3 \\ 0 & 2 & 5 & 1 & 1 \end{bmatrix},$$

Perform equalization using the histogram in previous Example , and plot the histogram for the equalized image.

Solution:

Using the histogram result in Table

Pixel $p(m, n)$ Level	Number of Pixels
0	2
1	7
2	5
3	3
4	1
5	1
6	1
7	0

We can compute an accumulative count for each grayscale level as shown in Table

Pixel $p(m, n)$ Level	Number of Pixels	Number of Pixels $\leq p(m, n)$	Equalized Pixel Level
0	2	2	1
1	7	9	3
2	5	14	5
3	3	17	6
4	1	18	6
5	1	19	7
6	1	20	7
7	0	20	7

The equalized pixel level using is given in the last column.

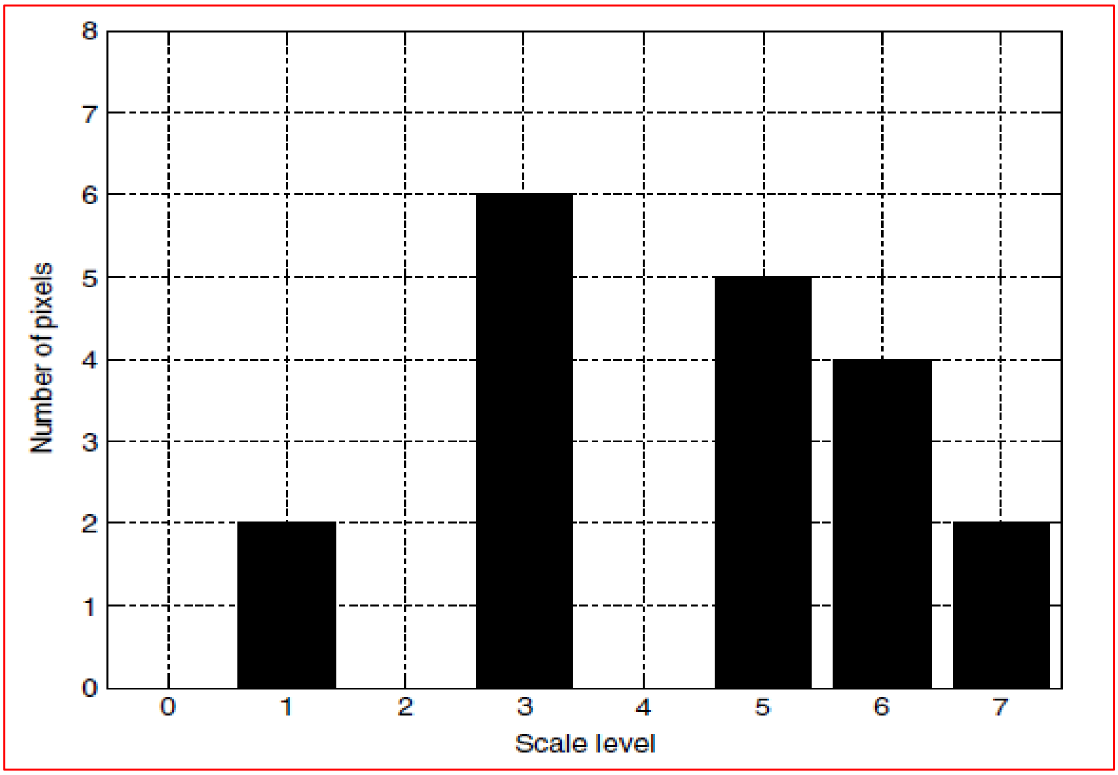
To see how the old pixel level p(m, n) = 4 is equalized to the new pixel level $p_{eq}(m, n)$ =6, we apply

$$p_{eq}(m,n) = round\left(\frac{18}{20} \times 7\right) = 6.$$

The equalized image using above Table is finally obtained by replacing each old pixel value in the old image with its corresponding equalized new pixel value and given by

$$\begin{bmatrix} 1 & 3 & 5 & 5 & 7 \\ 5 & 3 & 3 & 5 & 3 \\ 3 & 6 & 6 & 6 & 6 \\ 1 & 5 & 7 & 3 & 3 \end{bmatrix}.$$

Pixel $p(m, n)$ Level	Number of Pixels
0	0
1	2
2	0
3	6
4	0
5	5
6	4
7	2



Histogram for the equalized image

As we can see, the pixel levels in the equalized image are stretched to the larger scale levels. This technique woks for underexposed images.

7/References :-

1. Schaum's Outline of Theory and Problems ofDigital Signal processing.
2. Digital signal processing, principles, algorithms, and applications by John G. Proakis and Dimitris G. Manolakis.
3. Signal and systems, Alan Oppenheim.