

## LAB-04: Transformation Operations and Histogram Equalization

### Objective:

The objective of this lab is to implement thresholding on images to convert them to binary, perform different transformation operations on images, find out the histogram of an image and perform histogram equalization.

### Theory:

**Thresholding** is the operation through which an image can be converted into a binary image/black & white i.e. having only two distinct levels. For thresholding, a threshold value is calculated for the entire image and then using this calculated threshold, the image can be converted to binary. This threshold value can be the mean or median etc. of the image. Similarly, a negative image can also be generated using the binary image and applying the transformation  $s = (L - 1) - r$  where  $L$  is the total number of gray levels while  $s$  is newly computed value and  $r$  is the current value of a pixel.

**Transformation operations** help enhance the quality of the image by applying operations like log, inverse log and power on the entire image. Different type of transformation yields different results.

**Histogram** of an image shows the frequency of different intensities values present in the image. This gives a clear idea of what intensities dominate the image. **Histogram equalization** is a technique that uses this information to enhance the contrast.

### Some Useful Commands:

1. To calculate the mean of 2D array using NumPy: `my_mean = numpy.mean (my_array)`
2. To calculate the log of an array using NumPy: `array_log = numpy.log (my_array)`
3. To calculate the log10 of an array using NumPy: `array_log10 = numpy.log10 (my_array)`
4. To calculate the power of an array using NumPy: `array_power = numpy.power (my_array, power)`

**Matplotlib – A Python Package for creating graphs:**

**Installing matplotlib:** `pip install matplotlib`

**Importing matplotlib:** `import matplotlib.pyplot as plt`

1. To plot a simple plot using matplotlib: `plt.plot( my_data)`
2. For label along x axis: `plt.xlabel ( 'Some cooked up data')`

3. For label along y axis: **plt.ylabel ( 'Some value' )**
4. To provide the range of the axis: **plt.axis ([Starting\_range\_of\_x, Ending\_range\_of\_x, Starting\_range\_of\_y, Ending\_range\_of\_y])**
5. To create a bar graph: **plt.bar (list\_equivalent\_to\_the\_size\_of\_the\_data\_array, the\_array\_containing\_the\_data, color="blue")**
6. To create a stem plot: **plt.stem(vector\_of\_which\_the\_stem\_plot\_is\_to\_be\_created)**
7. To show the graph: **plt.show()**

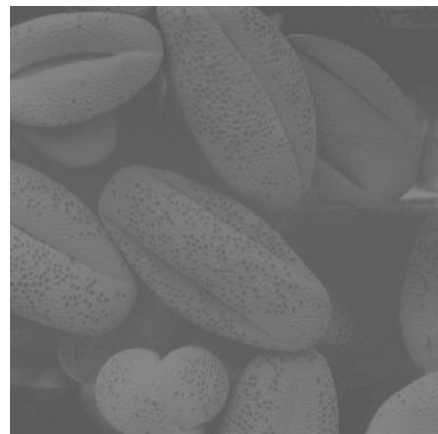
## Lab Tasks:

### Lab Task 1:

Read an image and convert it into black & white (binary image). Calculate threshold value by taking mean of whole image. Also, convert the same image into a negative image using the transformation:  $s = (L-1) - r$  where L are the total grayscale levels of the image.

### Lab Task 2:

Apply Power Law transformation for the following values of  $\gamma$  (0.9, 0.6, 1.3, and 1.8) for the images given below and display the results. Also, apply log transformation ( $\log_{10}$ ) on the images:



### Lab Task 3:

Perform the following steps to do Hist. equalization to enhance the grayscale image.

- Calculate the histogram of the image and display it using the appropriate command. (Don't use the built in function of OpenCV or Numpy etc.)
- Calculate probability density function (PDF) from the histogram and display it using the appropriate command  $PDF = H/(R*C)$ . Where H is the Histogram and R and C is the number of Rows and Columns of the image respectively.

- Calculate cumulative PDF and display it using the appropriate command.
- Multiply the Cumulative PDF with 255 to find the transformation function then display it too using the appropriate command.
- From the transformation function, replace the gray levels of the image to create contrast enhanced (histogram equalized) image.
- Display the enhanced image.

### **Conclusion:**

This lab gives an understanding of different transformation operations and how they affect an image. In addition, the lab also talks about thresholding and histogram equalization to improve the image quality.