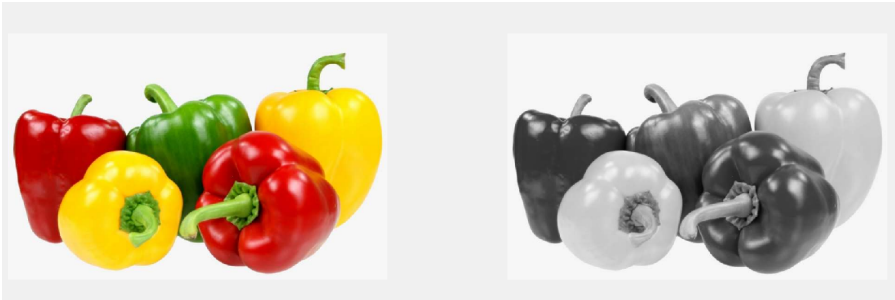


```
% Conversion of colour image to gray image
RGB = imread('peppers.jpg');
subplot(1,2,1);
imshow(RGB);
subplot(1,2,2);
x = rgb2gray(RGB);
imshow(x)
```

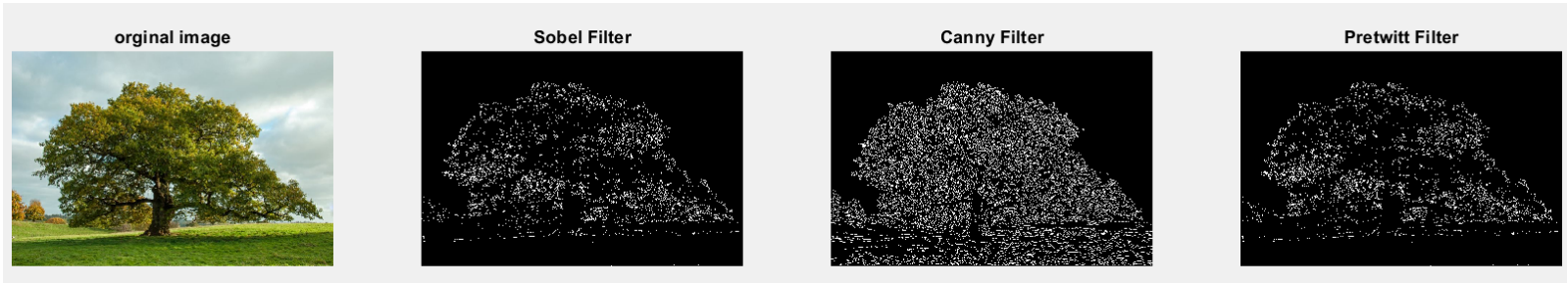


```
%Image enhancement
poutOriginal = imread('pout.jfif');
pout = rgb2gray(poutOriginal);
pout_imadjust = imadjust(pout);
pout_histeq = histeq(pout);
pout_adapthisteq = adapthisteq(pout);
montage({pout,pout_imadjust,pout_histeq,pout_adapthisteq},'Size',[1 4])
title('Original Image and Enhanced Images using imadjust, histeq, and adapthisteq')
```



```
% Program for Edge Detection
I = imread('trees.jpeg');
I1=rgb2gray(I);
BW1 = edge(I1,'sobel');
BW2 = edge(I1,'canny');
BW3= edge(I1,'Prewitt');
tiledlayout(1,4)
nexttile
imshow(I)
```

```
title('original image')
nexttile
imshow(BW1)
title('Sobel Filter')
nexttile
imshow(BW2)
title('Canny Filter')
nexttile
imshow(BW3)
title('Pretwitt Filter')
```



```

%image to histogram
originalImage = imread('coins.jpeg');
originalHistogram = imhist(originalImage);
equalizedImage = histeq(originalImage);
equalizedHistogram = imhist(equalizedImage);
subplot(2, 2, 1);
imshow(originalImage);
title('Original Image');
subplot(2, 2, 2);
bar(originalHistogram);
title('Original Histogram');
subplot(2, 2, 3);
imshow(equalizedImage);
title('Equalized Image');
subplot(2, 2, 4);
bar(equalizedHistogram);
title('Equalized Histogram');

```

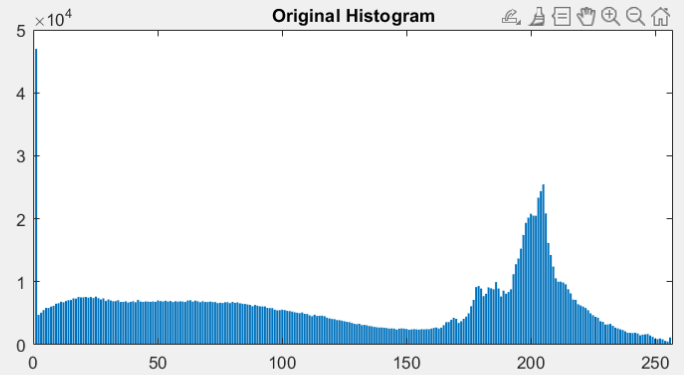
Original Image



Equalized Image



Original Histogram



Equalized Histogram

