

# Basic Intensity Transformations in Image Processing using MATLAB

Syed Munawwar<sup>1</sup> and K Sampath Kumar<sup>2</sup>

<sup>1,2</sup>Asst Professor, Department of ECE, Santhiram Engineering College, Nandyal, Andhra Pradesh, India

**Abstract:** Principle objective of Image enhancement is to process an image so that result is more suitable than original image for specific application. Digital image enhancement techniques provide a multitude of choices for improving the visual quality of images. The image enhancement techniques is done by gray-level transformation functions. The paper focuses on spatial domain techniques for image enhancement, with particular reference to point processing methods like image negative, log transformation and Contrast stretching operations implemented using MATLAB.

**Keywords:** Digital Image Processing, Image Enhancement and MATLAB.

## Introduction:

Image enhancement is basically improving the interpretability or perception of information in images for human viewers and providing 'better' input for other automated image processing techniques. The principal objective of image enhancement is to modify attributes of an image to make it more suitable for a given task and a specific observer. During this process, one or more attributes of the image are modified. The choice of attributes and the way they are modified are specific to a given task. Moreover, observer-specific factors, such as the human visual system and the observer's experience, will introduce a great deal of subjectivity into the choice of image enhancement methods. There exist many techniques that can enhance a digital image without spoiling it. The enhancement methods can broadly be divided into the following two categories: Spatial Domain Methods and Frequency Domain Methods.

In spatial domain techniques [1], we directly deal with the image pixels. The pixel values are manipulated to achieve desired enhancement. In frequency domain methods, the image is first transferred into frequency domain. It means that, the Fourier Transform of the image is computed first. All the enhancement operations are performed on the Fourier transform of the image and then the Inverse Fourier transform is performed to get the resultant image. These enhancement operations are performed in order to modify the image brightness, contrast or the distribution of the grey levels. As a consequence the pixel value (intensities) of the output image will be modified according to the transformation function applied on the input values.

Image enhancement simply means, transforming an image  $f$  into image  $g$  using  $T$ . (Where  $T$  is the transformation. The values of pixels in images  $f$  and  $g$  are denoted by  $r$  and  $s$ , respectively. As said, the pixel values  $r$  and  $s$  are related by the expression,

$$s = T(r) \quad (1)$$

Where  $T$  is a transformation that maps a pixel value  $r$  into a pixel value  $s$ . The results of this transformation are mapped into the grey scale range as we are dealing here only with grey scale digital images. So, the results are mapped back into the range  $[0, L-1]$ , where  $L=2^k$ ,  $k$  being the number of bits in the image being considered. So, for instance, for an 8-bit image the range of pixel values will be  $[0, 255]$ .

## 2. Histogram Processing

The histogram of a digital image with gray levels in the range  $[0, L-1]$  is a discrete function  $h(r_k) = (n_k)$ , where  $r_k$  is the  $k$ th gray level and  $n_k$  is the number of pixels in the image having gray level  $r_k$ . It is common practice to normalize a histogram by dividing each of its values by the total number of pixels in the image, denoted by  $n$ . Thus, a normalized histogram is given by

$$p(r_k) = n_k/n$$

for  $k=0,1,\dots,L-1$ . Loosely speaking,  $p(r_k)$  gives an estimate of the probability of occurrence of gray level  $r_k$ . Note that the sum of all components of a normalized histogram is equal to 1.

The histogram of a digital image with intensity levels in the range  $[0, L-1]$  is a discrete function

$$h(r_k) = n_k$$

$\downarrow$   $\downarrow$   
 $k^{\text{th}}$   $n_k$   
 intensity  $\downarrow$   $\downarrow$   
 value  $\downarrow$   $\downarrow$   
 Number of pixels in the  
 image with intensity  $r_k$

Histograms are frequently normalized by the total number of pixels in the image. Assuming an  $M \times N$  image, a normalized histogram.

## Histogram Equalization

Histogram equalization is a common technique for enhancing the appearance of images. Suppose we have an image which is predominantly dark. Then its histogram would be skewed towards the lower end of the grey scale and all the image detail is compressed into the dark end of the histogram.

### 3. Point Processing Operation

The simplest spatial domain operations occur when the neighborhood is simply the pixel itself. In this case  $T$  is referred to as a grey level transformation function or a point processing operation. Point processing operations take the form shown in equation (1).

#### 3.1 Create Negative of an Image

The most basic and simple operation in digital image processing is to compute the negative of an image. The pixel gray values are inverted to compute the negative of an image.

$$N(r, c) = 255 - I(r, c) \text{ where } 0 \leq r \leq R \text{ and } 0 \leq c \leq C \quad (2)$$

It can be seen that every pixel value from the original image is subtracted from the 255. The resultant image becomes negative of the original image. Negative images [3] are useful for enhancing white or grey detail embedded in dark regions of an image. In this paper we implemented negative of an image in MATLAB.

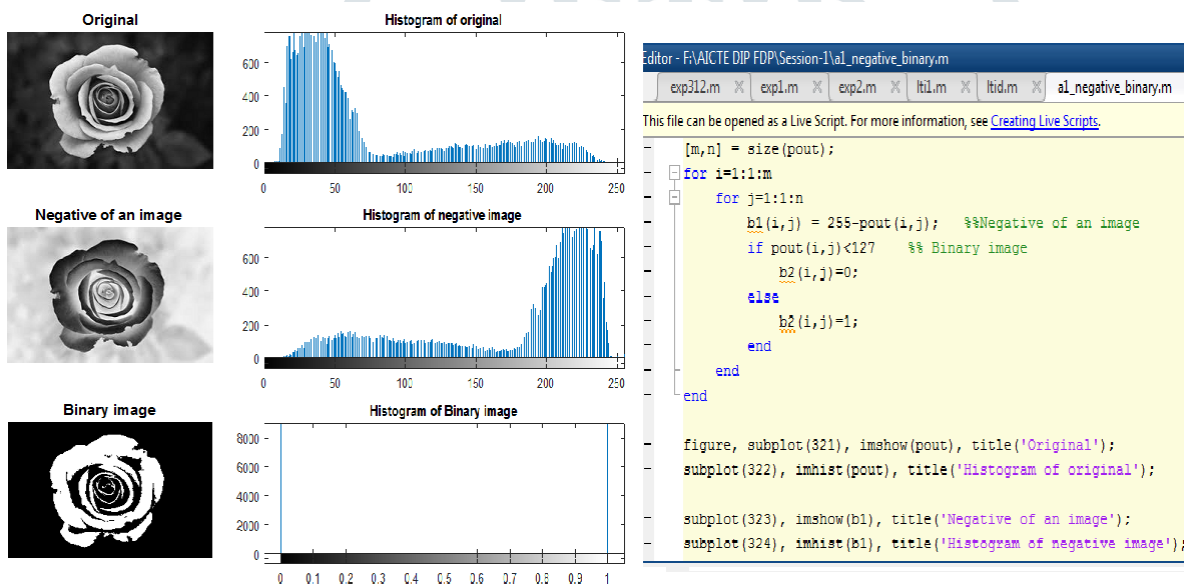


Fig1.shows Negative of an Image with Histogram and MATLAB code

#### 3.2 Logarithmic Transformations and Power-Law Transformations

The general form of the log transformation is

$$s = c * \log(1 + r) \quad (3)$$

The log transformation maps [3] a narrow range of low input grey level values into a wider range of output values. The inverse log transformation performs the opposite transformation. Log functions are particularly useful when the input grey level values may have an extremely large range of values.

Power-law transformations have the basic form

$$s = cr^g$$

where  $c$  and  $g$  are positive constants. Sometimes Eq. is written as  $s = c(r + \epsilon)^g$

As in the case of the log transformation, power-law curves with fractional values of  $g$  map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher values of input levels. In this paper we implemented Log and Power law Transformation of an image in MATLAB.

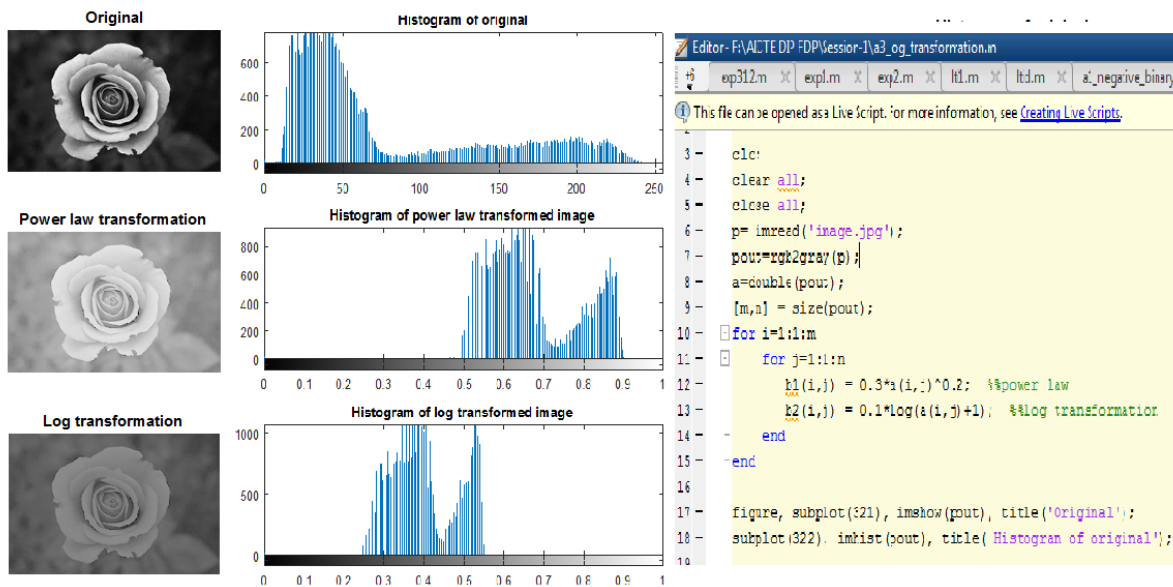


Fig2.shows Log and Power law Transformation of an Image with Histogram and MATLAB code

### 3.3 Contrast stretching:

One of the simplest piecewise linear functions is a contrast-stretching transformation. Low-contrast images can result from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture during image acquisition. The idea behind contrast stretching is to increase the dynamic range of the gray levels in the image being processed. In this paper we implemented contrast stretching of an image in MATLAB.

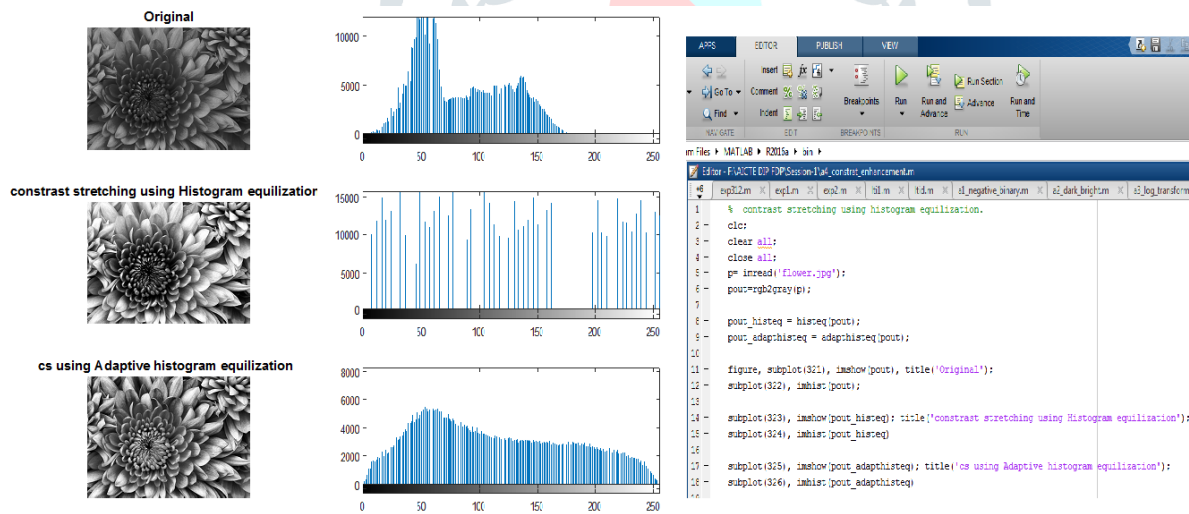


Fig3.shows Contrast stretching of an Image with Histogram and MATLAB code

### Conclusion

In this paper image enhancement in point processing methods like image negative, log transformation and Contrast stretching operations implemented. Image enhancement algorithms offer a wide variety of approaches for modifying images to achieve visually acceptable images. Image Negative is suited for enhancing white detail embedded in dark regions and has applications in medical imaging. Power-law transformations are useful for general- purpose contrast manipulation.. Log Transformation is Useful for enhancing details in the darker regions of the image at the expense of detail in the brighter regions the higher-level values. The histogram of an image provides important information regarding the contrast of an image. Histogram equalization is a transformation that stretches the contrast by redistributing the gray-level values uniformly. Here we discussed the image negative, log transformation, power-law transformations and Contrast stretching operations implemented with histogram using MATLAB.

### References

[1] R.C.Gonzalez & R.E.Woods, 'Digital Image Processing', Addison Wesley, second edition, ISBN:0-201-60078.

- [2] S.Jayaraman & S Esakkirajan T Veerakumar, 'Digital Image Processing', Tata McGraw Hill publication.
- [3] A. K. Jain, *Fundamentals of Digital Image Processing*.Englewood Cliffs, NJ: Prentice Hall, 1989.
- [4] Journal of computing,volume 2,issue 3, March 2010, ISSN 2151-9617.
- [5] <http://www.mathworks.in>.

