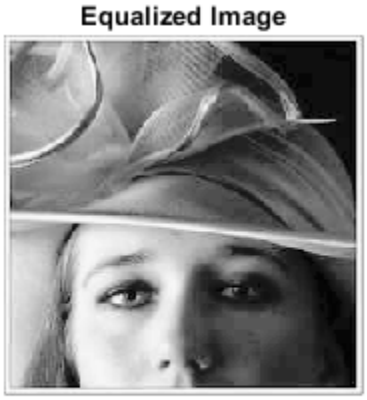
Classwork 6

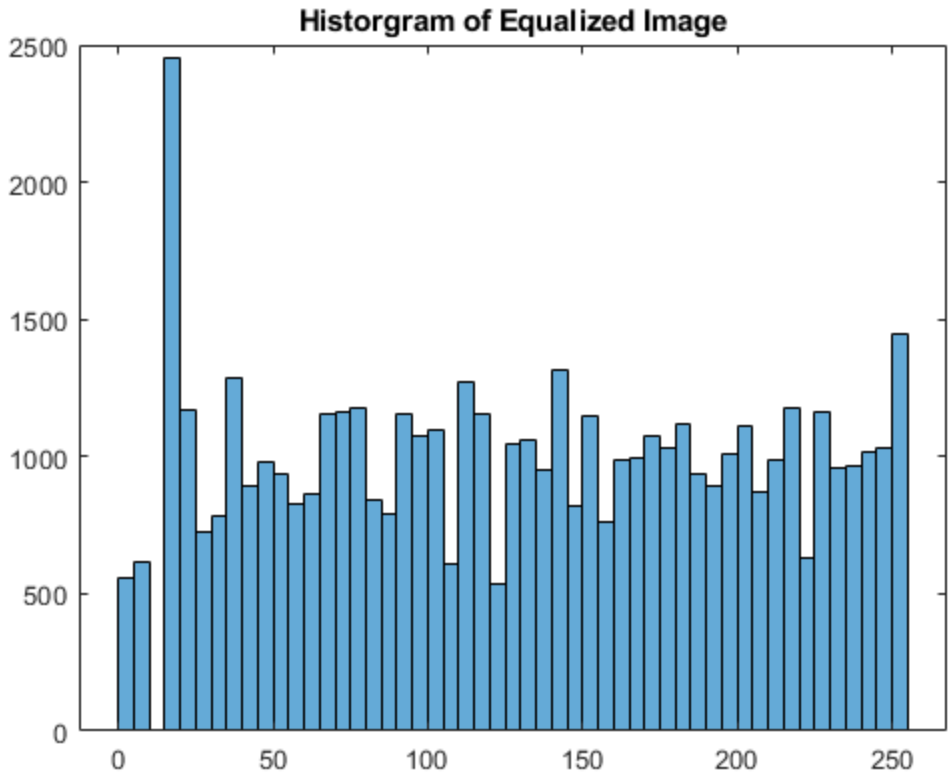
Table of Contents

Problem 1

my_histeq function

```
%Put into a function
A = imread('lowcontrast.jpg');
B = my_histeq(A);
% Test
figure(1)
imshow(B);
title('Equalized Image');
figure(2)
histogram(B);
title('Historgram of Equalized Image');
```





Problem 2

```
A = imread("lena.bmp");
A_gray = rgb2gray(A);   % has to be gray scale for edge detection
figure(3)
imshow(A_gray);
title('Gray Scale of Image')
```

Edge Detection - Roberts Operator

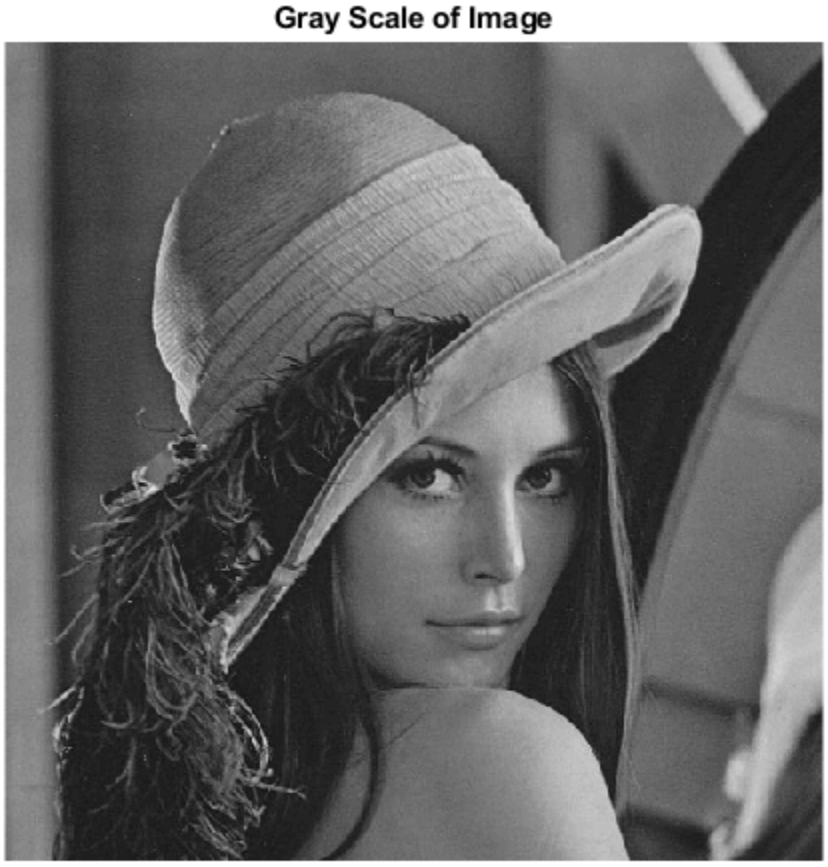
```
rob = edge(A_gray, 'roberts');
figure(4)
imshow(rob);
title('Edge Detection using Roberts Operator')
```

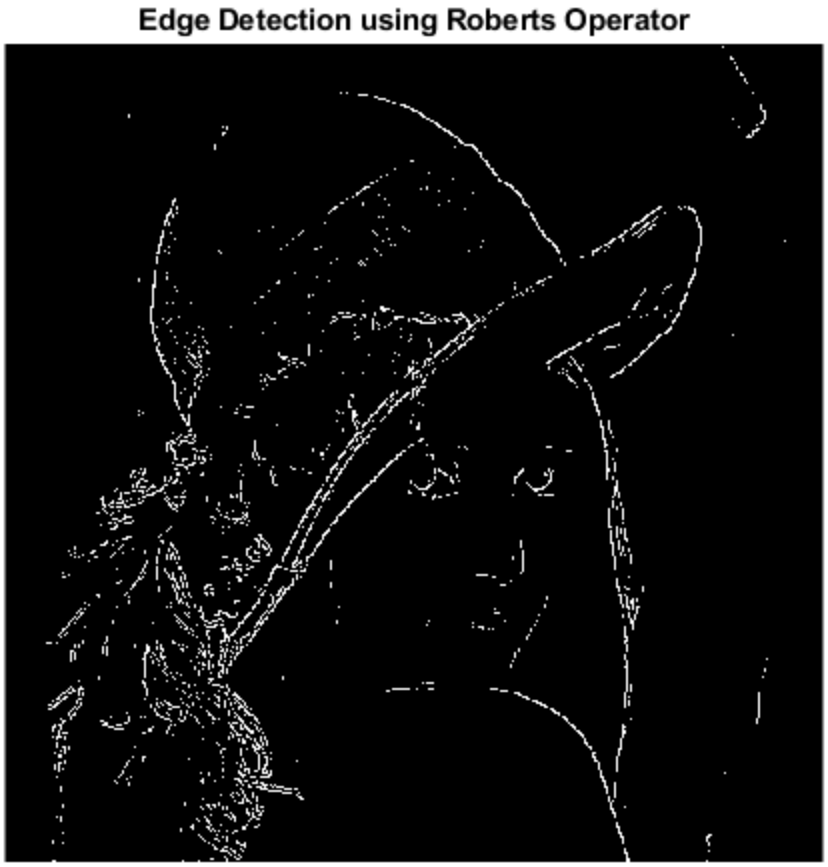
Edge Detection - Prewitt Operator

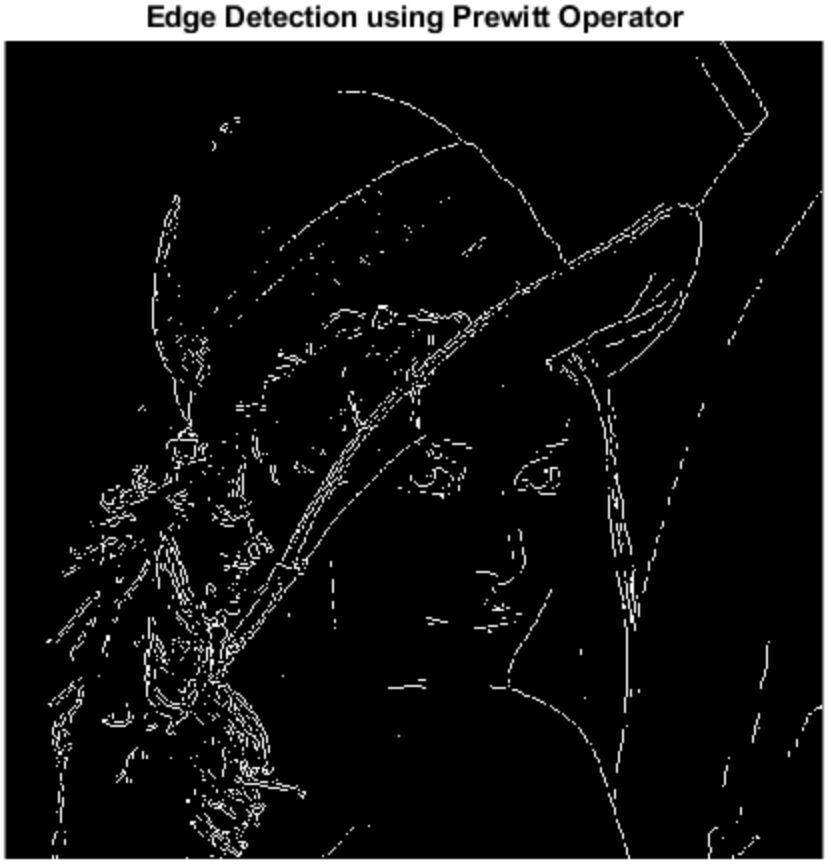
```
pre = edge(A_gray, 'prewitt');
figure(5)
imshow(pre);
title('Edge Detection using Prewitt Operator')
```

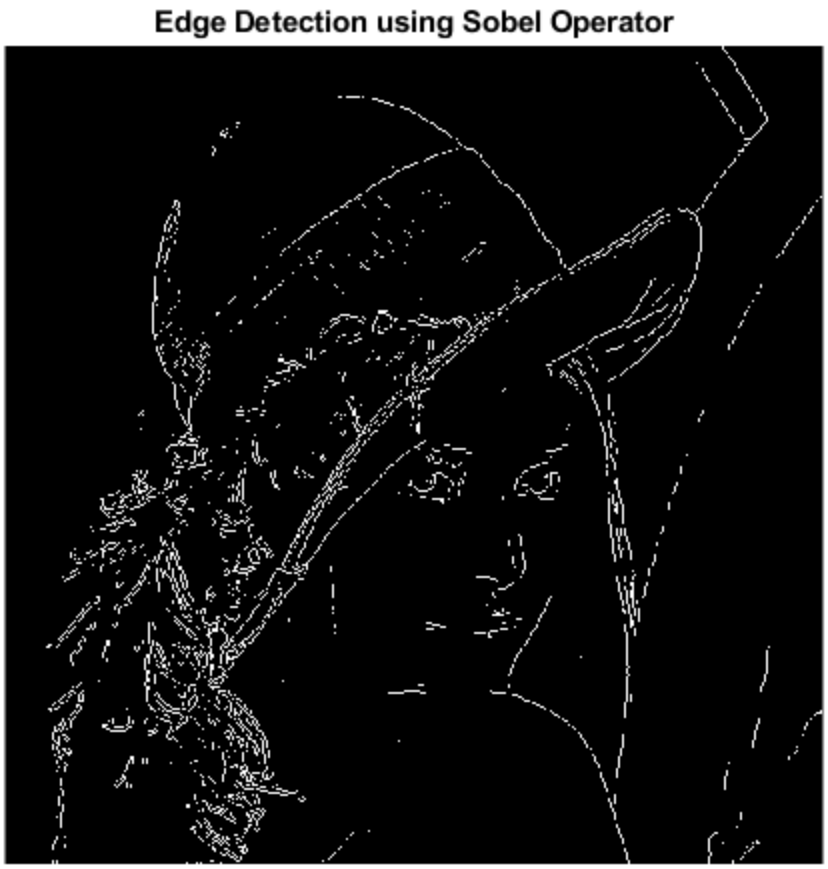
Edge Detection - Sobel Operator

```
sob = edge(A_gray, 'sobel');
figure(6)
imshow(sob);
title('Edge Detection using Sobel Operator')
```




Edge Detection using Roberts Operator


Edge Detection using Prewitt Operator



Problem 3

Laplacian Edge Sharpening

```
B = im2double(imread('blur_image.jpg'));
my_laplacianSharpening(B);
```

Histogram Equalization Funciton

```
function FinVal = my_histeq(inputImage)
    L = 256;    % intensity
    FreqPix = imhist(inputImage);   % histogram
    CumHist = cumsum(FreqPix);  % cum distribution function
    CDFPix = CumHist/numel(inputImage);     % normalize
    Valrep = L*CDFPix(double(inputImage) + 1);  % add +1 bc if its a 0 you'll
 get an error
    FinVal = uint8(Valrep);

end
```

Laplacian Edge Sharpening Funciton

```
function outputImage = my_laplacianSharpening(inputImage)
mask = [-1 -1 -1; -1 8 -1; -1 -1 -1];   % the usual standard for laplacian
 mask
% Use imfilter to perform convoultion of image and mask
ConvImage = imfilter(inputImage, mask);  %convoultion is the double derivative
 (2nd half of equation)
% Normalize convoluted image
```

```
minConv = min(ConvImage(:));
maxConv = max(ConvImage(:));
ConvImage = (ConvImage - minConv) / (maxConv - minConv);

% Add original image
SharpImage = inputImage + ConvImage;

%Normalize sharpened result
minSharp = min(SharpImage(:));
maxSharp = max(SharpImage(:));
outputImage = (SharpImage - minSharp) / (maxSharp - minSharp);

% Extra Part
% Contrast Enhancement
outputImage = imadjust(outputImage, [60/256 200/256], [0,1]);

figure(7)
subplot(1,3,1); imshow(inputImage); title('Original Image')
subplot(1,3,2); imshow(ConvImage); title('Laplacian Filtered Image')
subplot(1,3,3); imshow(outputImage); title('Sharpened Image')
end
```

Published with MATLAB® R2022b