**Task:**

**Implement the following filters in MATLAB without using MATLAB build-in functions:  
•Canny  
•Sobel  
•Roberts  
•Prewitt**

**Robert Filter:**

clc;

clear

clc

close all

a=imread('C:\Users\user\Desktop\Matlab\cameraman.tif');

b=im2double(a);

[m,n]=size(a);

%ROBERT

L(1:m,1:n)=0;

for i=1:m-2

for j=1:m-2

L(i,j)=-1\*b(i,j)+0+0+1\*b(i+1,j+1);

end

end

M(1:m,1:n)=0;

for i=1:m-2

for j=1:m-2

M(i,j)=0-1\*b(i,j+1)+1\*b(i+1,j)+0;

end

end

figure;

subplot(2,2,1)

imshow(L)

title('Robert Gx');

subplot(2,2,2)

imshow(M)

title('Robert Gy');

N=M+L;

subplot(2,2,3)

imshow(N)

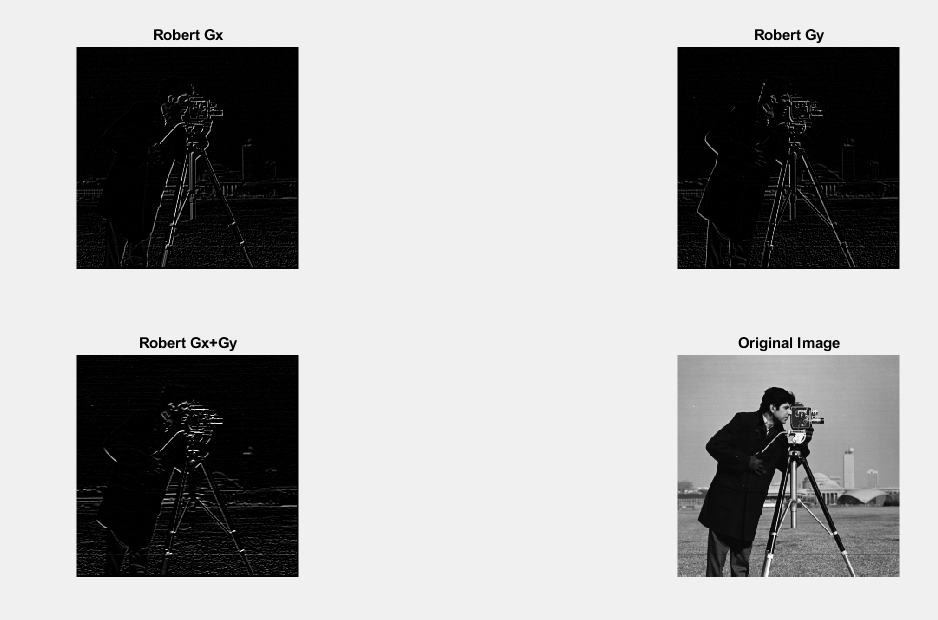
title('Robert Gx+Gy');

subplot(2,2,4)

imshow(b)

title('Original Image');

**Output:**



**Prewit Filter:**

clc;

clear

clc

close all

original =imread('C:\Users\user\Desktop\Matlab\cameraman.tif');

dblimg=im2double(original);

[m,n]=size(original);

%PREWIT

N(1:m,1:n)=0;

for i=1:m-2

for j=1:m-2

N(i,j)=-1\*dblimg(i,j)-1\*dblimg(i,j+1)-1\*dblimg(i,j+2)+0+0+0+1\*dblimg(i+2,j)+1\*dblimg(i+2,j+1)+1\*dblimg(i+2,j+2);

end

end

O(1:m,1:n)=0;

for i=1:m-2

for j=1:m-2

O(i,j)=-1\*dblimg(i,j)+0+1\*dblimg(i,j+2)-1\*dblimg(i+2,j)+0+1\*dblimg(i+1,j+2)-1\*dblimg(i+2,j)+0+1\*dblimg(i+2,j+2);

end

end

figure;

subplot(2,2,1)

imshow(N)

title('Prewit Gx');

subplot(2,2,2)

imshow(O)

title('Prewit Gy');

Z=N+O;

subplot(2,2,3)

imshow(Z)

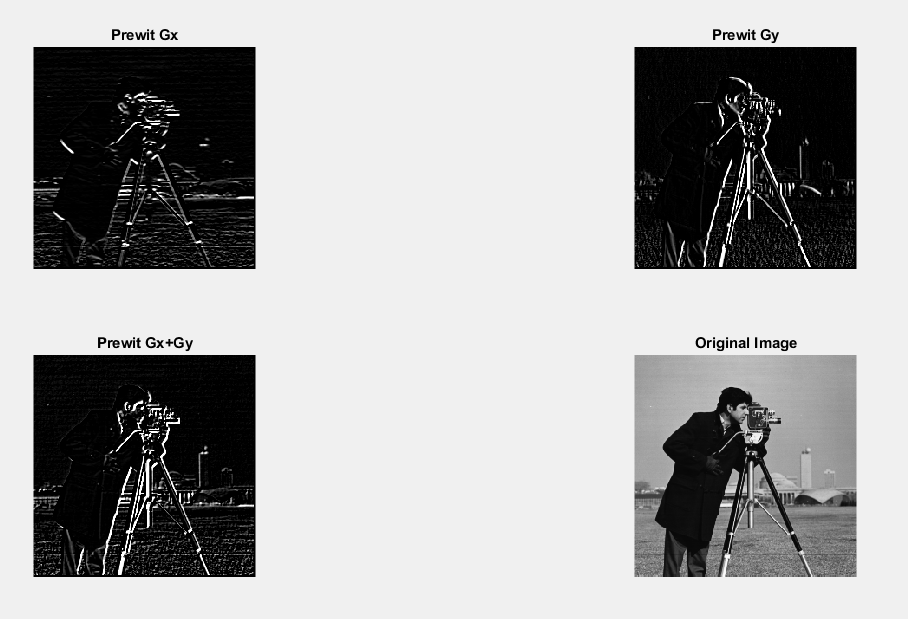
title('Prewit Gx+Gy');

subplot(2,2,4)

imshow(dblimg)

title('Original Image');

**Output:**



**Sobel Filter:**

clear

clc

close all

original =imread('C:\Users\user\Desktop\Matlab\cameraman.tif');

dblImg=im2double(original);

[m,n]=size(original);

P(1:m,1:n)=0;

for i=1:m-2

for j=1:m-2

P(i,j)=-1\*dblImg(i,j)-2\*dblImg(i,j+1)-1\*dblImg(i,j+2)+0+0+0+1\*dblImg(i+2,j)+2\*dblImg(i+2,j+1)+1\*dblImg(i+2,j+2);

end

end

R(1:m,1:n)=0;

for i=1:m-2

for j=1:m-2

R(i,j)=-1\*dblImg(i,j)+0+1\*dblImg(i,j+2)-2\*dblImg(i+1,j)+0+2\*dblImg(i+1,j+2)-1\*dblImg(i+2,j)+0+1\*dblImg(i+2,j+2);

end

end

figure;

subplot(2,2,1)

imshow(P)

title('Sobel Gx');

subplot(2,2,2)

imshow(R)

title('Sobel Gy');

Y=P+R;

subplot(2,2,3)

imshow(Y)

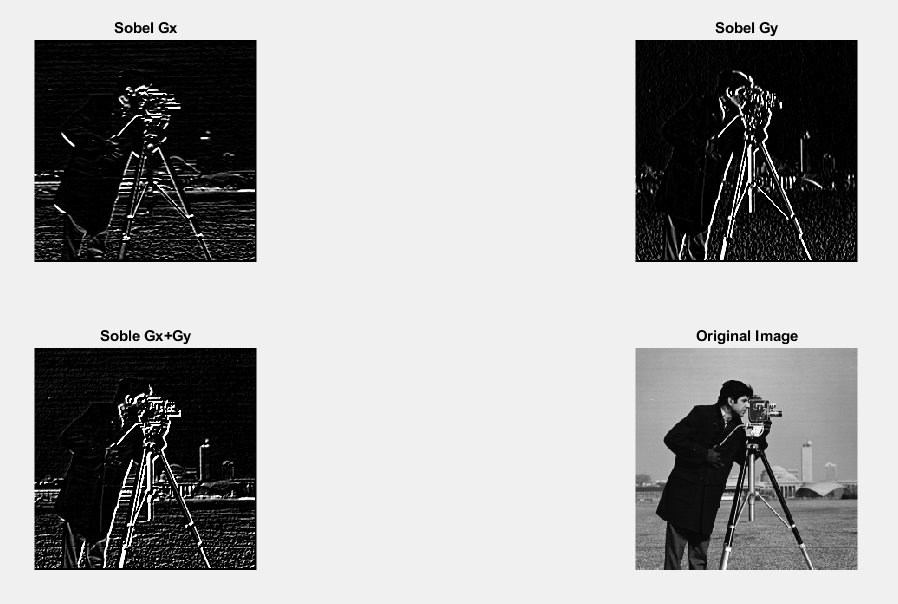
title('Soble Gx+Gy');

subplot(2,2,4)

imshow(dblImg)

title('Original Image');

**Result:**



**Canny Filter:**

clear all

close all

clc

img = imread('C:\Users\user\Desktop\Matlab\cameraman.tif');

I=double(img); %read image

In=I;

mask1=[1, 0, -1;1, 0, -1;1, 0, -1];

mask2=[1, 1, 1;0, 0, 0;-1, -1, -1];

mask3=[0, -1, -1;1, 0, -1;1, 1, 0];

mask4=[1, 1, 0;1, 0, -1;0, -1, -1];

mask1=flipud(mask1);

mask1=fliplr(mask1);

mask2=flipud(mask2);

mask2=fliplr(mask2);

mask3=flipud(mask3);

mask3=fliplr(mask3);

mask4=flipud(mask4);

mask4=fliplr(mask4);

for i=2:size(I, 1)-1

for j=2:size(I, 2)-1

neighbour\_matrix1=mask1.\*In(i-1:i+1, j-1:j+1);

avg\_value1=sum(neighbour\_matrix1(:));

neighbour\_matrix2=mask2.\*In(i-1:i+1, j-1:j+1);

avg\_value2=sum(neighbour\_matrix2(:));

neighbour\_matrix3=mask3.\*In(i-1:i+1, j-1:j+1);

avg\_value3=sum(neighbour\_matrix3(:));

neighbour\_matrix4=mask4.\*In(i-1:i+1, j-1:j+1);

avg\_value4=sum(neighbour\_matrix4(:));

%using max function for detection of final edges

I(i, j)=max([avg\_value1, avg\_value2, avg\_value3, avg\_value4]);

end

end

% figure, imshow(uint8(I));

figure;

subplot(1,2,1)

imshow(uint8(I))

title('Canny Filter');

subplot(1,2,2)

imshow(img)

title('Original Image');

**Result:**

