POKHARA UNIVERSITY

DEPARTMENT OF COMPUTER SCIENCE

APEX COLLEGE

Mid-Baneshwor, kathmandu

CGIP

Lab-Report:01

SUBMITTED BY:

Diamond Nath

Roll No:14

BCIS-MAXTHON

SUBMITTED TO:

Bijaya Khadka

Abinash Khanal

Signature: ........................

**INTRODUCTION TO IMAGE PROCESSING TOOLS**

LAB OBJECTIVES

* To  understand image/image processing
* To read, show and write image in octave
* To be familiar with octave

IMAGE

            An image is an array or a matrix of square pixels (picture elements) arranged in columns and rows.  An image may be defined as a two –dimensional function, f(x, y) ,  where x and y are spatial coordinate ,and the amplitude of   ‘f’ at any pair of co-ordinates (x, y)  is called the intensity of gray  level of image at that point. When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a digital image.

Digital image is an image composed of picture elements also known as pixels.

IMAGE PROCESSING

           It is the method to perform some operations on image, in order to get an enhanced image or to extract some useful information from it. Digital processing is the use of digital computers to process digital image through an algorithm.

 STEPS OF IT:

1. Image acquisition,

2. Image analysis/manipulation,

3. Output.

OCTAVE

         GUI Octave is a Graphical User Interface for the GNU OCTAVE high-level language. GNU OCTAVE provides a command interface for solving linear and nonlinear problems numerically. It is open source and less resource intensive but has limited functionality.

COMMANDS

* Reading and image- imread()

Syntax:

            variable\_name= imread (‘filename’, ‘format’);

* Displaying an image- imshow()

Syntax:

             imshow(variable\_name)

* Writing and image

Syntax:

             imwrite(variable\_name, ‘filename’, ‘format’);

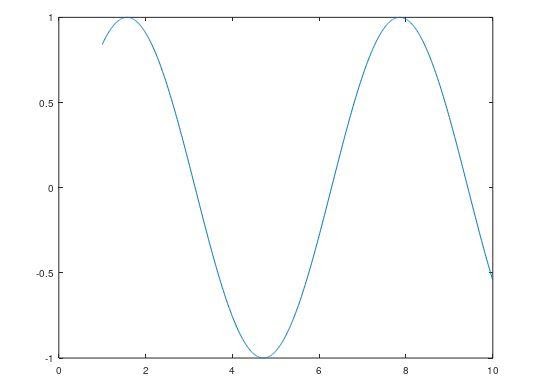
CODES

1. x=1:0.01:10;

              y=sin(x);

              plot (x,y)

OUTPUT

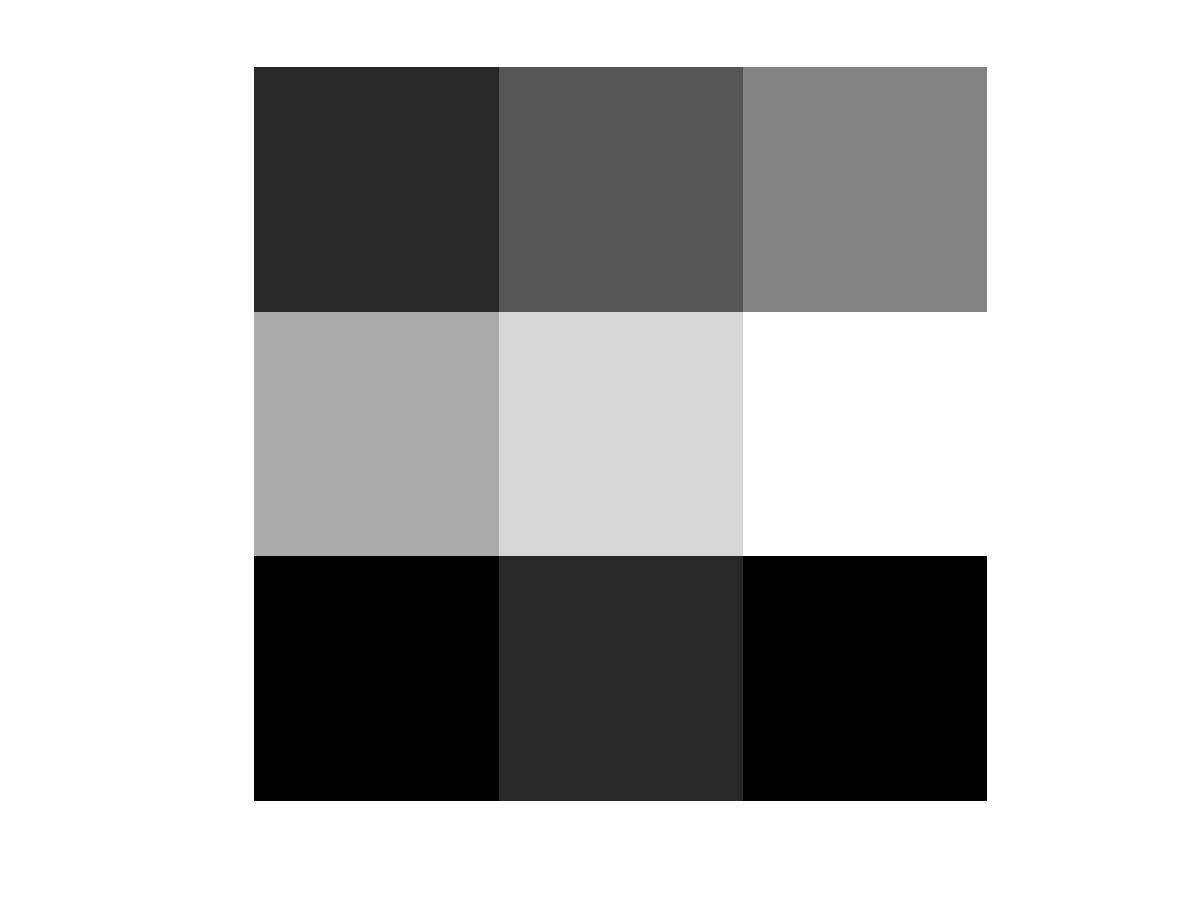


 Here, x=1:0.01:10; performs start, step, stop operation   y=sin(x); refers y is the sinusoidal function of x and   plot(x, y) plots x and y in the graph.

2. img=[1 2 3 ;4 5 6;0 1 0]

   Imshow(img,[])

OUTPUT

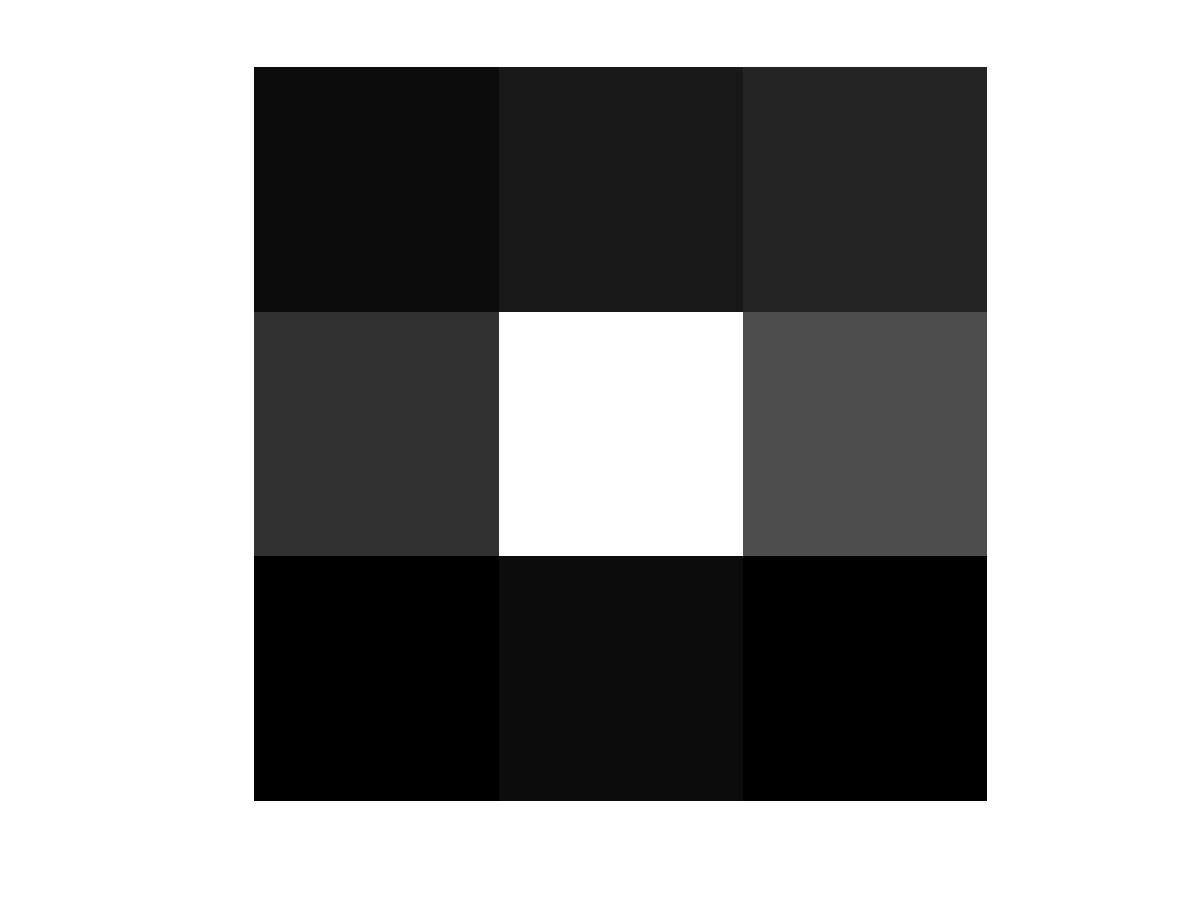


This displays an image of 3\*3 matrix and imshow is used to show the  image .

 -img(2,2)=20

 imshow(img,[])

OUTPUT

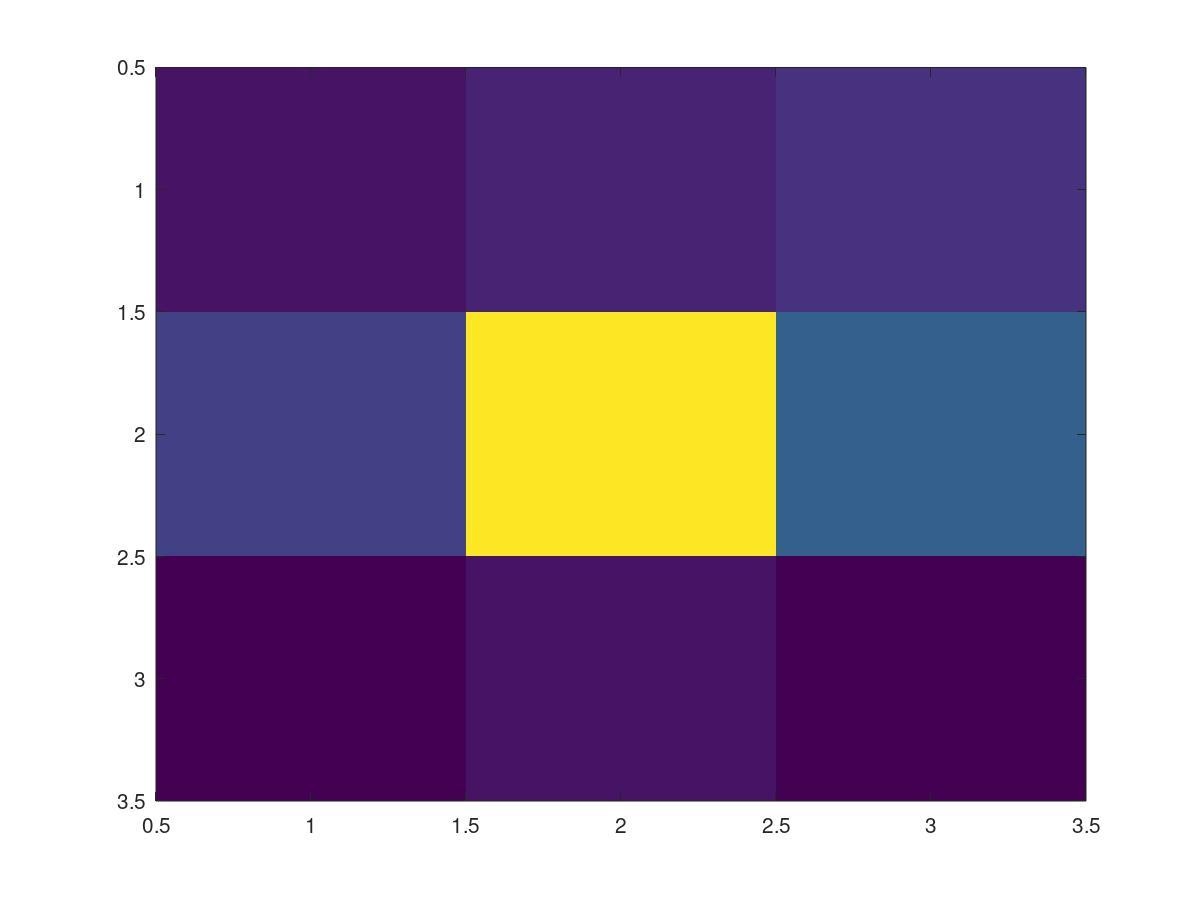


The above command img(2,2)=20 accesses the central element of the 3\*3  matrix and imshow shows the image.

imagesc(img)

 colorbar

OUTPUT



imagesc(img) displays the image in the colored graph format with scales  and colorbar is used to display the colorbar for the graphical colored  image.

3. img1=ones(5)

 img2=zeros(5)

 img2(: ,2)=1

 img2(: , [3,4]) =2

 img1([2,3], :)=5

 img3=img1([2,1], [3,2])

 img3=img1([2:4], [3:5])

In the above commands, img1=ones(5) displays a square matrix of all 1s  of size 5\*5 whereas , img2=zeros(5) displays a square matrix of all os of  5\*5 size.

Similarly, img2(: ,2)=1 is used to display value 1 in second column of all  the rows.

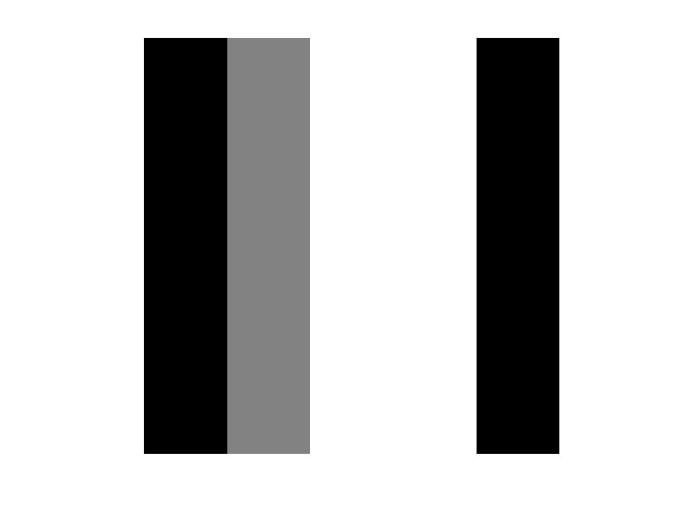
The command img2(: , [3,4]) =2 displays value 2 in the third and fourth  columns together in the same matrix.

Similarly, img1([2,3], :)=5 displays 5 in the second and third rows  together in the same matrix.

              The command img3=img1([2:4], [3:5]) is used to make a new         matrix  with the 2nd row to 4th row and 3rd column to 5th column.

4. x=2:2:14  imshow(img2, [])

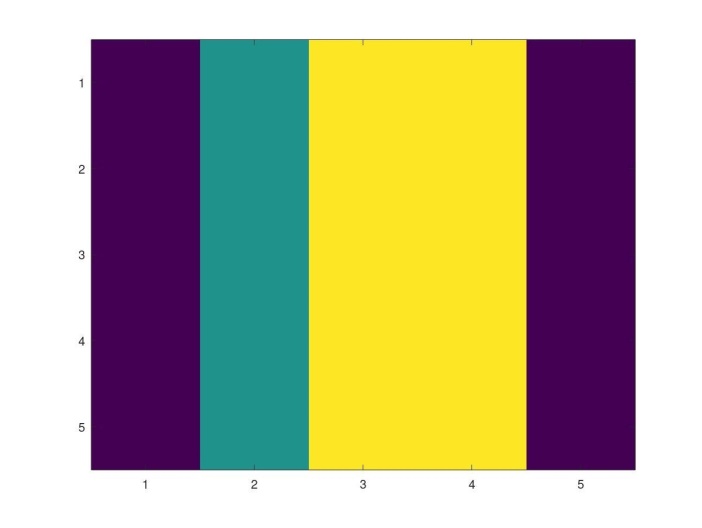
OUTPUT



imagesc(img2)

s colorbar

OUTPUT



5.  img3= zeros(4,6)

 This command creates a matrix of 0s with 4 rows and 6 columns.

 img3(2,:)=1;

 img3(3,:)=2;

 img3(4,:)=3;

 imshow(img3, [])

OUTPUT



6.it provides detailed information on currently defined variables.

7.  test\_img=imread(‘lenna.png’);

     imshow(test\_img);

     test\_gray=rgb2gray(test\_img);

     imshow(test\_gray);

OUTPUT



Discussion:

From this lab, we came to know that Image quantization level can be  accessed using the command images. imshow() command is used to show  the image, imagesec () command scales amd colours the image of matrix  also. imread() is used to read the image whereas imwrite() is used to save  the modified image. Similarly, whos command diaplays elements, class  and size.

Conclusion:

Using Octave for image processing was the main learning of this lab class.  Reading and writing image, displaying the image and changing the image  in grayscale was also learnt during this lab class. Similarly, we also learnt  how to write different commands and what is their role in processing of  images.

ASSIGNMENT

1. Read an image, and replace the pixel value of coordinate x = 200 +  your\_roll\_no , y= 200 - your\_roll\_no with your roll\_no. (Note: the  size of image should be greater than equal to 300 X 300). Write into  an image file and submit.

test\_img\_one=imread(‘lenna.png’);

test\_img\_one(223,177)=14;

imshow(test\_img\_one);

imwrite(test\_img\_one, ‘lab1.jpeg’);

OUTPUT



2. Crop the face of 'lenna.png'. You should replace one pixel value of  the resulting image with your roll number and specify the position of  replaced pixel in the report. Submit the final image.

(Cropping the image)

test\_img\_one=imread(‘lenna.png’);

pkg load image

a= imcrop(test\_img\_one, [193 185 178 218]);

imshow(a, []);

(Replacing pixel value with roll number)

test\_img\_one(50,100)=14;

imshow(test\_img\_one,[]);

OUTPUT

