**Experiment No. 5**

**Aim:** To implement Image negative, Gray level Slicing and Thresholding

**Objective:**

1. Convert an Image from RGB to Gray Level Image

2. Apply Image Negative

3. Apply Gray level Slicing( with and without preserving background)

4. Apply Thresholding

**Input Specifications:**

• Image of size MxN

**Theory:**

**Image Negative**

The negative of an image with grey levels in the range [0, L-1] is obtained by the negative transformation given below.

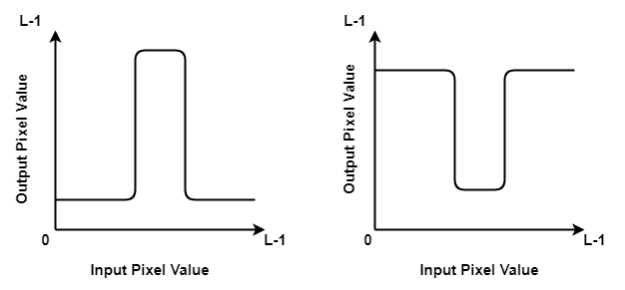
***s = L – 1 – r****.*

This expression results in reversing of the gray level intensities of the image thereby producing a negative like image. In negative transformation, each value of the input image is subtracted from the L-1 and mapped onto the output image. This is particularly useful for enhancing white or gray details embedded in dark regions of an image.

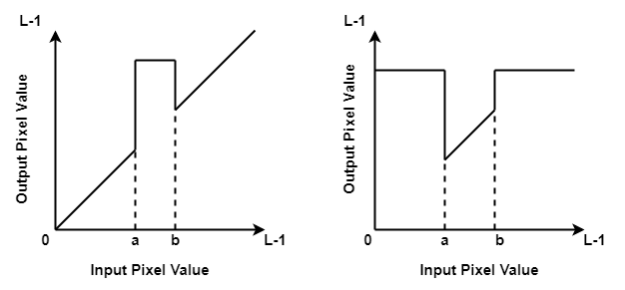
**Gray Level Slicing (Intensity Level Slicing)**

Intensity level slicing means highlighting a specific range of intensities in an image. In other words, we segment certain gray level regions from the rest of the image.

In the first type, we display the desired range of intensities in white and suppress all other intensities to black or vice versa. This results in a binary image. The transformation function for both the cases is shown below.



In the second type, we brighten or darken the desired range of intensities(a to b as shown below) and leave other intensities unchanged or vice versa. The transformation function for both the cases, first where the desired range is changed and second where it is unchanged, is shown below.



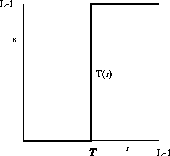
**Thresholding**

The simplest approach to segment an image is using thresholding.

A simple mapping function is defined by the thresholding operator:

Eqn:eqnptin2

The corresponding graph is shown below



**Problem Definition:**

1. Take a color image of size MxN
2. Convert Color image to Gray Scale Image
3. Find the highest range(L-1) of gray level in the range [0, L-1] from the image
4. Perform Image Negative.
5. Take Threshold values a and b as input.
6. Apply gray level slicing with and without preserving the background
7. Take Threshold value ‘T as input.
8. Apply thresholding on input image.
9. Conclude by specifying the applications where these operations can be used

**Code:**

| clc; positiveImage = imread('CameraMan.tif'); negativeImage = 255 - positiveImage; subplot(2,4,1), imshow(positiveImage) title('Original Image') subplot(2,4,5), imshow(negativeImage) title('Negetive Image') ThresholdImage = imread('CameraMan.tif'); [row , col] = size(ThresholdImage); t1 = 50; for i=1:row-1  for j=1:col-1  if ThresholdImage(i,j)<t1  ThresholdImage(i,j) = 0;  else  ThresholdImage(i,j) = 255;  end  end end subplot(2,4,2), imshow(ThresholdImage) title(['Threshold of ' , num2str(t1) , ' on Original']) ThresholdImage2 = imread('CameraMan.tif'); t2 = 180; for i=1:row-1  for j=1:col-1  if ThresholdImage2(i,j)<t2  ThresholdImage2(i,j) = 0;  else  ThresholdImage2(i,j) = 255;  end  end end subplot(2,4, 6), imshow(ThresholdImage2) title(['Threshold of ' , num2str(t2) , ' on Original']) % GREY SLICING g\_slice = positiveImage; [row, col] = size(positiveImage); a1 = 100; a2 = 255; for i=1:row-1  for j=1:col-1  if g\_slice(i,j)>a1 && g\_slice(i,j)<a2  g\_slice(i,j) = 255;  else  g\_slice(i,j) = 0;  end  end end subplot(2,4,3) imshow(g\_slice) title(['Gray Slicing of ', num2str(a1), '-', num2str(a2), ' without background']) neg\_gslice = negativeImage; for i=1:row-1  for j=1:col-1  if neg\_gslice(i,j)>a1 && neg\_gslice(i,j)<a2  neg\_gslice(i,j) = 255;  else  neg\_gslice(i,j) = 0;  end  end end subplot(2,4,7) imshow(neg\_gslice) title(['Neg Gray Slicing of ', num2str(a1), '-', num2str(a2), ' without bg']) % GRAY SCLICING WITH BACKGROUND g\_slice2 = positiveImage; a3=120; a4=150; for i=1:row-1  for j=1:col-1  if g\_slice2(i,j)>a3 && g\_slice2(i,j)<a4  g\_slice2(i,j) = 255;  else  g\_slice2(i,j) = positiveImage(i,j);  end  end end subplot(2,4,4) imshow(g\_slice2) title(['Gray Slicing of ', num2str(a3), '-', num2str(a4), ' with background']) neg\_g2 = negativeImage; for i=1:row-1  for j=1:col-1  if neg\_g2(i,j)>a3 && neg\_g2(i,j)<a4  neg\_g2(i,j) = 255;  else  neg\_g2(i,j) = negativeImage(i,j);  end  end end subplot(2,4,8) imshow(neg\_g2) title(['Neg Gray Slicing of ', num2str(a3), '-', num2str(a4), ' with bg']) |
| --- |

**Output**:



**Conclusion:**

In this experiment, we explored image processing in matlab. We converted images to negative versions, modified pixels based on a threshold, and performed gray slicing of images based on a user-defined threshold.

**Name:** Junaid Altaf Girkar

**SAP ID:** 60004190057

**Division:** BE Comps A2