**Aim: To understand and implement New border images by applying Edge detection operators using MATLAB**

**Matlab Code:**

clc;

close all;

clear all;

% Read Colour Image and convert it to a grey level Image

% Display the original Image

mycolourimage = imread('grayflower256.jpg');

myimage = rgb2gray(mycolourimage);

subplot(3,3,1);

imshow(myimage); title('Original Image');

% Apply Sobel Operator

% Display only the horizontal Edges

sobelhz = edge(myimage,'sobel','horizontal');

subplot(3,3,2);

imshow(sobelhz,[]); title('Sobel - Horizontal Edges');

% Apply Sobel Operator

% Display only the vertical Edges

sobelvrt = edge(myimage,'sobel','vertical');

subplot(3,3,3);

imshow(sobelhz,[]); title('Sobel - Vertical Edges');

% Apply Sobel Operator

% Display both horizontal and vertical Edges

sobelvrthz = edge(myimage,'sobel','both');

subplot(3,3,4);

imshow(sobelvrthz,[]); title('Sobel - All edges');

% Apply Roberts Operator

% Display both horizontal and vertical Edges

robertsedg = edge(myimage,'roberts');

subplot(3,3,5);

imshow(robertsedg,[]); title('Roberts - Edges');

% Apply Prewitt Operator

% Display both horizontal and vertical Edges

robertsedg = edge(myimage,'prewitt');

subplot(3,3,6);

imshow(robertsedg,[]); title('Prewitt - Edges');

% Apply Laplacian Filter

f=fspecial('laplacian');

lapedg = imfilter(myimage,f,'symmetric');

subplot(3,3,7);

imshow(lapedg,[]); title('Laplacian Filter');

% Apply LOG edge detection

% The sigma used is 3

f=fspecial('log',[15,15],3.0);

logedg1 = edge(myimage,'zerocross',[],f);

subplot(3,3,8);

imshow(logedg1); title('Log with sigma 3');

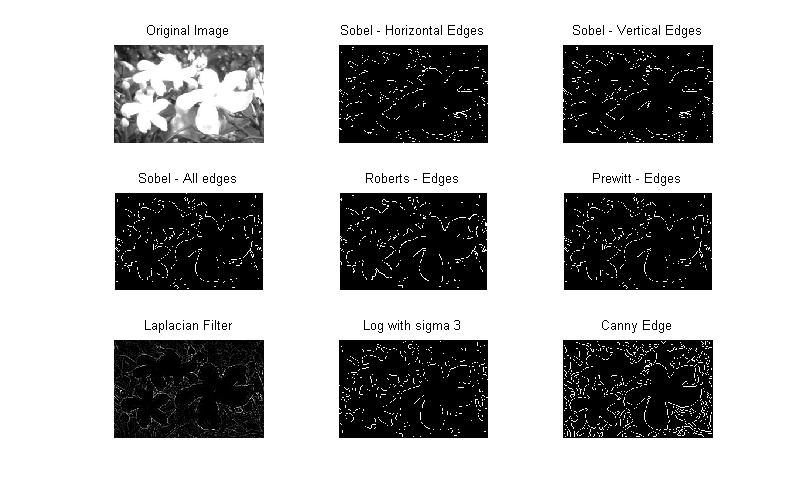
% Apply Canny edge detection

cannyedg = edge(myimage,'canny');

subplot(3,3,9);

imshow(cannyedg,[]); title('Canny Edge');

**Results:**



**Conclusion:** Thus we have concluded from this experiment that edges can be implemented and extracted for synthetic images.