**INDEX**

|  |  |  |
| --- | --- | --- |
| Serial No. | Contents | Page No. |
| **1** | Assignment 1 |  |
| **2** | Assignment 2 |  |
| **3** | Assignment 3 |  |
| **4** | Assignment 4 |  |
| **5** | Assignment 5 |  |
| **6** | Assignment 6 |  |
| **7** | Assignment 7 |  |
| **8** | Assignment 8 |  |
| **9** | Assignment 9 |  |
| **10** | Assignment 10 |  |

**ASSIGNMENT 1**

1. Write a program to display two pictures in a single frame in a single window.

**Background:** The following code displays 2 images in a single frame in a single window.TheRGB image is converted into grayscale image. The grayscale images are displayed alongside.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(filename)

* subplot():It divides the current figure into m by n grid and creates axes in the specified position.

Syntax: imread(m,n,p)

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

subplot(1,2,1);

imshow(I);

subplot(1,2,2);

imshow(I);

**Output:** fig 1

2. Write a program to display two pictures in a single frame in a single window with corresponding titles.

**Background:**The following code displays 2 images in a single frame in a single window.The RGB image is converted into grayscale image. The grayscale images are displayed alongside. Titles of each image are displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(filename)

* subplot():It divides the current figure into m by n grid and creates axes in the specified position.

Syntax: imread(m,n,p)

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

subplot(1,2,1);

imshow(I);

title(‘Image1’);

subplot(1,2,2);

imshow(I);

title(‘Image 2’);

**Output:** fig 2

3. Write a program to display negative of an image.

**Background:**The RGB image is converted into grayscale image. The following code displays the negative of the image.

**Functions used:**

* imread():It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

J=255-I;

imshow(J);

**Output**: fig 3

4. Write a program to increase saturation of an image.

**Background:** RGB image is converted into grayscale image. The following code increases the saturation of the image and displays.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* rgb2hsv(): It converts RGB image to HSV.
* hsv2rgb(): It converts HSV to RGB.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

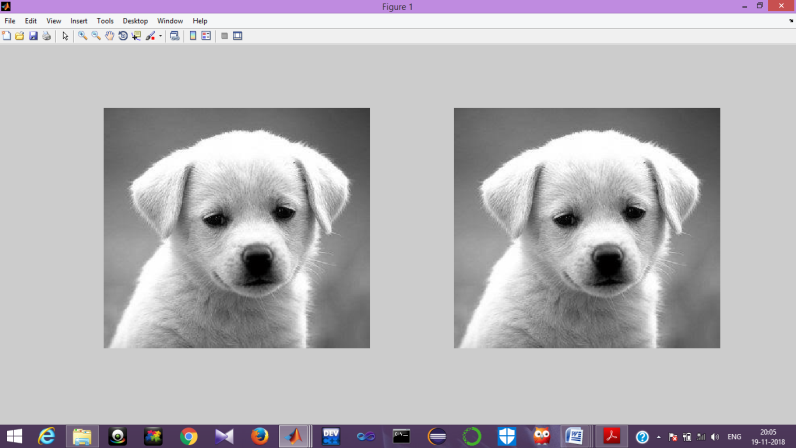
I=rgb2hsv(I);

I=I\*1.2;

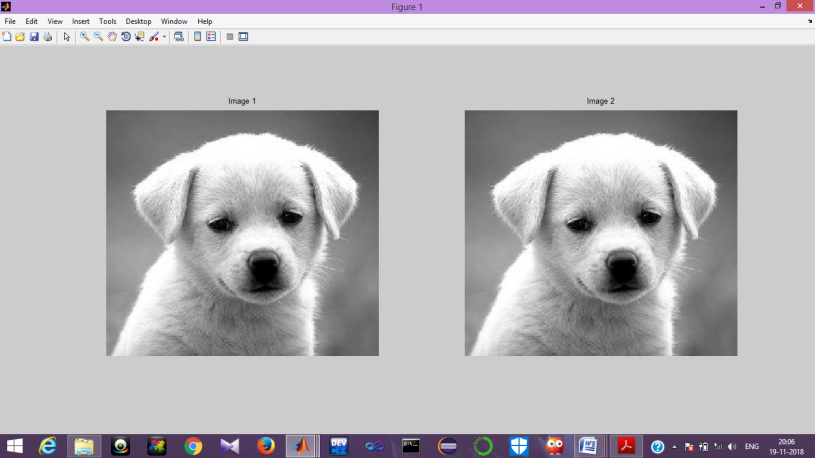
I=hsv2rgb(I);

imshow(I);

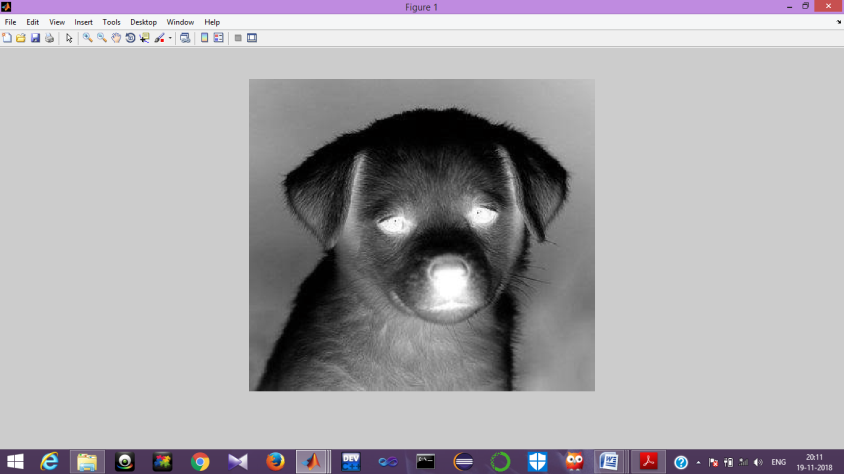
**Output:**  fig 4

****

**Fig 1**

****

**Fig 2**

****

**Fig 3**

**ASSIGNMENT 2**

1. Write a program to display an image with its corresponding negative image.

**Background:** RGB image is converted into grayscale imageThe following code displays the negative of the image along with the original image.

**Functions used:**

* imread():It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshowpair( , ,’montage’):It creates composite image overlaid in different colour bands. It uses the visualization method specified. ‘montage’ places the images next to each other.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

J=255-I;

imshowpair(I,J,'montage');

**Output**: fig 4

2. Write a program to plot sine graph with a marker and color.

**Background:** It plots a sine graph with marker and specified colour.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* plot(): It plots function in window. Syntax: plot(x,y)
* sin(): It generates sine function. Syntax: sin(x)

**Source Code:**

t=1:0.01:5;

a=sin(t);

plot(t,a,'b--o');

xlabel('x axis');

ylabel('y axis');

title('Sine');

**Output**: Fig 5

3. Write a program to plot Cosine graph with a marker and color.

**Background:** It plots a cosine graph with marker and specified colour.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* plot(): It plots function in window. Syntax: plot(x,y)
* sin(): It generates sine function. Syntax: sin(x)

**Source Code:**

t=1:0.01:5;

a=cos(t);

plot(t,a,'b--o');

xlabel('x axis');

ylabel('y axis');

title('Cosine');

**Output**: Fig 5

4. Write a program to crop an image and display it with the original one in a single frame.

**Background:**RGB image is converted into grayscale image .The following program crops an image and displays the original image along with the cropped image

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(filename)
* subplot():It divides the current figure into m by n grid and creates axes in the specified position.

Syntax: subplot(m,n,p)

* imcrop():It creates an interactive crop image tool associated with the image.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I2=imcrop(I,[75,68,130,112]);

subplot(1,2,1);

imshow(I2);

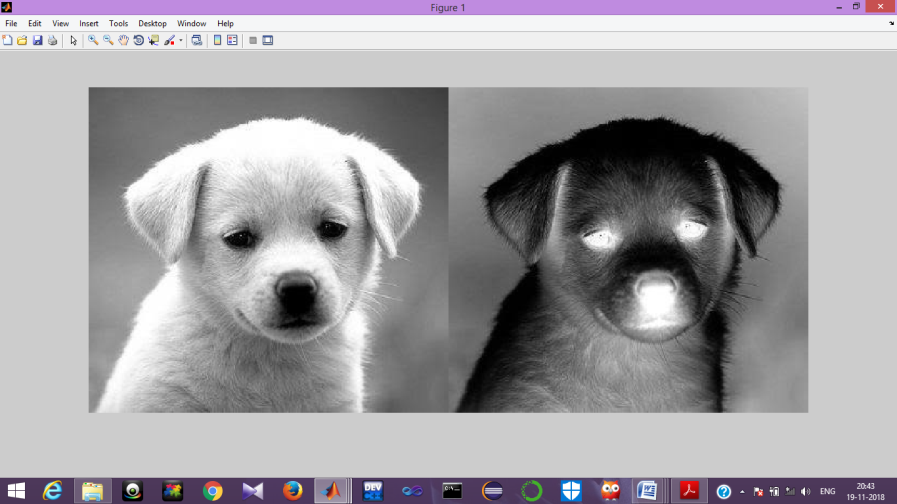
title('Cropped image');

subplot(1,2,2);

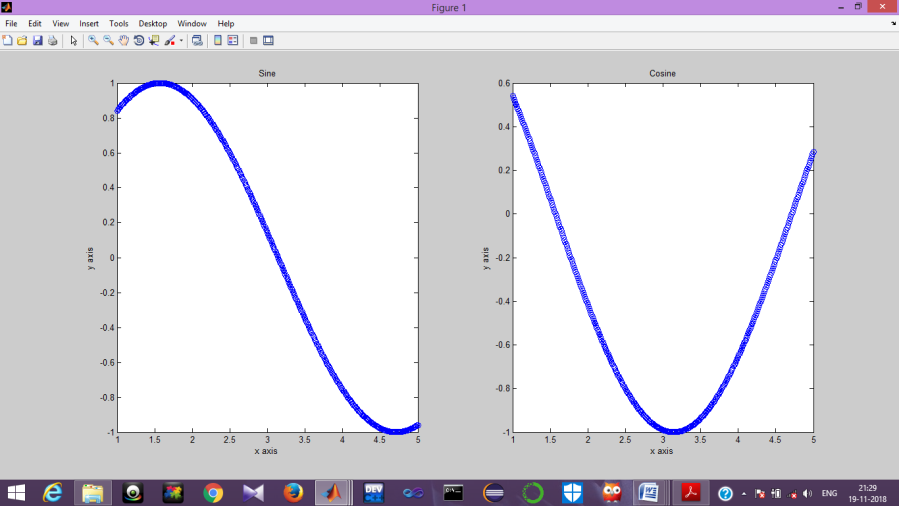
imshow(I);

title('Original image');

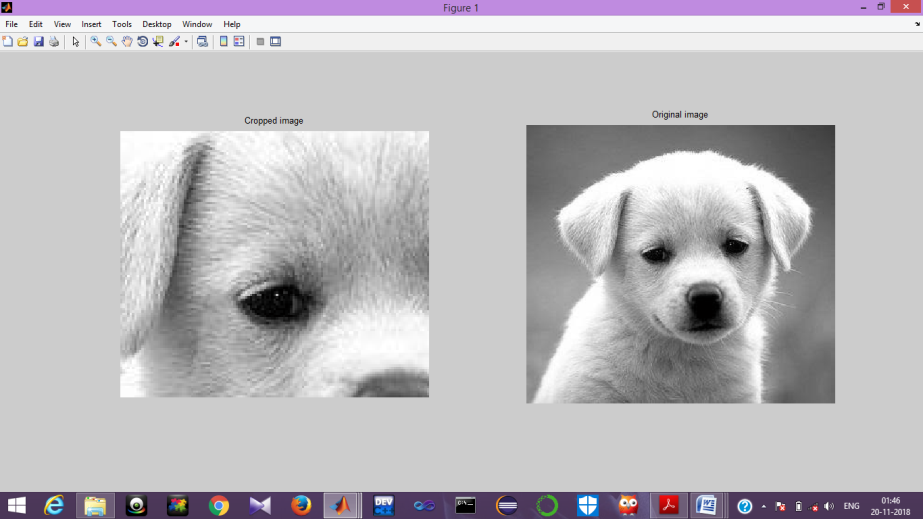
**Output:** Fig 6

****

**Fig 4**

****

**Fig 5**

****

**Fig 6**

**ASSIGNMENT 3**

1. Write a program to display an image with salt and pepper noise.

**Background:**RGB image is converted into grayscale image. It adds salt and pepper noise to the image and display the image.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* imnoise(I,’salt& pepper’): It adds salt and pepper noise with default noise density 0.05. It affects 5% of pixels.

**Source code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=imnoise(I,’salt& pepper’,0.02);

imshow(J);

**Output:** Fig 7

2. Write a program to read a noise less image and display a blur image.

**Background:**RGB image is converted into grayscale image. The image must be noiseless.The image is blurred and displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

l=31;t=11;

p=fspecial('motion',l,t);

b=imfilter(I,p);

imshow(b);

**Output:** Fig 7

3. Implement disk,mean and Gaussian filter.

**Background:**RGB image is converted into grayscale image. Filters such as disk, mean and Gaussian filter are applied on the image. The image is displayed in a window.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(filename)
* subplot():It divides the current figure into m by n grid and creates axes in the specified position.Syntax: subplot(m,n,p)
* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter
* medfilt2(): It performs median filtering of the image in 2 dimensions.

Syntax:medfilt2(image)

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

subplot(1,3,1);

p=fspecial('disk',5);

b=imfilter(I,p);

imshow(b);

title('Disk Filter');

subplot(1,3,2);

p=fspecial('gaussian',5,5);

b=imfilter(I,p);

imshow(b);

title('Gaussian Filter');

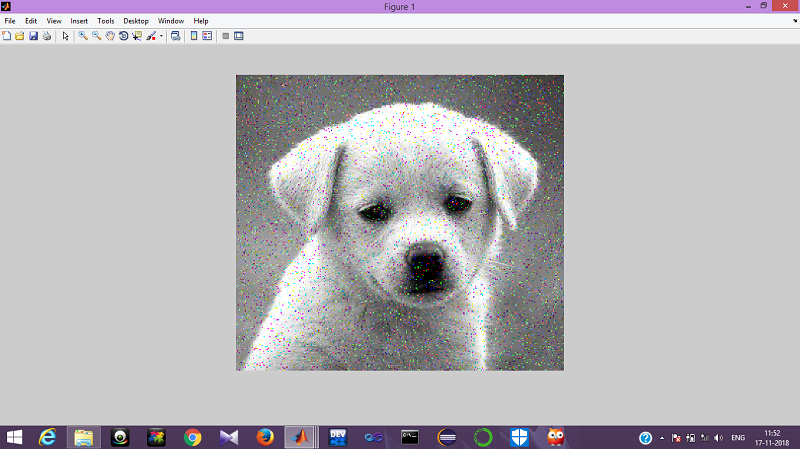
subplot(1,3,3);

J=medfilt2(I);

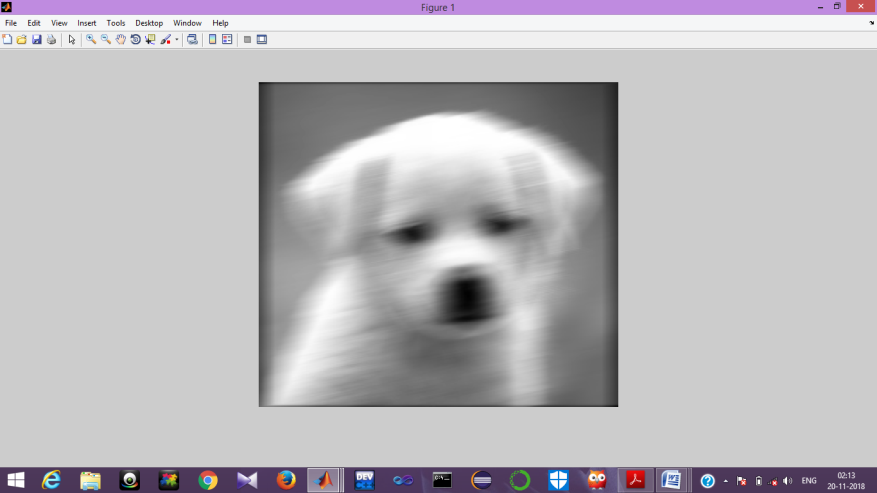
imshow(J);

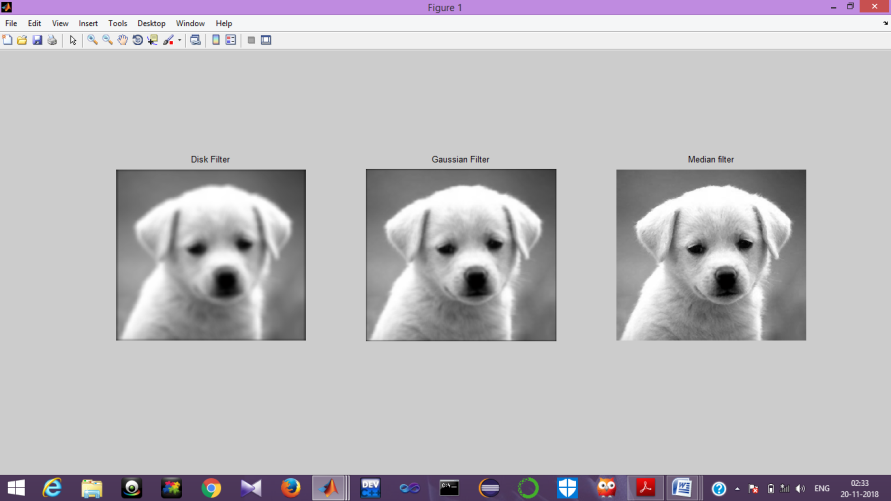
title('Median filter');

**Output**: Fig 8

****

**Fig 7**

****

****

**Fig 8**

**ASSIGNMENT 4**

1. Write a program to filter the image with isotropic Gaussian smoothing.

**Background:** RGB image is converted into grayscale image. Gaussian filter is applied on the image. The filtered image along with the original image is displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(filename)
* subplot():It divides the current figure into m by n grid and creates axes in the specified position.Syntax: subplot(m,n,p)
* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

p=fspecial('gaussian',5,5);

b=imfilter(I,p);

imshow(b);

title('Gaussian Filter');

**Output:** Fig 9

2. Write a program to implement motion blur.

**Background:**RGB image is converted into grayscale image. The image is blurred and displayed in a window.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

l=31;t=11;

p=fspecial('motion',l,t);

b=imfilter(I,p);

imshow(b);

**Output:** Fig 10

3. Write down the code to generate sine, cosine, and magic 4.

**Background:**Sine,cosine and magic4 functions are plotted in the same window.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* plot(): It plots function in window. Syntax: plot(x,y)
* sin(): It generates sine function. Syntax: sin(x)

**Source Code:**

t=1:0.01:5;

subplot(1,3,1);

a=sin(t); plot(t,a);

xlabel('x axis');ylabel('y axis');

title('Sine');

subplot(1,3,2);

a=cos(t); plot(t,a);

xlabel('x axis'); ylabel('y axis');

title('Cosine');

subplot(1,3,3);

k=magic(4); plot(k);

title('Magic 4');

**Output:** Fig 11

4. Write a program for image smoothening with mean filter using salt and pepper noise.

**Background:** RGB image is converted into grayscaleimage.Salt and pepper noise is introduced in the image. The image is smoothened using mean filter and displayed in a window.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* imnoise(I,’salt& pepper’): It adds salt and pepper noise with default noise density 0.05. It affects 5% of pixels.
* medfilt2(): It performs median filtering of the image in 2 dimensions.

Syntax:medfilt2(image)

**Source Code:**

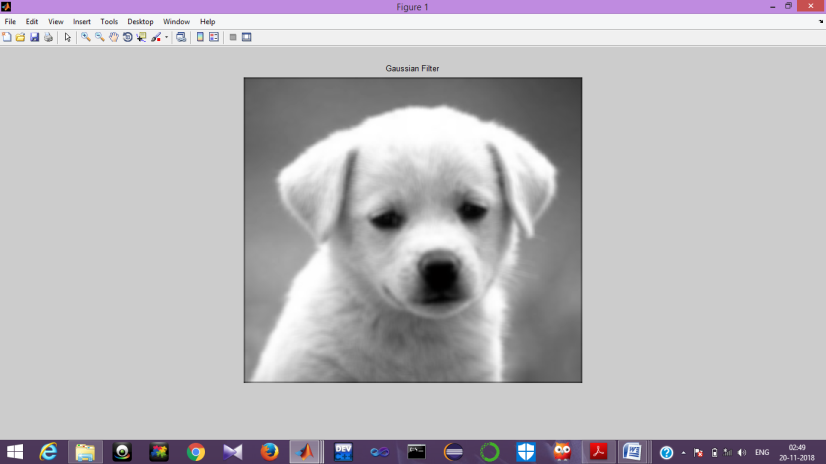
I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

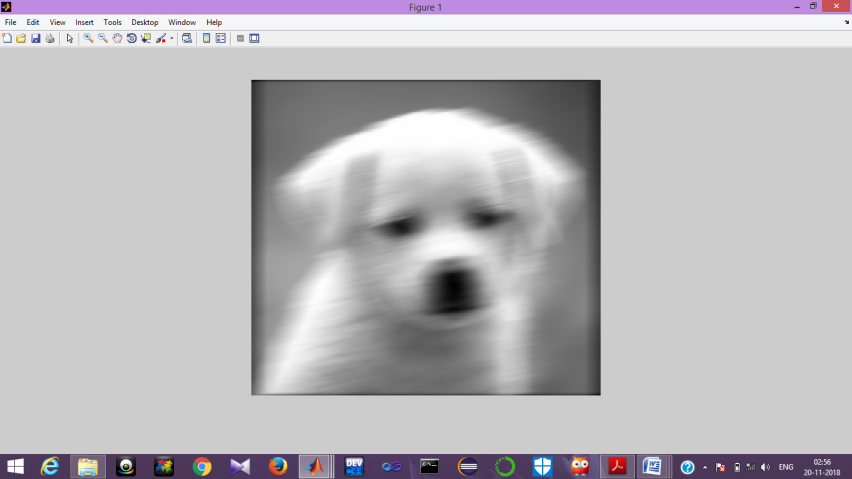
J=imnoise(I,'salt& pepper',0.02);

J=medfilt2(J);

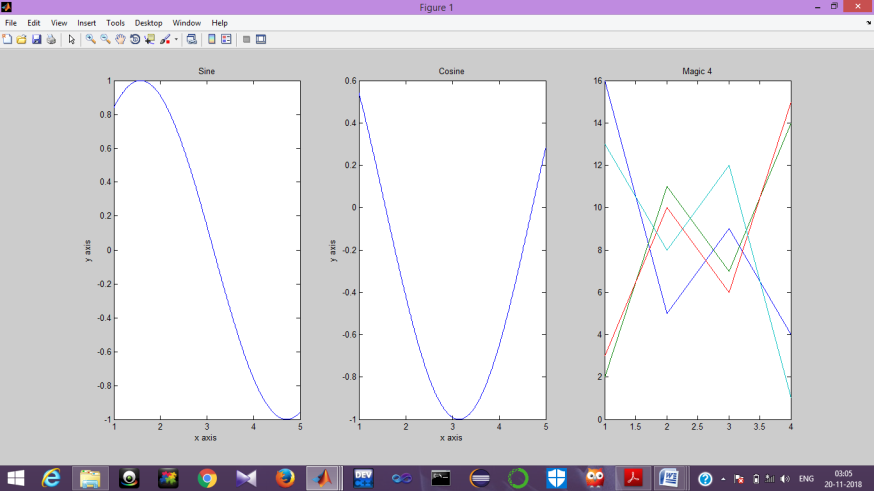
imshow(J);

****

**Fig 9**

****

**Fig 10**

****

**Fig 11**

**ASSIGNMENT 5**

1. Write a matlab function: [im\_q] = quant(im, N) that performs uniform

quantization of the image im to N gray levels (0≤N≤255).

**Background:** RGB image is converted into grayscale image. The image is quantized using specified gray levels. The quantized image is displayed in a window.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* imquantize(): It quantizes image using specified quantization levels and output values.

**Source Code:**

function [ imq ] = Untitled2( im,N )

s=imquantize(im,N);

imq=label2rgb(s);

imshow(imq);

end

**Output:** Fig 12

2. Write program to calculate the histogram values and display histograph for grayscale image

**Background :** RGB image is converted into grayscale image. The histogram of the image is displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* imhist(): It calculates the histogram for grayscaleimage.Syntax: imhist(Image)

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

imhist(I);

**Output:** Fig 13

3. Implement edge detection using Sobel, Roberts,Prewitt.

**Background:**RGB image is converted into grayscale image. Edge detection techniques are applied on the image. The filtered images are displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* edge(I,method): It detects edges in image using the edge detection algorithm specified by method.

Sobbel: Finds edge at those points where the gradient of the image is maximum.

Prewitt: Finds edge at those points where the gradient of the image is maximum.

Roberts: Finds edges by looking for maxima of the gradient of image.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

subplot(1,3,1);

J=edge(I,'sobel');

imshow(J);

title('Sobel');

subplot(1,3,2);

J=edge(I,'prewitt')

imshow(J);

title('Prewitt');

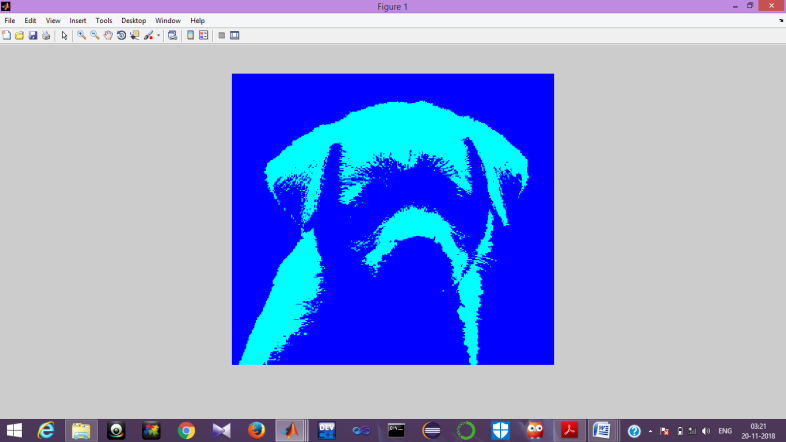
subplot(1,3,3);

J=edge(I,'roberts');

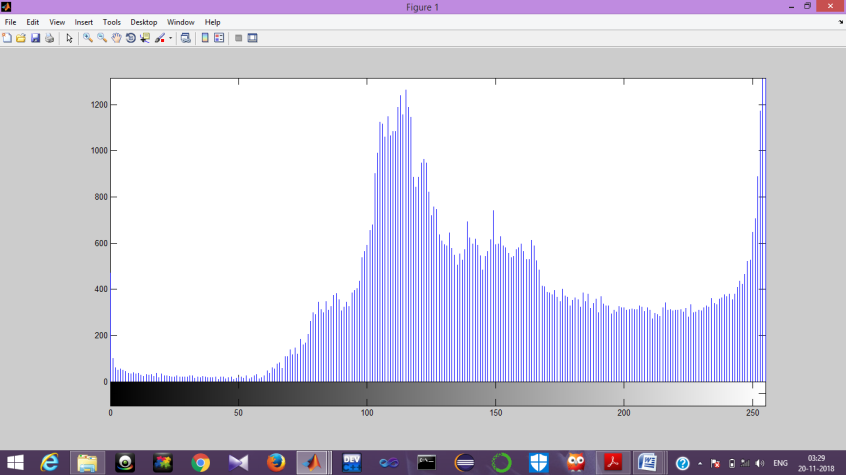
imshow(J);

title('Roberts');

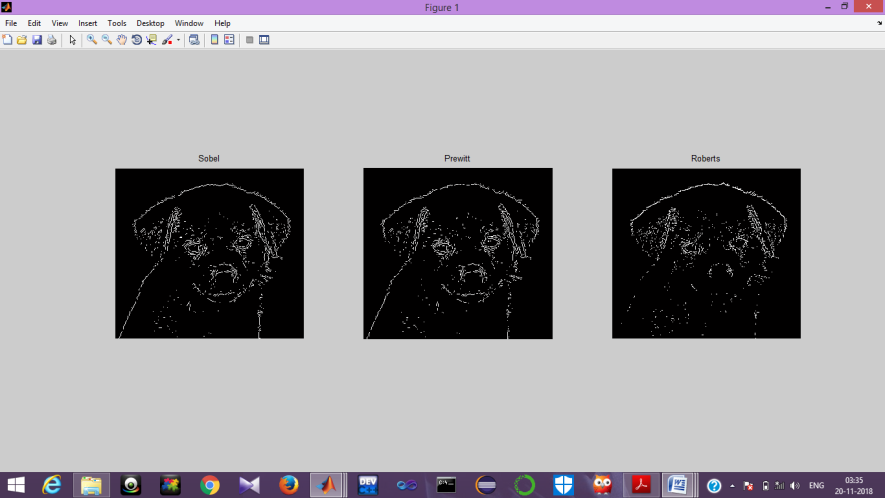
**Output:** Fig 13

****

**Fig 12**

****

**Fig 13**

****

**Fig 14**

**ASSIGNMENT 6**

1.Write a program to implement Laplacian-Gaussian edge detection operator

**Background:**RGB image is converted into grayscale image. Laplacian Gaussian edge detection technique is applied on the image and displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* edge(I,method): It detects edges in image using the edge detection algorithm specified by method.

**Source code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=edge(I,'log');

imshow(J);

**Output:** Fig 15

2. Write a program to implement Canny edge detection operator.

**Background:** RGB image is converted into grayscale image. canny edge detection technique is applied on the image and the resultant image is displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* edge(I,method): It detects edges in image using the edge detection algorithm specified by method.

**Source code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=edge(I,'canny');

imshow(J);

**Output:** Fig 16

3. Write a program to plot tan graph with a marker and color.

**Background:** It plots a sine graph with marker and specified colour.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* plot(): It plots function in window. Syntax: plot(x,y)
* sin(): It generates sine function. Syntax: sin(x)

**Source Code:**

t=1:0.01:5;

a=tan(t);

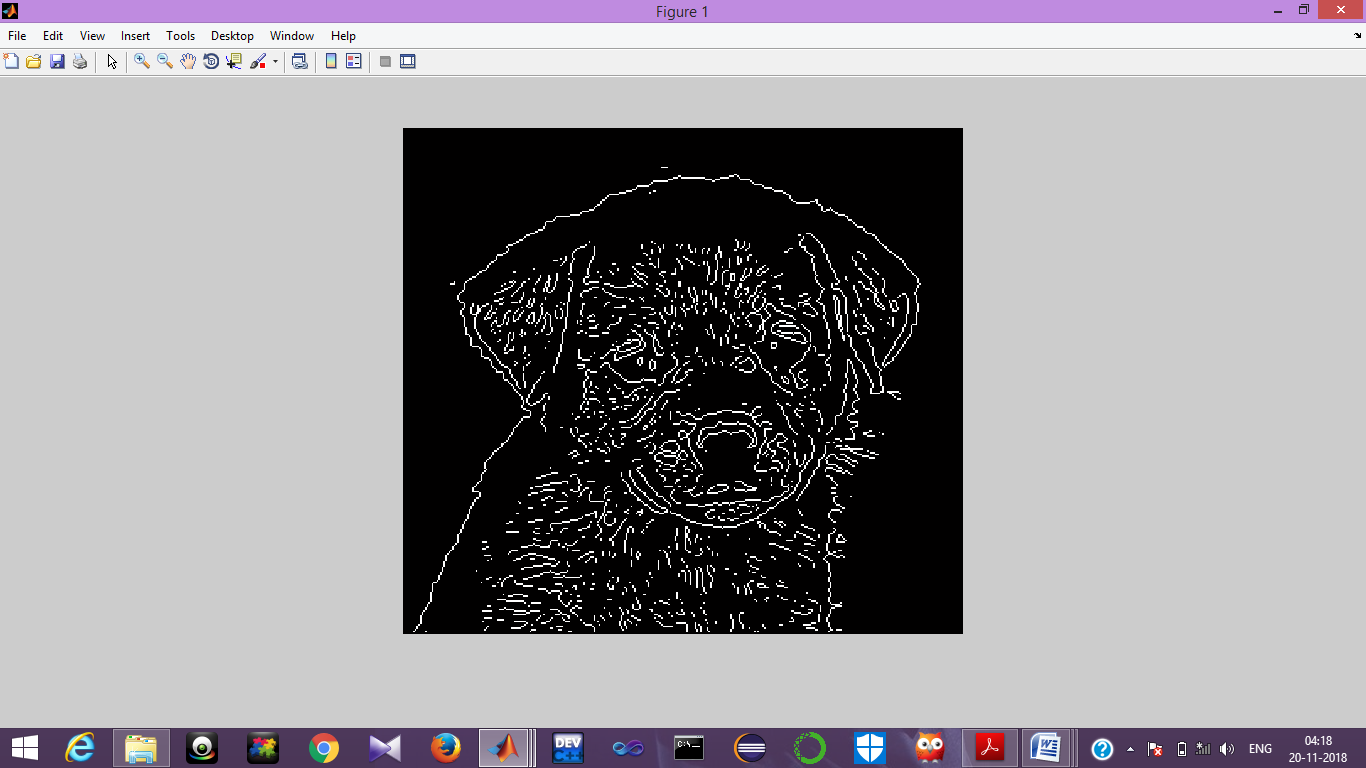
plot(t,a,'b--o');

xlabel('x axis');

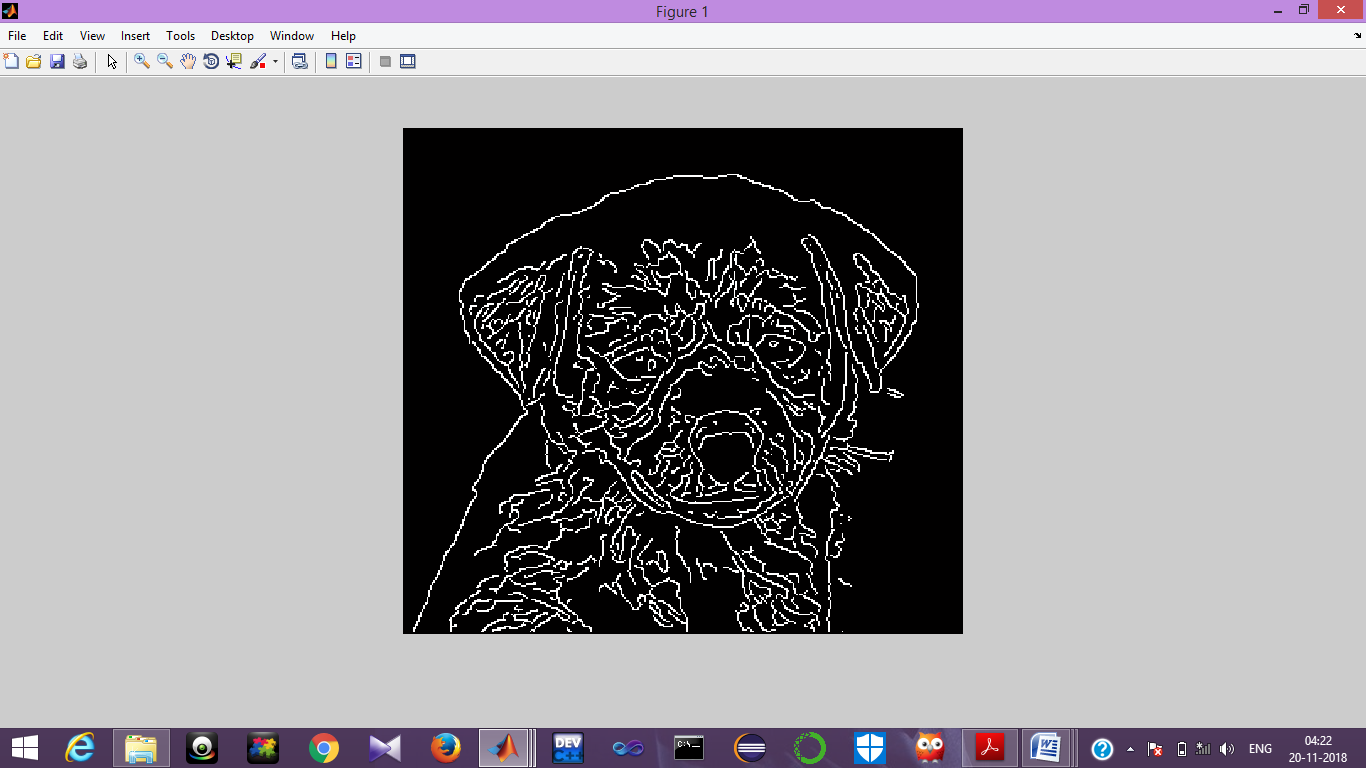
ylabel('y axis');

title(‘Tan');

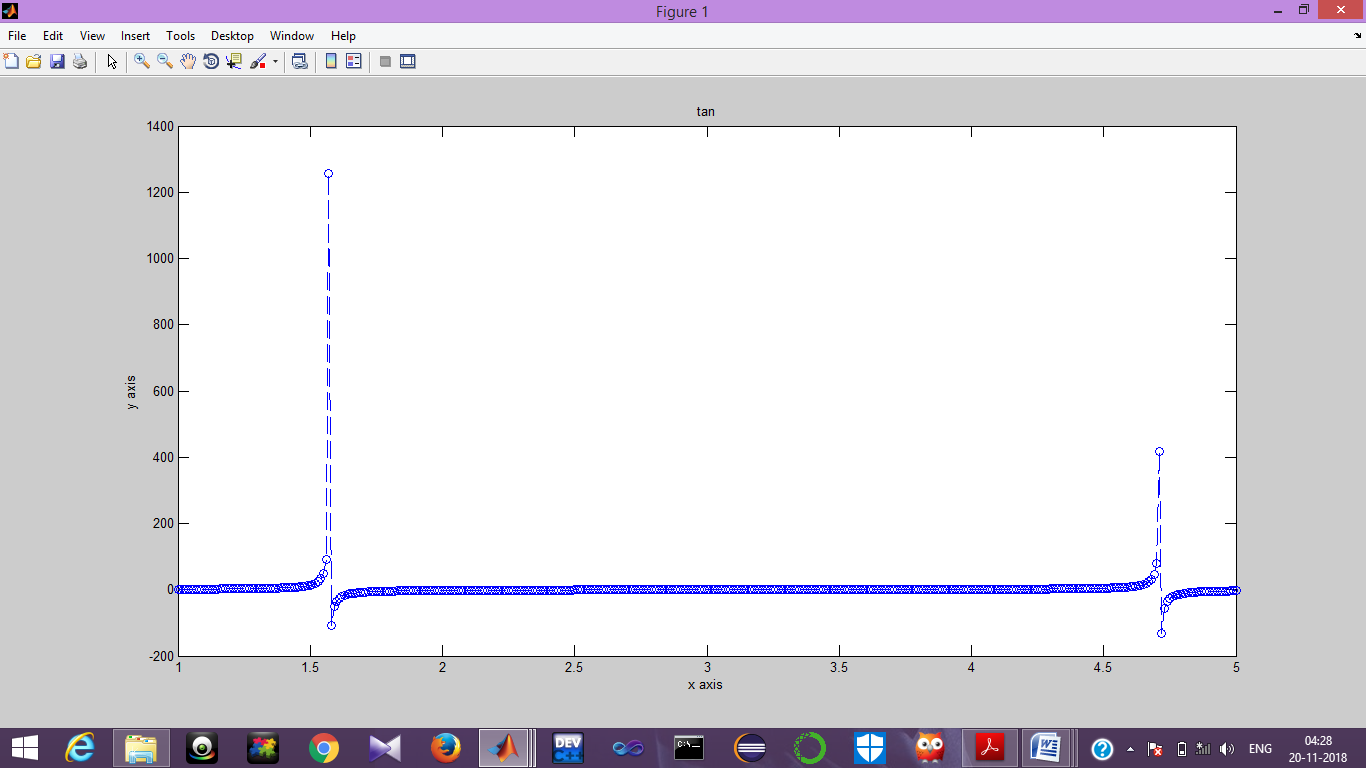
**Output**: Fig 17

****

**Fig 15**

****

**Fig 16**

****

**ASSIGNMENT 7**

1. Write a program to introduce Gaussian noise in an image and remove the noise using average filtering.

2. Write a program to introduce Gaussian noise in an image and remove the noise using median filtering.

3. Write a program to introduce Gaussian noise in an image and remove the noise using adaptive filtering.

4. Write a program to introduce Salt and Pepper noise in an image and remove the noise using average filtering.

**Background:** RGB image is converted into grayscale image. Gaussian noise is introduced in the image and different types of filtering are carried out.The resultant images are displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter
* imnoise(I,’salt&pepper’,d): It adds salt and pepper noise with d as noise density. It affects approximately d\*numel(I) of pixels.
* wiener2(): It is used for adaptive filtering

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

K=imnoise(I,'gaussian');

subplot(2,2,1);

p=filter2(fspecial('average',3),K)/255;

imshow(p);title('Average filter');

subplot(2,2,2);p=medfilt2(K);

imshow(p);

title('Median filter');

subplot(2,2,3);

p=wiener2(K,[5,5]);

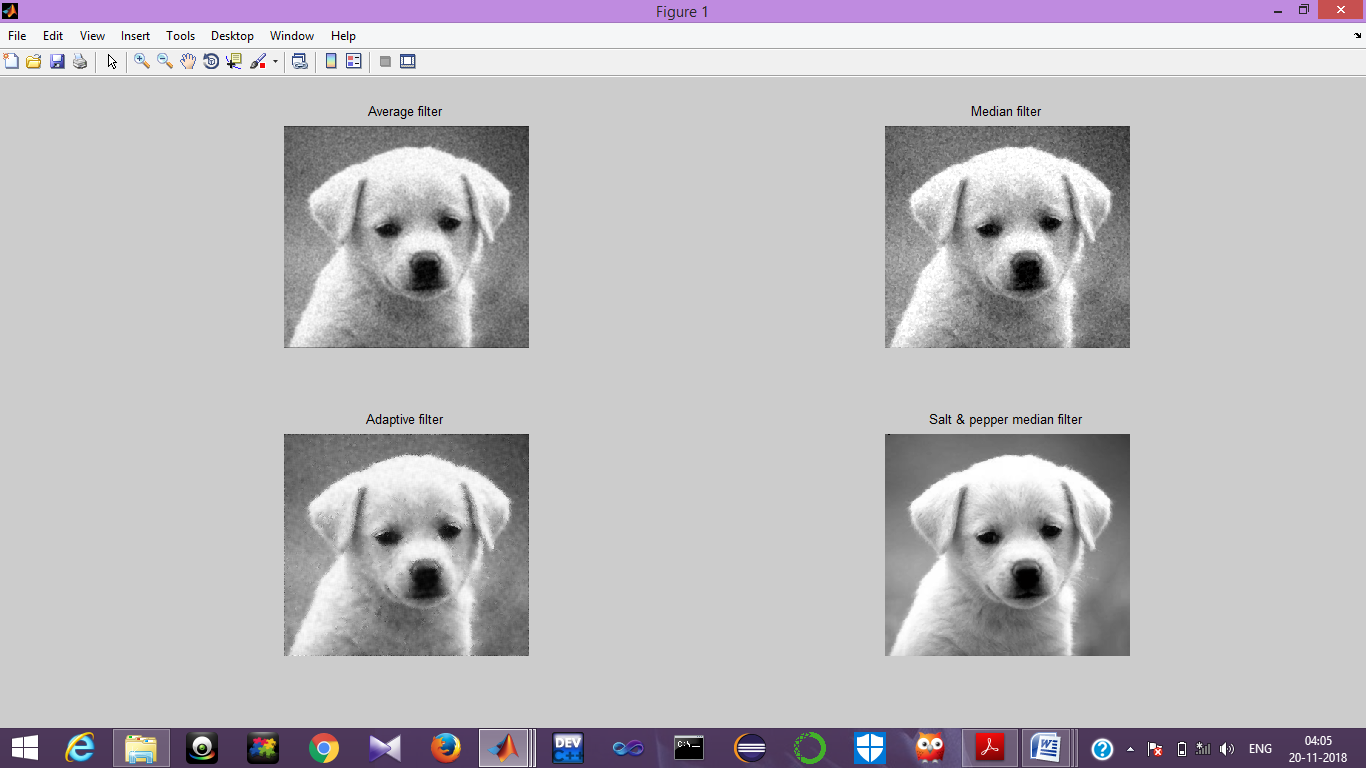
imshow(p);

title('Adaptive filter');subplot(2,2,4);

J=imnoise(I,'salt& pepper',0.02);J=medfilt2(J);

imshow(J);

title('Salt & pepper median filter');

****

**ASSIGNMENT 8**

1. Write a program to introduce Salt and Pepper noise in an image and remove the noise using median filtering.

**Background:**RGB image is converted into grayscaleimage.Noise is introduced in the image and it is removed using median filtering.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* imnoise(I,’salt&pepper’,d): It adds salt and pepper noise with d as noise density. It affects approximately d\*numel(I) of pixels.
* medfilt2(): It performs median filtering of the image in 2 dimensions.

Syntax:medfilt2(image)

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=imnoise(I,'salt& pepper',0.02);

K=medfilt2(J);

imshowpair(J,K,'montage');

OutPut: Fig 18

2.Write a program to implement Sobel edge detector on one particular image which is preprocessed by the method of average filtering.

3. Write a program to implement Canny edge detector on one particular image which is preprocessed by the method of average filtering.

4. Write a program to implement Roberts edge detector on one particular image which is preprocessed by the method of average filtering

**Background:** RGB image is converted into grayscale image. Edge dectection techniques are used on the image and the filtered images are displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents.

Syntax: imread(filename)

* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.

Syntax: imshow(Image)

* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter
* edge(I,method): It detects edges in image using the edge detection algorithm specified by method.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=filter2(fspecial('average',3),K)/255;

i=edge(I,'sobel');

j=edge(I,'canny');

k=edge(I,'roberts');

subplot(1,3,1);

imshow(i);

title('Sobel');

subplot(1,3,2);

imshow(j);

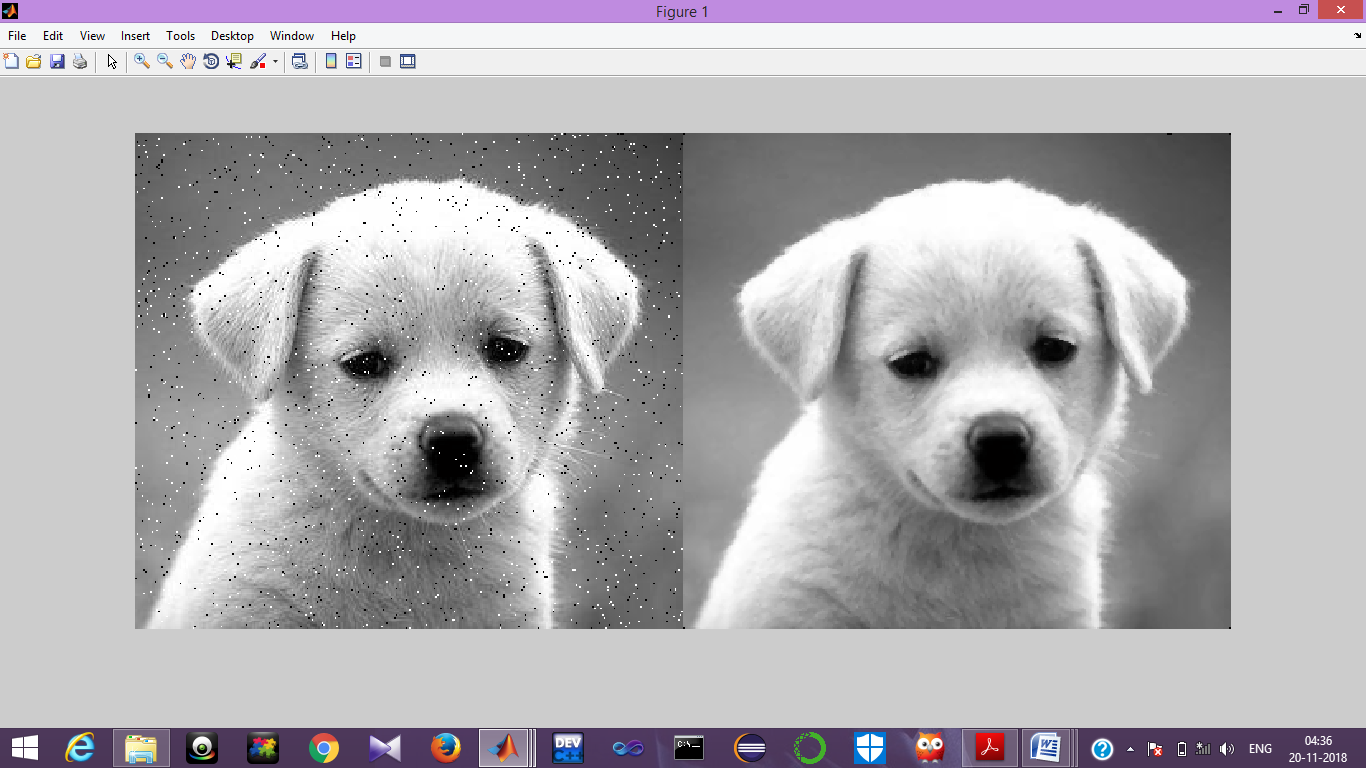
title('Canny');

subplot(1,3,3);

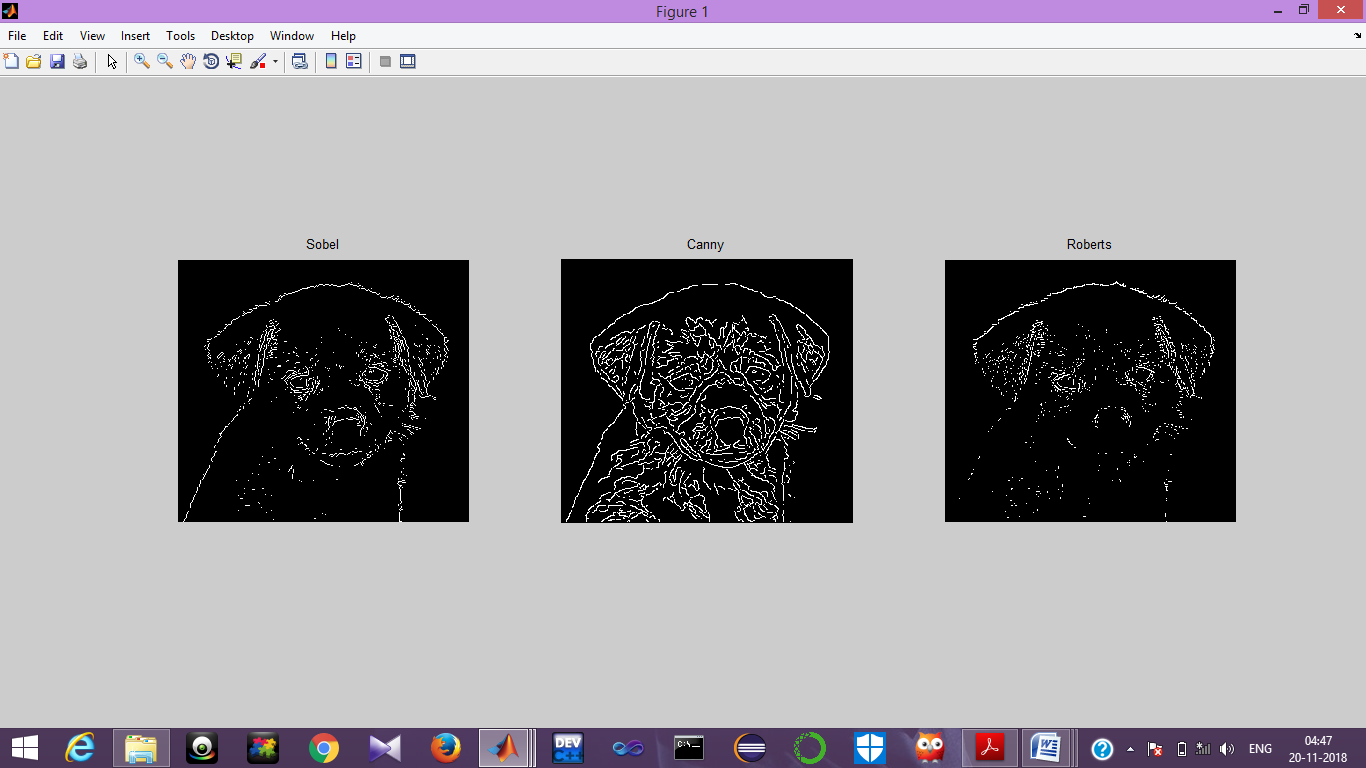
imshow(k);

title('Roberts');

**Output:** Fig 19

****

**Fig 18**

****

**Fig 19**

**ASSIGNMENT 9**

1.Write a program to implement Prewitt edge detector on one particular image which is preprocessed by the method of average filtering.

2. Write a program to implement Laplacian-Gaussian edge detector on one particular image which is preprocessed by the method of average filtering.

3. Write a program to implement Sobel edge detector on one particular image which is preprocessed by the method of mean filtering.

4. Write a program to implement Canny edge detector on one particular image which is preprocessed by the method of mean filtering.

5. Write a program to implement Roberts edge detector on one particular image which is preprocessed by the method of mean filtering.

**Background:** RGB image is converted into grayscale image. Edge dectection techniques are used on the image and the filtered images are displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* fspecial():It creates a two dimensional filter h of the specified type.
* imfilter(): It filters the multidimensional array with multidimensional filter
* edge(I,method): It detects edges in image using the edge detection algorithm specified by method.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=filter2(fspecial('average',3),K)/255;

i=edge(I,'prewitt');m=edge(I,'log');n=edge(I,'sobel');j=edge(I,'canny');k=edge(I,'roberts');

subplot(2,3,1);imshow(I);

title('Original');

subplot(2,3,2);imshow(i);

title('Prewitt');

subplot(2,3,3);imshow(m);

title('Laplacian');

subplot(2,3,4);imshow(n);

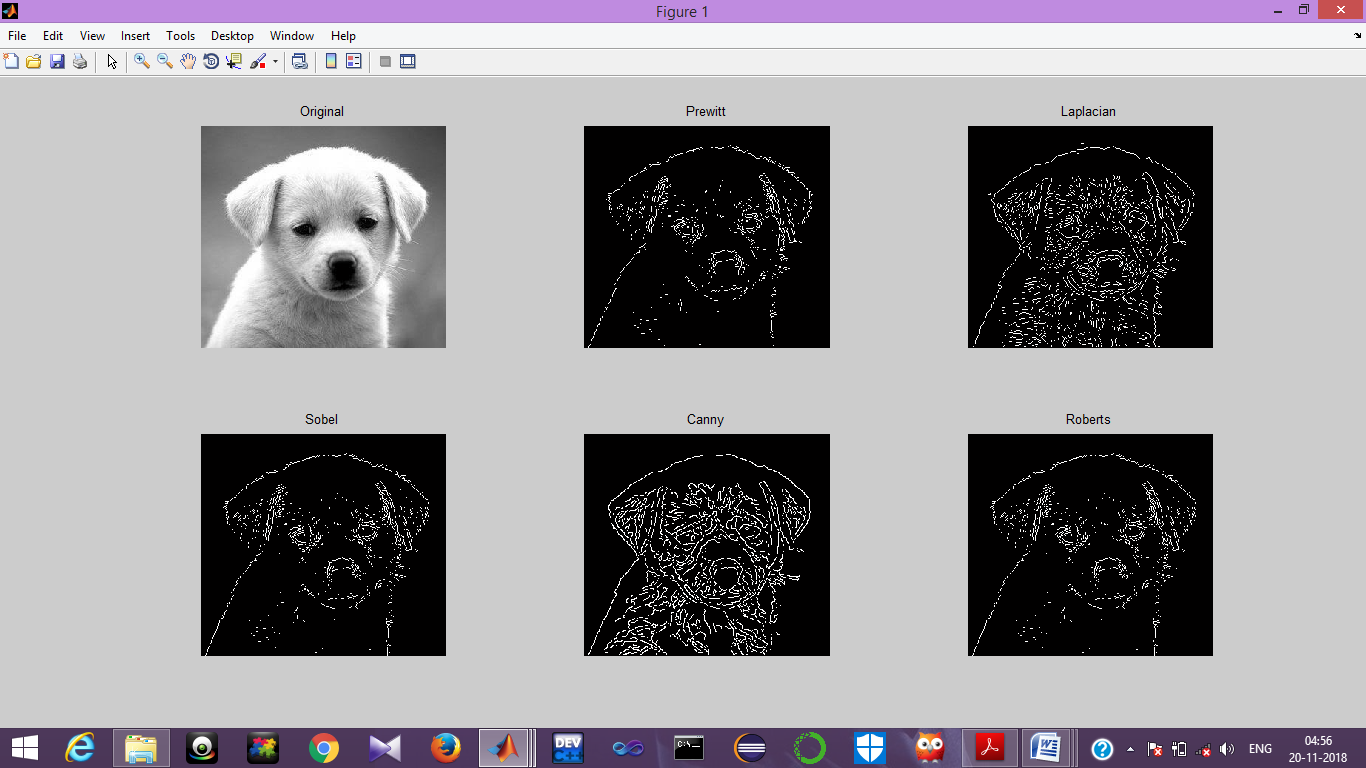
title('Sobel');

subplot(2,3,5);imshow(j);

title('Canny');

subplot(2,3,6);imshow(n);

title('Roberts');

****

**ASSIGNMENT 10**

1.Write a program to implement Prewitt edge detector on one particular image which is preprocessed by the method of mean filtering.

2. Write a program to implement Laplacian-Gaussian edge detector on one particular image which is preprocessed by the method of mean filtering.

3. Write a program to implement Sobel edge detector on one particular image which is preprocessed by the method of median filtering.

4. Write a program to implement Canny edge detector on one particular image which is preprocessed by the method of median filtering.

5. Write a program to implement Roberts edge detector on one particular image which is preprocessed by the method of median filtering.

**Background:** RGB image is converted into grayscale image. Edge dectection techniques are used on the image and the filtered images are displayed.

**Functions used:**

* imread(): It reads the image from the file specified, inferring the format of the file from its contents. Syntax: imread(filename)
* imshow(): It displays the image. It optimizes figure,axes and image object properties for image display.Syntax: imshow(Image)
* medfilt2(): It performs median filtering of the image in 2 dimensions.

Syntax:medfilt2(image)

* edge(I,method): It detects edges in image using the edge detection algorithm specified by method.

**Source Code:**

I=imread('C:\Users\USER\Desktop\pic\im.jpeg');

I=rgb2gray(I);

J=medfilt2(I);

i=edge(I,'prewitt');

m=edge(I,'log');

n=edge(I,'sobel');

j=edge(I,'canny');

k=edge(I,'roberts');

subplot(2,3,1);

imshow(I);

title('Original');

subplot(2,3,2);

imshow(i);

title('Prewitt');

subplot(2,3,3);

imshow(m);

title('Laplacian');

subplot(2,3,4);

imshow(n);

title('Sobel');

subplot(2,3,5);

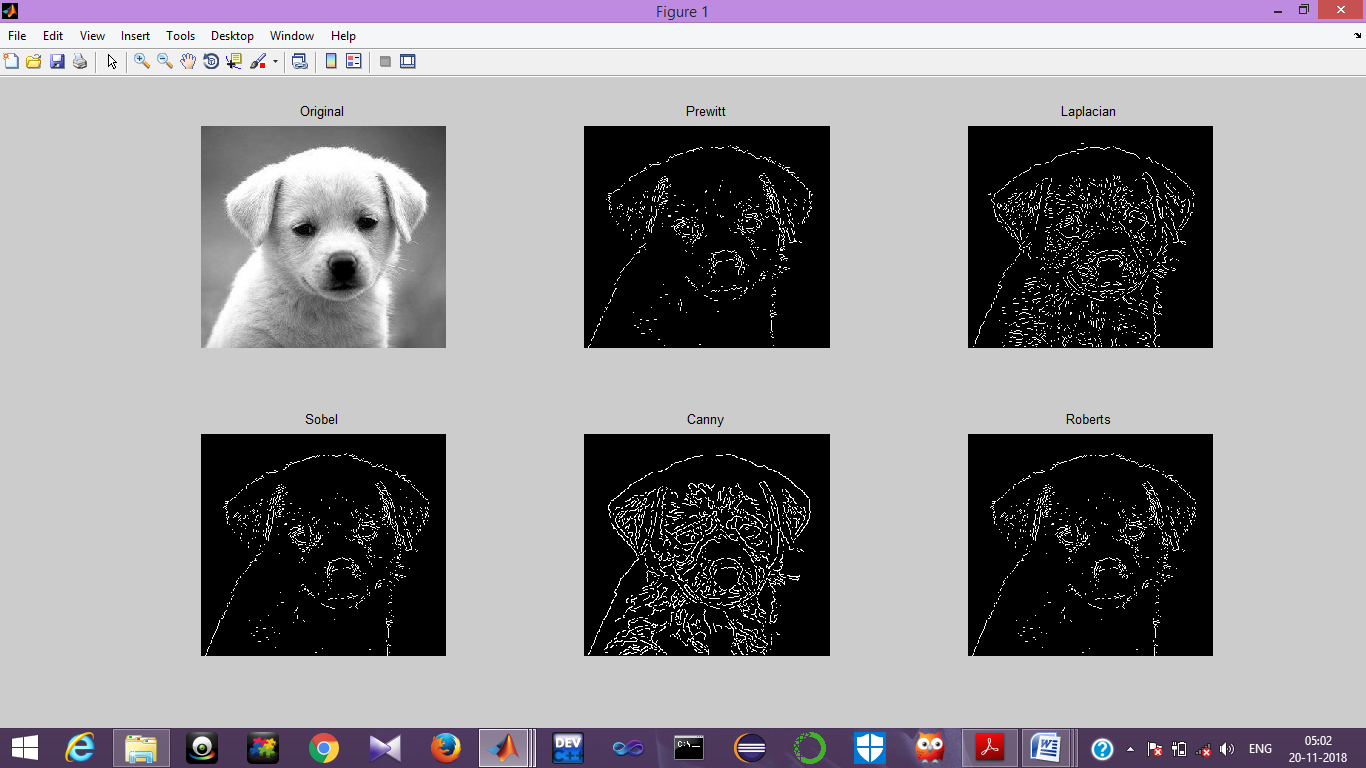
imshow(j);

title('Canny');

subplot(2,3,6);

imshow(n);

title('Roberts');

****