LAB Manual

PART A

(PART A : TO BE REFFERED BY STUDENTS)

**Experiment No.03**

**A.1 Aim:**

**Implementation of Point Processing image enhancement Operations in Spatial Domain.**

**A.2 Prerequisite:**

1Matlab programming syntax (Refer the Matlab manual).

2.Knowledge of point processing operations.

3. Availability of Soft copy of your Photograph for experiment.

**A.3 Outcome:**

**After successful completion of this experiment students will be able to**

1. Implement following point processing image enhancement operations.

a. Negation of an image

b. Thresholding of an image

c. Contrast Stretching of an image

d. Gray level Slicing

e. Bit Plane slicing.

1. Understand the effect of point processing operations on an Image.
2. Identify the real-life application of point processing techniques implemented.

**A.4 Theory:**

1. ***Negation of an image***

The negative of an image with gray levels in the range [ 0, L-1] is obtained by using the negative transformation given by the expression

S= L – 1 – r …… Equation (1)

This is according to the transformation S = T ( r ) In above transformation ( 1 ) ,the intensity of the output image decreases as the intensity of the input increases. The type of processing is particularly suited for enhancing white or gray detail embedded in dark regions of an image especially when black areas are dominant in site.

****

**Figure 1: The concept of image negation**

1. ***Thresholding of an Image***

**Theory:**

Thresholding is a simple process to separate the interested object from the background. It gives the binary image. The formula for achieving thresholding is as follows

s = 0; if r <= t ……Equation (2)

s = L-1; if r > t



**Figure 2: The concept of image thresholding**

1. ***Contrast Stretching of an Image***

***Theory:***

Low contrast images can result from poor illumination, lack of dynamic range inthe imaging sensor etc. The idea behind contrast stretching is to increase thedynamic range of the gray levels in the image being processed. Thetransformation function for contrast stretching is given by

….Equation (3)

Where: f(x, y) is the Piecewise Linear Contrast Stretch in the image, a, b, and c are appropriate constants, which are the slopes in the respective regions and B is the maximum intensity value.



**Figure 3: The concept of Contrast stretching**

The location of the points (r1 , s1) & (r2 , s2) control the shape of the transformation function.

1. **Gray Level Slicing ( Intensity Level Slicing) :**

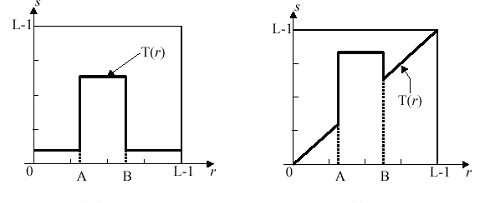
**Theory:**

Highlighting a specific range of gray-levels in an image is often desired.Applications include enhancing features such as masses of water, crop regions, or certain elevation area in satellite imagery.Another application is enhancing flaws in x-ray. There are two main different approaches:

        highlight a range of intensities while diminishing all others to a constant low level.

        highlight a range of intensities but preserve all others.

The fig. illustrates the intensity level slicing process. The left figures show a transformation function that highlights a range [A,B] while diminishing all the others. The right figures highlights a range [A,B] but preserves all the others.



**Figure 4: The concept of Gray level slicing with and without background**

The formulation for grey level slicing without background is

S = L-1 ; if A ≤ r ≤ B … Equation (4)

S = 0; Otherwise

The formulation for grey level slicing with background is

S = L-1 ; if A ≤ r ≤ B ….Equation (5)

S = r; Otherwise

1. **Bit Plane Slicing:**

**Theory:**

Bit plane slicing is new way of looking at an image. In bit plane slicing the image is considered to be a stack of binary images. The images closes to the bottom are least significant and the images on top are most significant.Instead of highlighting intensity ranges, highlighting the contribution made to the total image appearance by specific bit might be desired.Imagine that the image is composed of eight 1-bit planes, ranging from plane 0 for least significant bit to plane 7 for the most significant bit.

|  |
| --- |
| http://bme.med.upatras.gr/improc/bit_plane_slicing1.jpg |
| **Figure 5. A model of the  bit-planes** |

**A.5 Procedure/Algorithm:**

**A.5.1 TASK 1:**

**Negation of an image**

1. Read i/p image

2. Read maximum gray level pixel of i/p image

3. Replace input image by ( maximum – i/p ) = o/p

4. Display o/p image

5. Observe the output and complete PART B of lab manual.

6. Save and close the file and name it as **EXP3\_Task1\_your Roll no.m**

**A.5.2 TASK 2:**

***Thresholding of an Image***

1. Read input image

2. Enter thresholding value t

3. If image pixel is less than t replace it by zero.

4. If image pixel is > t replace it by 255

5. Display input image

6. Display threshold image

7. Display input image

8. Display threshold image

9. Observe the output and complete PART B of lab manual.

10. Save and close the file and name it as **EXP3\_Task2\_your Roll no.m**

**A.5.3 TASK 3:**

***Contrast Stretching of an Image***

1. Read input image

2. Enter values r1,r2,s1,s2

3. Calculate alpha,beta and gamma slopes.

4. if input pixel value is <= r1 then o/p = alpha x input

5. If input pixel is > r1and <=r2 then o/p = beta x (r-r1)+s1

6. otherwise o/p = gamma x (r-r2)+s2

7. Display i/p image

8. Display o/p image.

9. Observe the output and complete PART B of lab manual.

10. Save and close the file and name it as **EXP3\_Task3\_your Roll no.m**

**A.5.4 TASK 4:**

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

1. Read input image

2. Enter values A, B.

3. Use the appropriate formula for without background and with background.

4. Find the output image for without background and with background.

5. Display i/p image

6. Display o/p images.

7. Observe the output and complete PART B of lab manual.

8. Save and close the file and name it as **EXP3\_Task4\_your Roll no.m**

**A.5.5 TASK 5:**

**Bit Plane Slicing**

1. Read input image.

2. For bit value from 1 to 8, find corresponding image planes.

3. Display i/p image

4. Display all 8 o/p image planes separately

5. Observe the output and complete PART B of lab manual.

6. Save and close the file and name it as **EXP3\_Task5\_your Roll no.m**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

PART B

(PART B : TO BE COMPLETED BY STUDENTS)

***(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Black board access available)***

|  |  |
| --- | --- |
| Roll No. N049 | Name: Tarun Tanmay |
| Class: MBA Tech CE | Batch : B3 |
| Date of Experiment: | Date of Submission |
| Grade : |  |

**B.1 Software Code written by student:**

***(Paste your Matlab code completed during the 2 hours of practical in the lab here)***

clear all;

clc;

img1=imread('/Users/tjrox0825/Desktop/Levis.jpg');

%img1=imread('/Users/tjrox0825/Desktop/Levis.jpg');

re\_img1=imresize(rgb2gray(img1),[300,300]);

figure();

imshow(re\_img1);

title('Orignal')

img\_neg = re\_img1;

%img\_neg = double(img\_neg);

img2 = img\_neg(:);

L = max(img2);

for i=1:300

for j=1:300

img\_neg(i,j)= L-img\_neg(i,j);

end

end

%img\_neg = unit8(img\_neg)

figure();

imshow(img\_neg);

title("Negative")

img\_thres = re\_img1;

t = 100;

for i=1:300

for j=1:300

if (img\_thres(i,j)<=t)

img\_thres(i,j)=0;

else

img\_thres(i,j)=255;

end

end

end

figure();

imshow(img\_thres);

title("Thresholding")

img\_cont = re\_img1;

r1 = input("Enter the value of r1: ");

r2 = input("Enter the value of r2: ");

s1 = input("Enter the value of s1: ");

s2 = input("Enter the value of s2: ");

alpha=(s1)/(r1);

beta=(s2-s1)/(r2-r1);

gamma=(L-s2)/(L-s1);

for i=1:300

for j=1:300

pix=img\_cont(i,j);

if (pix <= r1)

newpix=alpha\*pix;

img\_cont(i,j)=newpix;

elseif((pix<r1)&& pix<=r2)

newpix=beta\*(pix-r1)+s1;

img\_cont(i,j)=newpix;

else

newpix = gamma\*(r1-r2)+s2;

img\_cont(i,j)=newpix;

end

end

end

figure();

imshow(img\_cont);

title("Image contrasting")

img\_bac = re\_img1;

wbimg1 = re\_img1;

for row=1:300

for col=1:300

if(wbimg1(row,col)>= r1 && wbimg1(row,col)<= r2)

wbimg1(row,col)= L-1;

else

wbimg1(row,col) = 0;

end

end

end

for i=1:300

for j=1:300

if (img\_bac(i,j)>=r1 && img\_bac(i,j)<=r2)

img\_bac(i,j)=L-1;

end

end

end

figure();

imshow(wbimg1);

title("Grey slicing(No background)")

figure();

imshow(img\_bac);

title("Grey slicing(With background)")

img\_copy = re\_img1;

figure();

C=zeros(300,300,8);

for k=1:8

for i=1:size(re\_img1,1)

for j=1:size(re\_img1,2)

C(i,j,k)=bitget(img\_copy(i,j),k) ; %Bit slicing

end

end

subplot(3,3,k+1), imshow(C(:,:,k)), title(['Bit Plane ',num2str(k-1)]);

end

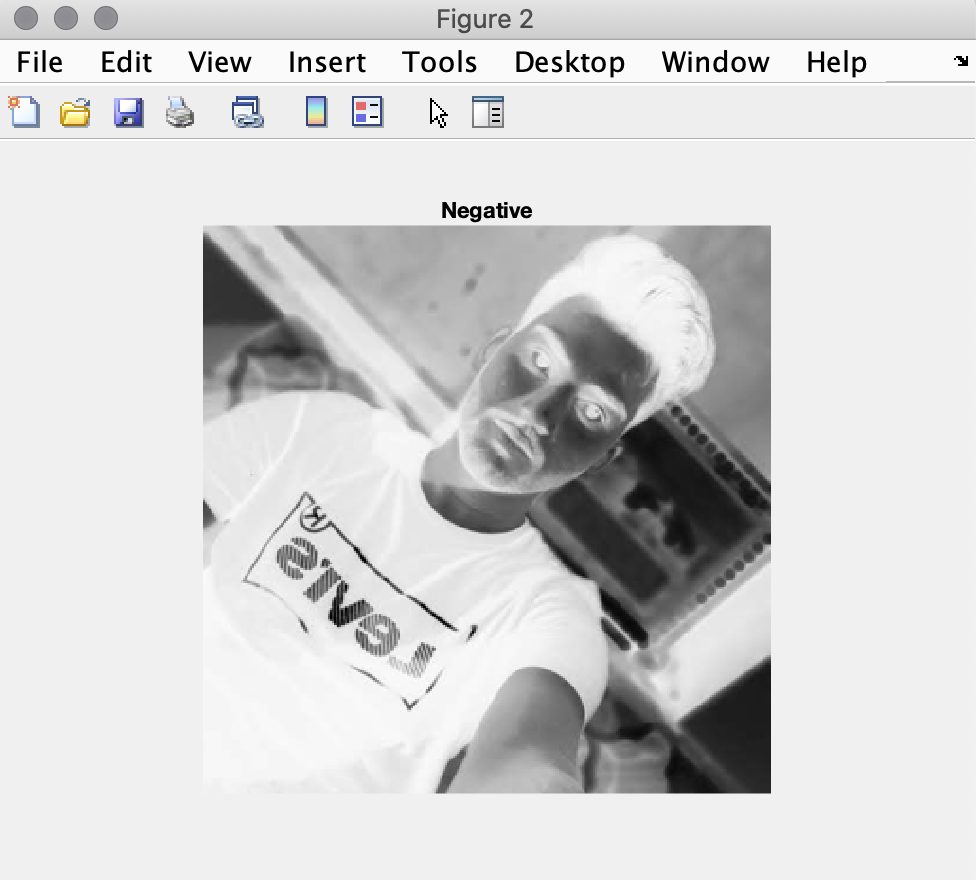
**B.2 Input and Output:**

***(Paste your program input and output in following format, If there is error then paste the specific error in the output part. In case of error with due permission of the faculty extension can be given to submit the error free code with output in due course of time. Students will be graded accordingly.)***

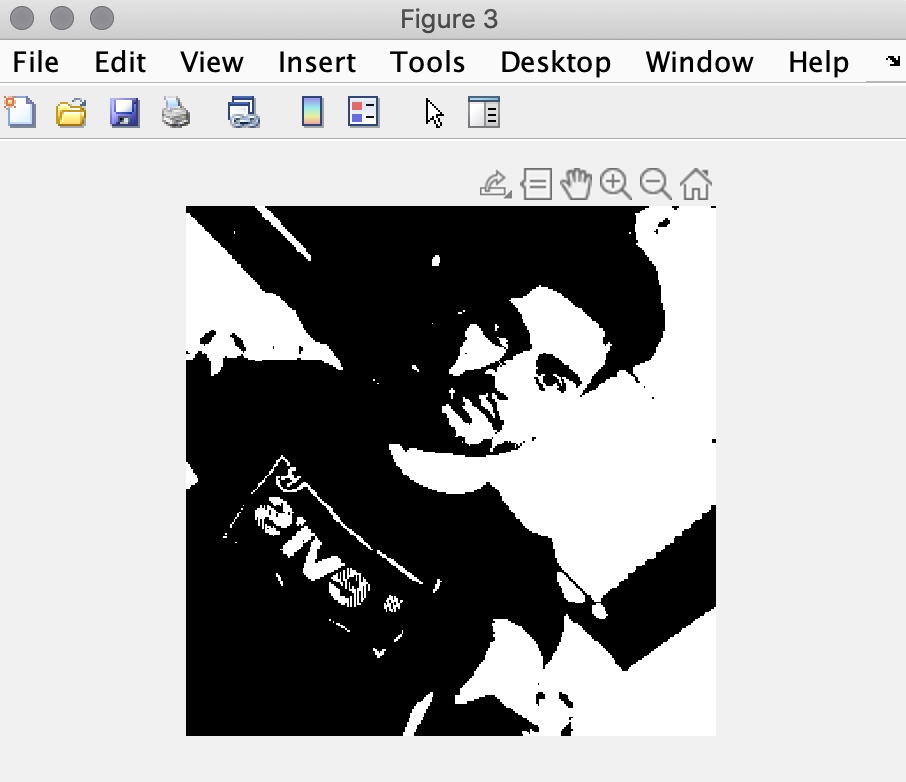
**Input Images: A screenshot of a cell phone

Description automatically generated**

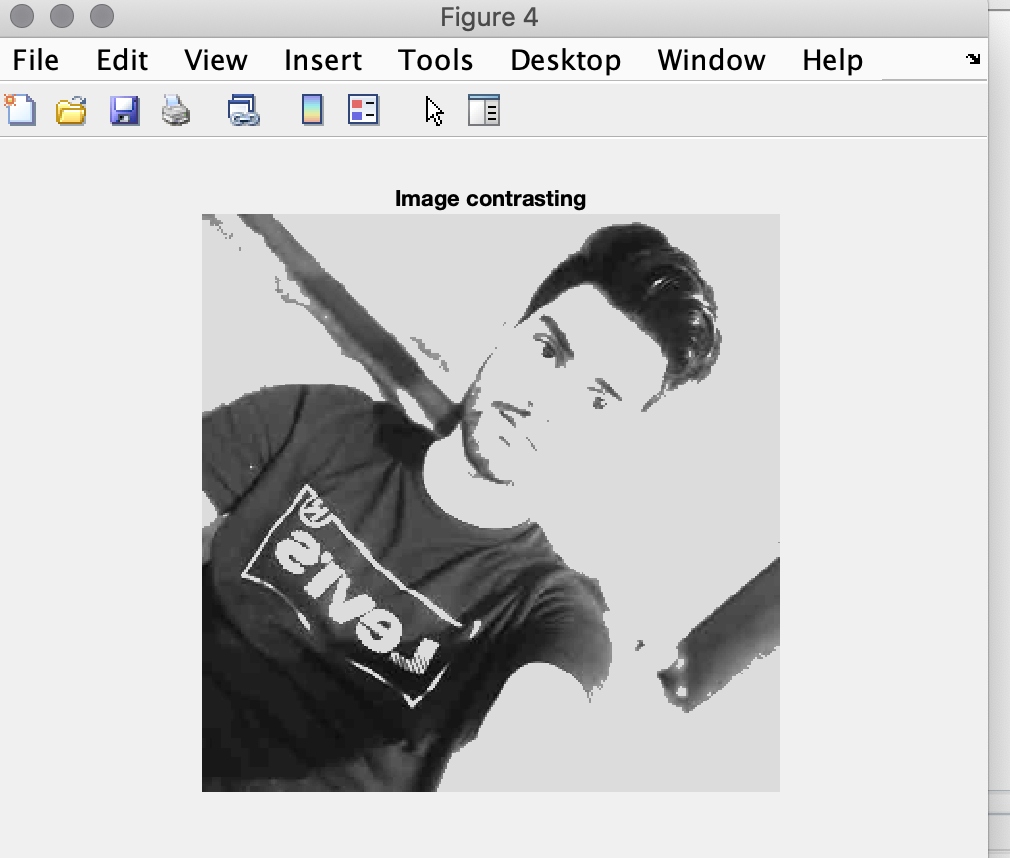
**Output:**

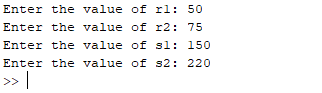
**Negative:**

**Thresholding of images:**

****

**Contrasting Image:**

****

****

**Grey image slicing:**

**1)Without background:**

**A screenshot of a cell phone

Description automatically generated**

**2)With background**

**A picture containing photo, holding, player, person

Description automatically generated**

**Bit Plane Slicing**

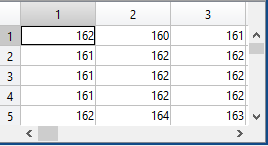
**A screenshot of a cell phone

Description automatically generated**

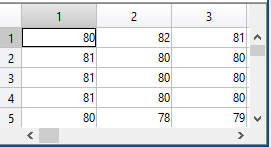
1. **Output Image for each point processing operation performed on input image.**
2. **Comments based on matrix content of output image (for each operation) as seen in Workspace.**

**Matrices:**

**Original:**

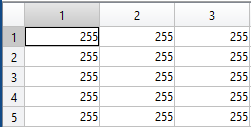
****

**Negative:**

****

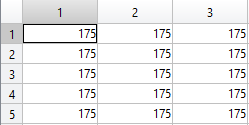
**Pixel values replaced with max of pixel value – original pixel value**

**Thresholding:**

****

**Pixel value greater than a 100 then replace by 255 else replace with 0**

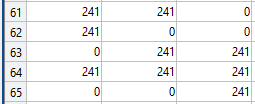
**Contrast Stretching:**

****

**Grey Level slicing**

**Without background: If pixel value lies in interval then new value**

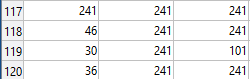
**boosted to 241(max pixel value).**

****

**Else 0**

**With background: If pixel value lies in interval then new value**

**boosted to 241(max pixel value). Else remains same.**

****

**B.3 Observations and learning:**

***(Students are expected to comment on the output obtained with clear observations and learning for each task/ sub part assigned)***

***Various image enhancement techniques were learned and implemented.***

**B.4 Conclusion:**

*(****Students must write the conclusion as per the attainment of individual outcome listed above and learning/observation noted in section B.3)***

***Various image enhancement techniques were learned and implemented.***

**B.5 Question of Curiosity**

***Solve following: a \* f1 + (1- a) \* f2 Where a = 0.5, f1 = your image and f2 = your neighbor classmate’s image***

clear all;

clc;

img1=imread('/Users/tjrox0825/Desktop/Trip.jpg');

re\_img1=imresize(rgb2gray(img1),[300,300]);

figure();

imshow(re\_img1);

img2=imread('/Users/tjrox0825/Desktop/College.jpg');

re\_img2 = imresize(rgb2gray(img2),[300,300]);

figure();

imshow(re\_img2);

new\_img=re\_img1;

new\_img=(0.5\*re\_img1+(1-0.5)\*re\_img2);

figure();

imshow(new\_img);

***A screenshot of a social media post

Description automatically generated***

***A screenshot of a person

Description automatically generated***

***A screenshot of a cell phone

Description automatically generated***

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*