

1 - Getting Started

Load the data that we will use.

```
I = imread('lab1/images/napoleon.png');  
Il = imread('lab1/images/napoleon_light.png');  
Id = imread('lab1/images/napoleon_dark.png');  
Z = imread('lab1/images/zebra.png');
```

Q1 - Pixel Value

Print out value. We see that it is of type UINT-8 and the value is 89.

```
I(1, 1)           % It prints out 89
```

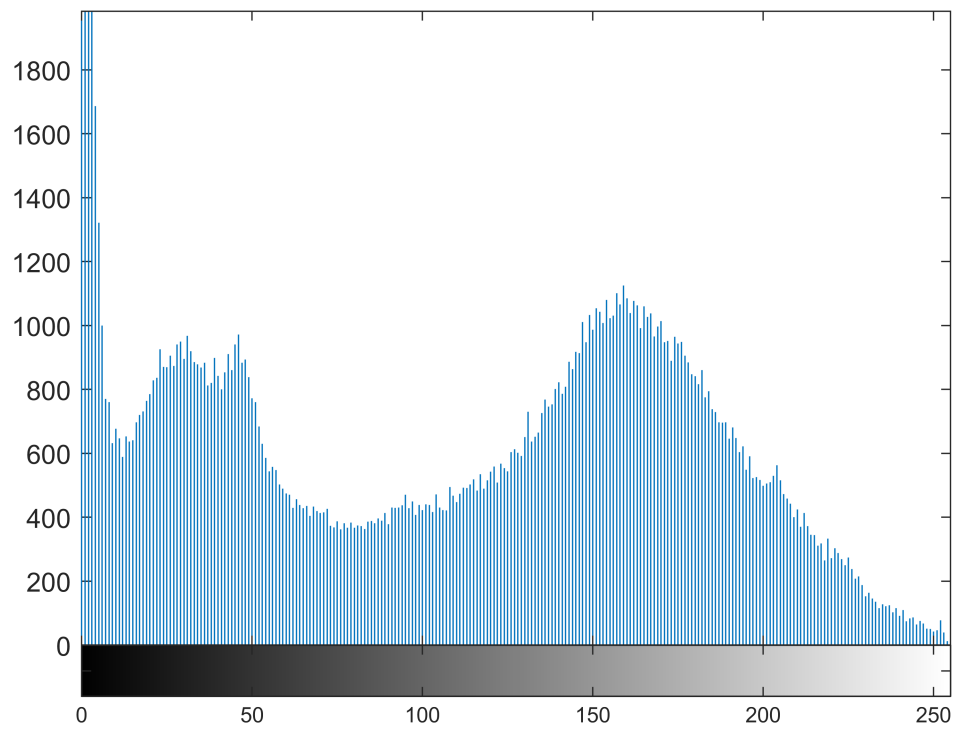
```
ans = uint8
```

```
89
```

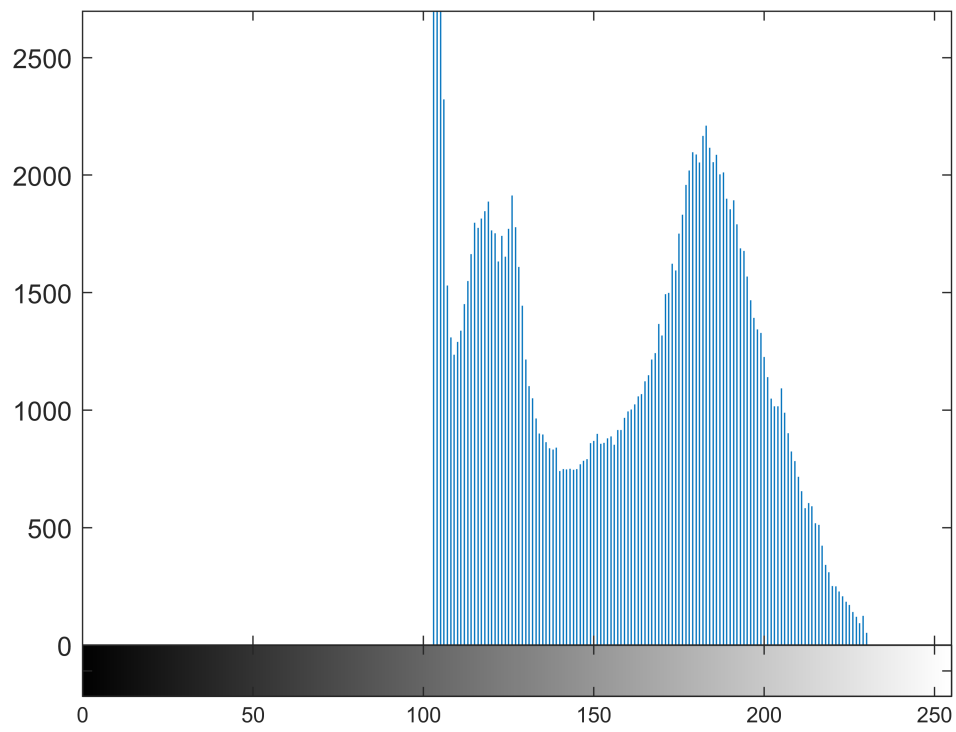
Q2 - Histograms

From figure 1, which is the regular image, we see that the pixel values are spread across the domain 0, 255. The second figure is skewed towards the lighter pixel values and this image should be the high contrast one. The third histogram shows the third image and how it is skewed in the other direction and is thus the histogram of the low contrast image.

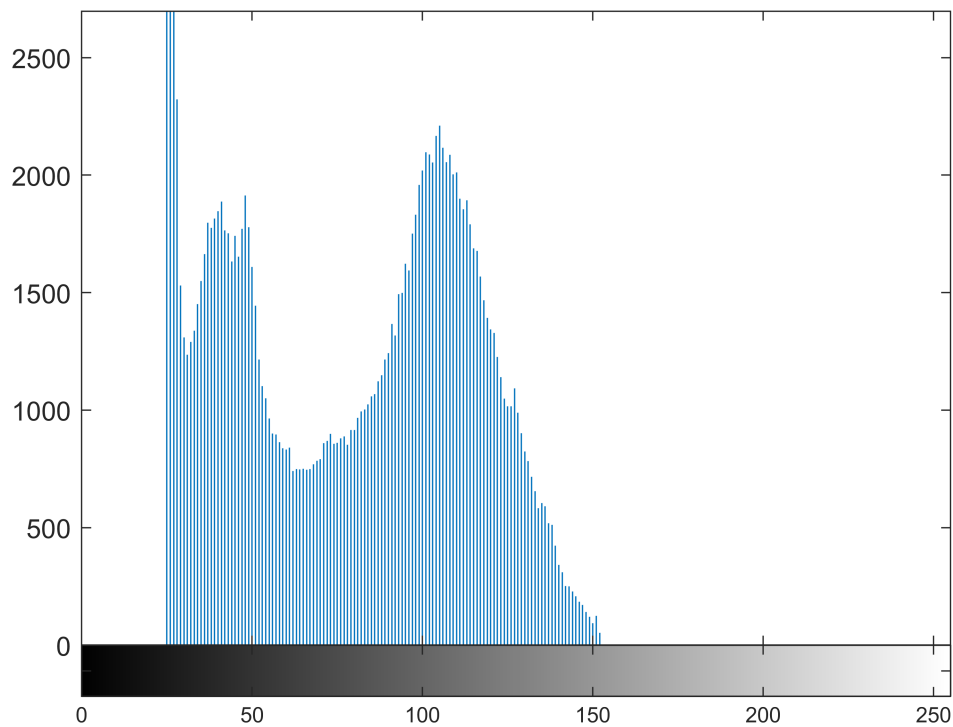
```
figure(1);  
imhist(I);
```



```
figure(2);  
imhist(I1);
```



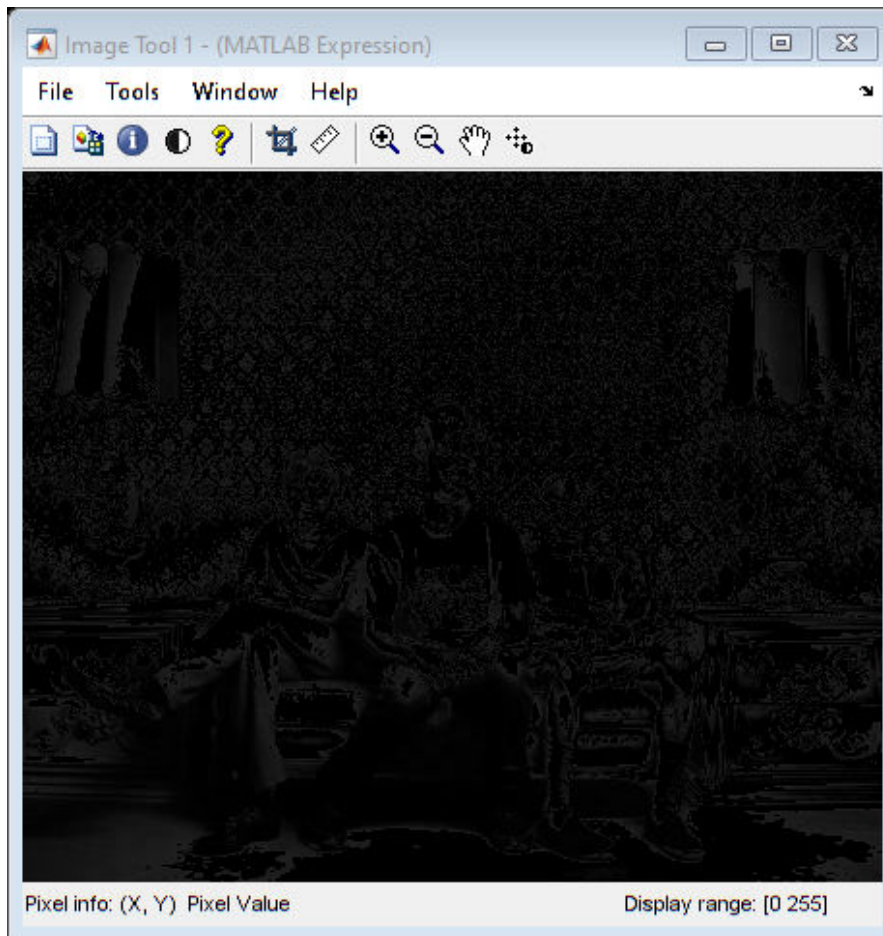
```
figure(3);  
imhist(Id);
```



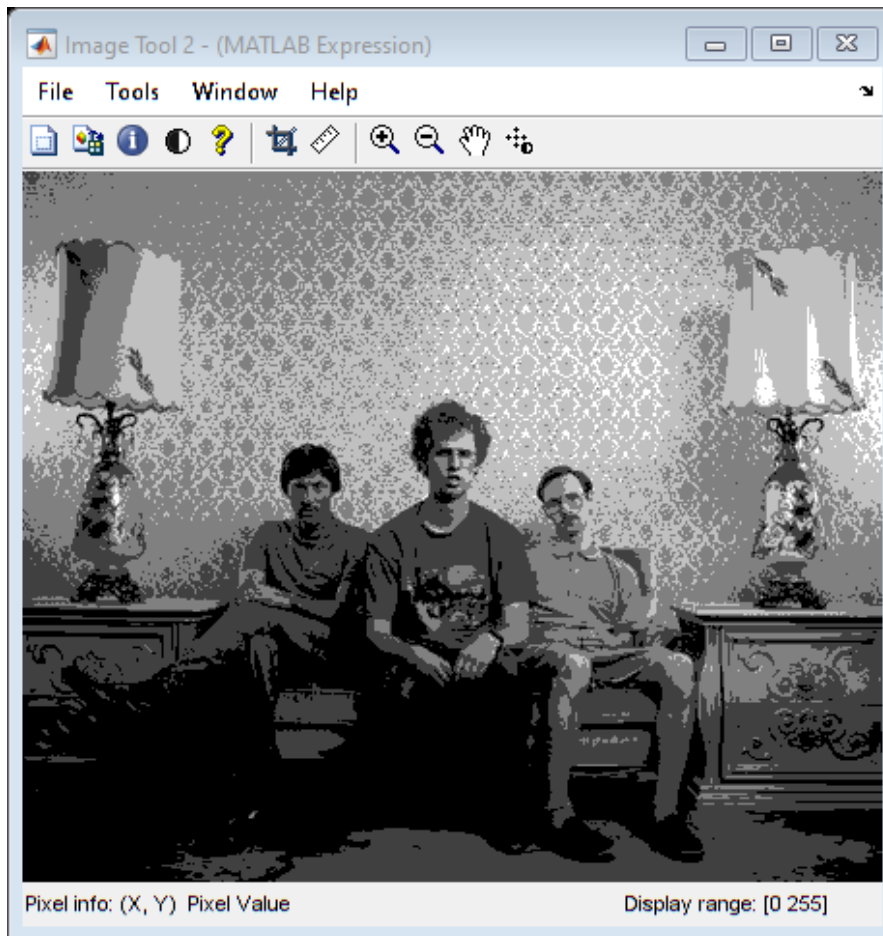
Q3 - Explain differences

The problem here is that when `imtool` gets passed a uint image it is able to adjust the visualisation range between 0 and 255 but when it gets passed a single precision matrix, which has values from 0 to 255, it tries to plot it as a image with values between 0 and 1. Therefore it needs to be normalised to be plotted correctly. Below we have plotted the difference between the original image and the examples from the assignment. We see that in the UINT8 one there is considerable difference between the two and this is related to doingh arithmetic on UINT8 variables.

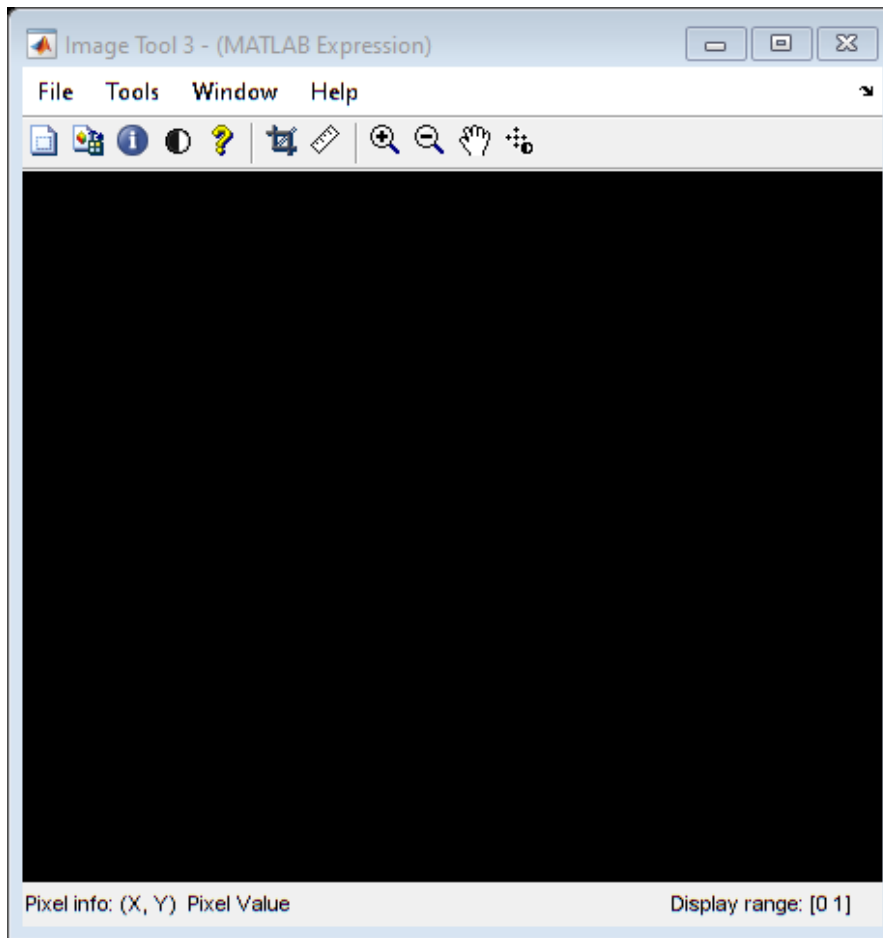
```
imtool(I - (I/64)*64);
```



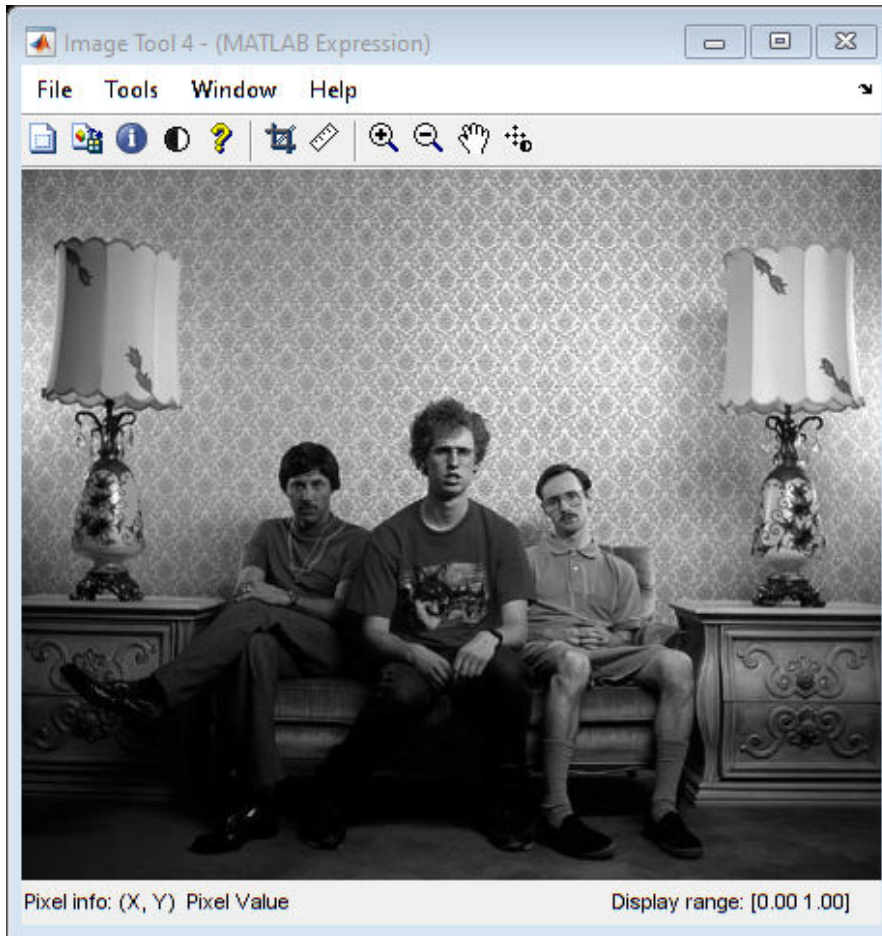
```
imtool((I/64)*64);
```



```
imtool((Is/255) - (Is/64)*64/255);
```



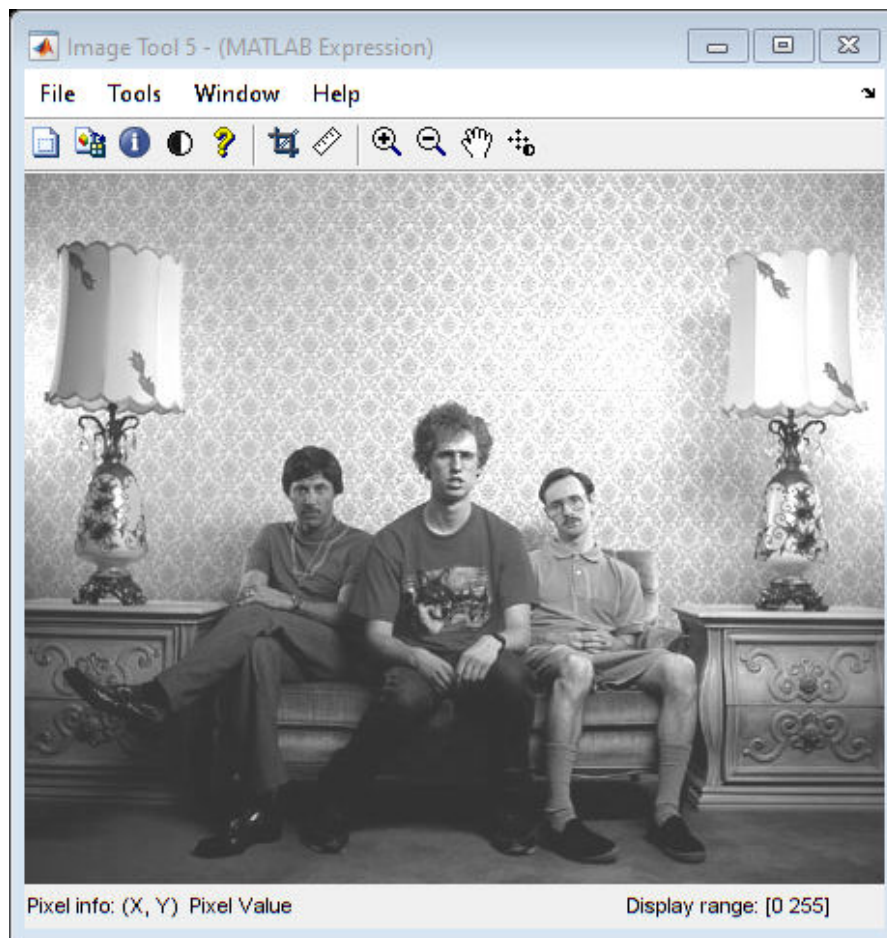
```
imtool((Is/64)*64/255);
```



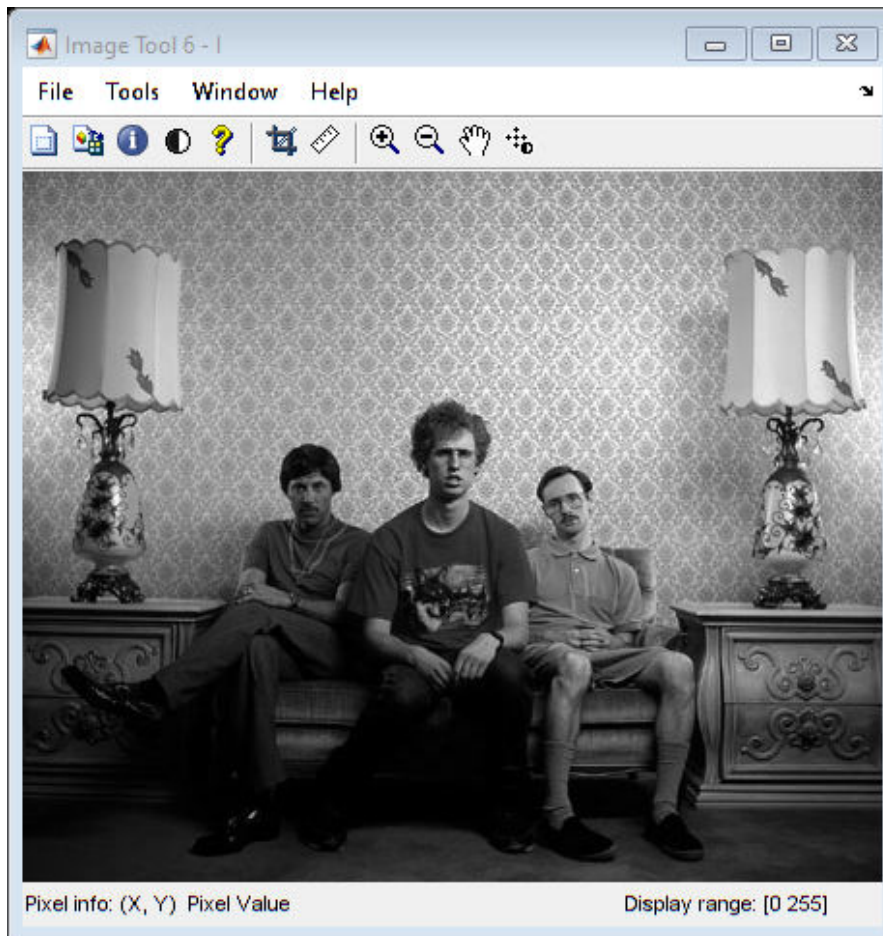
Q4 - Make images brighter

To make images brighter. Take the original image $f(x, y)$ and apply the transform $T = f(x, y) + C$ where $C > 1$.

```
imtool(I + 50);
```

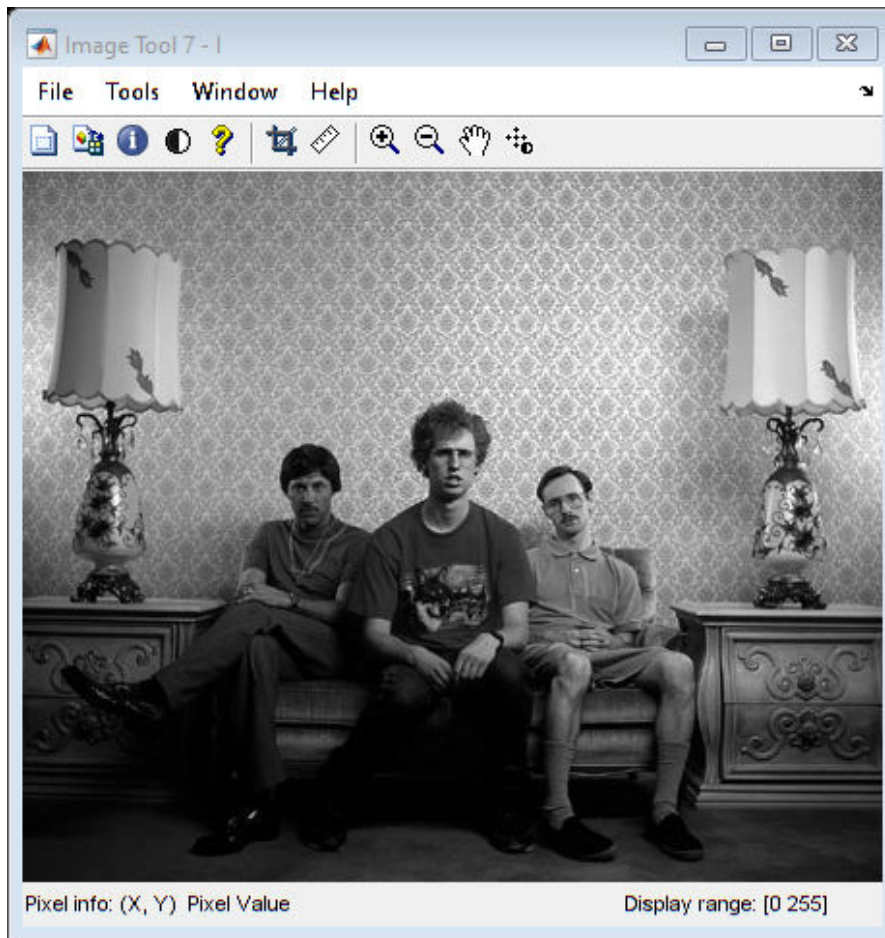
```
imtool(I);
```



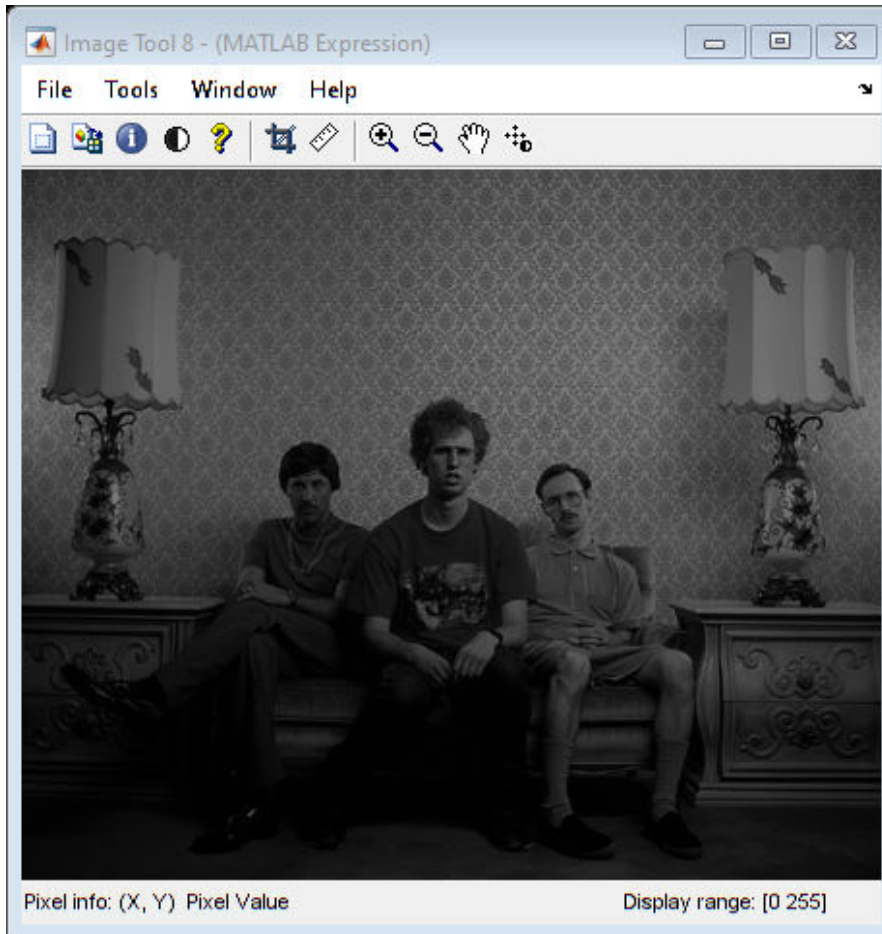
Q5 - Make images lower contrast

To make the image lower contrast we apply the transform $T = f(x, y) \times C$ where $C < 1$

```
imtool(I);
```



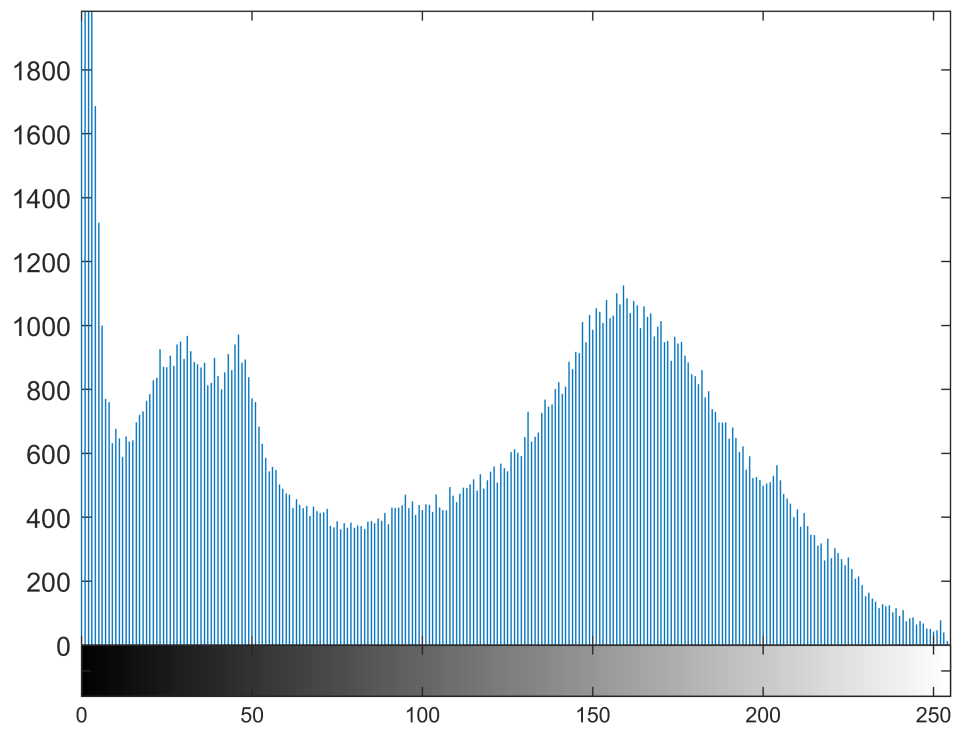
```
imtool(I * 0.5);
```



Q6 - Pixel Wise Transforms

Below we have performed a gamma transform which is $T = C \times f(x, y)^\gamma$ where $C = 1$ and $\gamma \in 0.5, 2$. We also compare this with the original image.

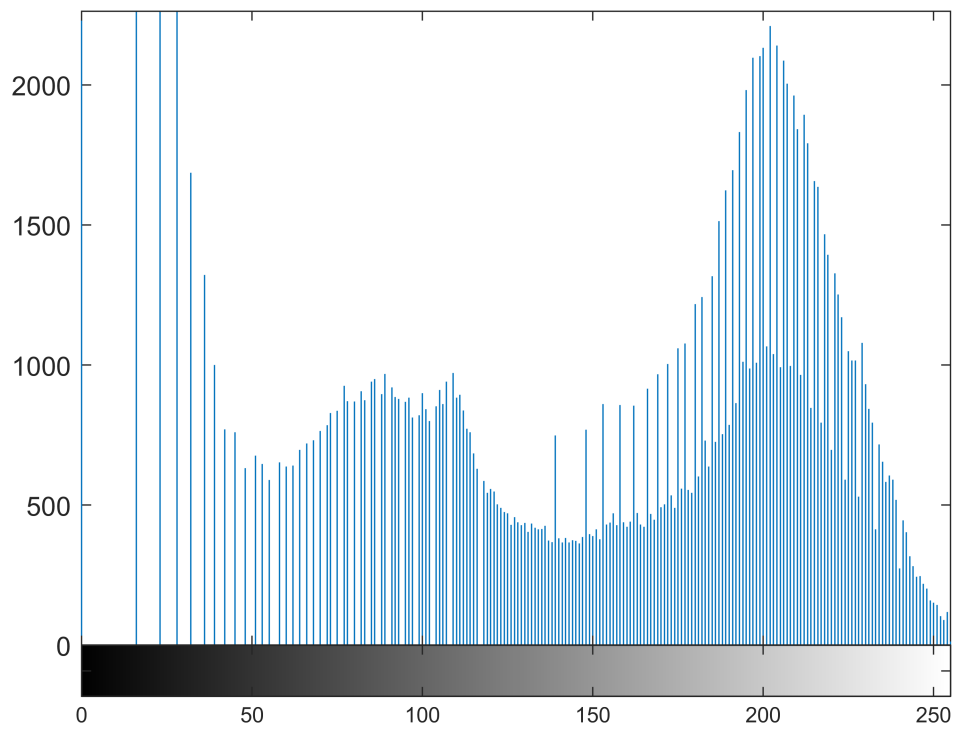
```
figure(1);  
imhist(I);
```



```
figure(2);  
imshow(I);
```



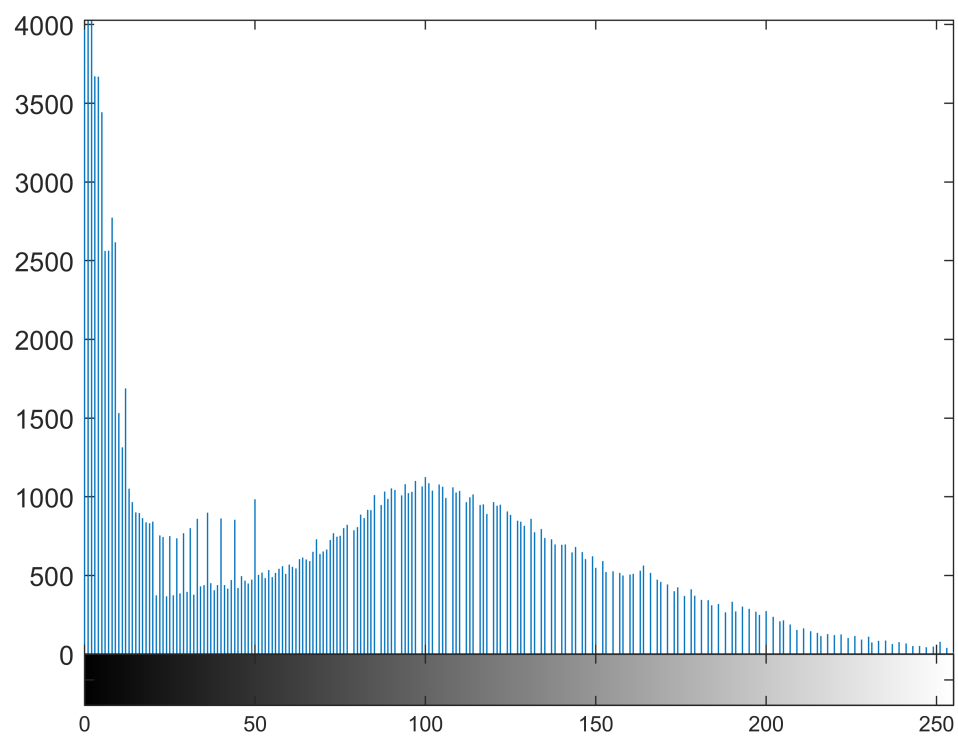
```
g = 0.5;  
L = double(I).^g;  
out = uint8(L .* (255/max(max(L))));  
  
figure(3);  
imhist(out);
```



```
figure(4);  
imshow(out);
```



```
g = 2;  
L = double(I).^g;  
out = uint8(L .* (255/max(max(L))));  
  
figure(5)  
imhist(out);
```

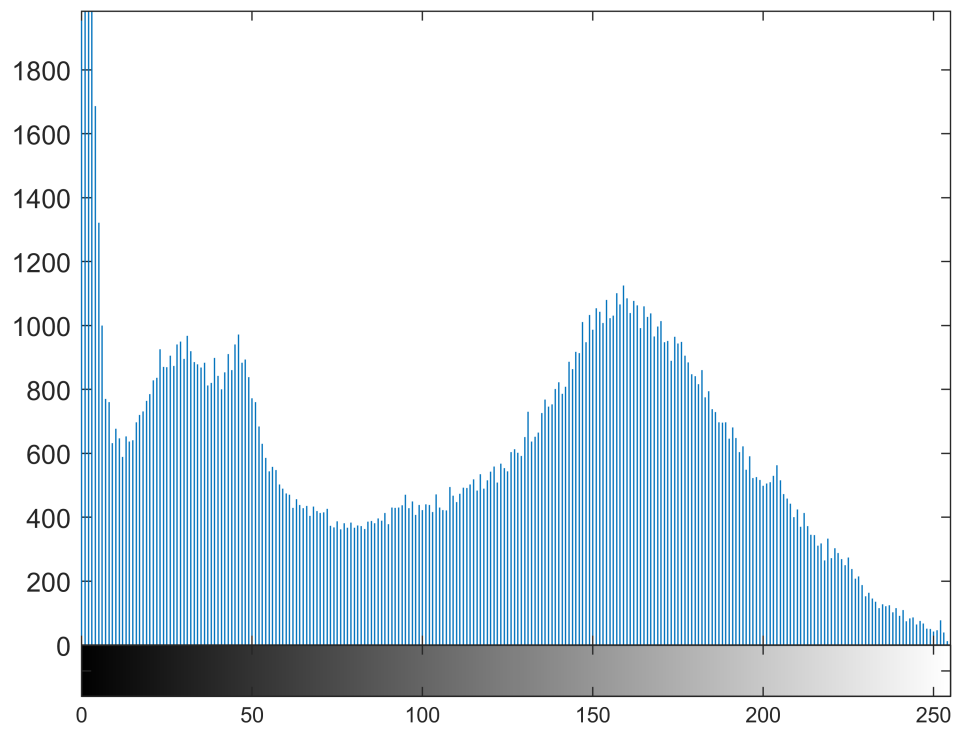
```
figure(6);  
imshow(out);
```



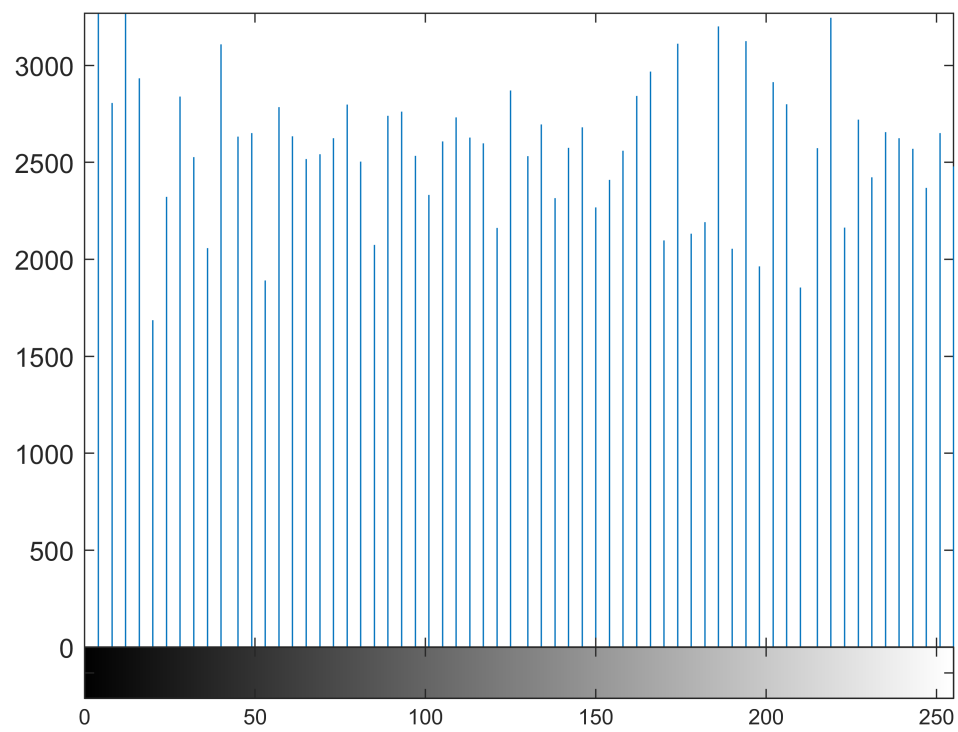
Q7 - Histogram Equalization napoleon - Histogram

Below is the histograms of the original image, and the original, light and dark image and their equalized histograms.

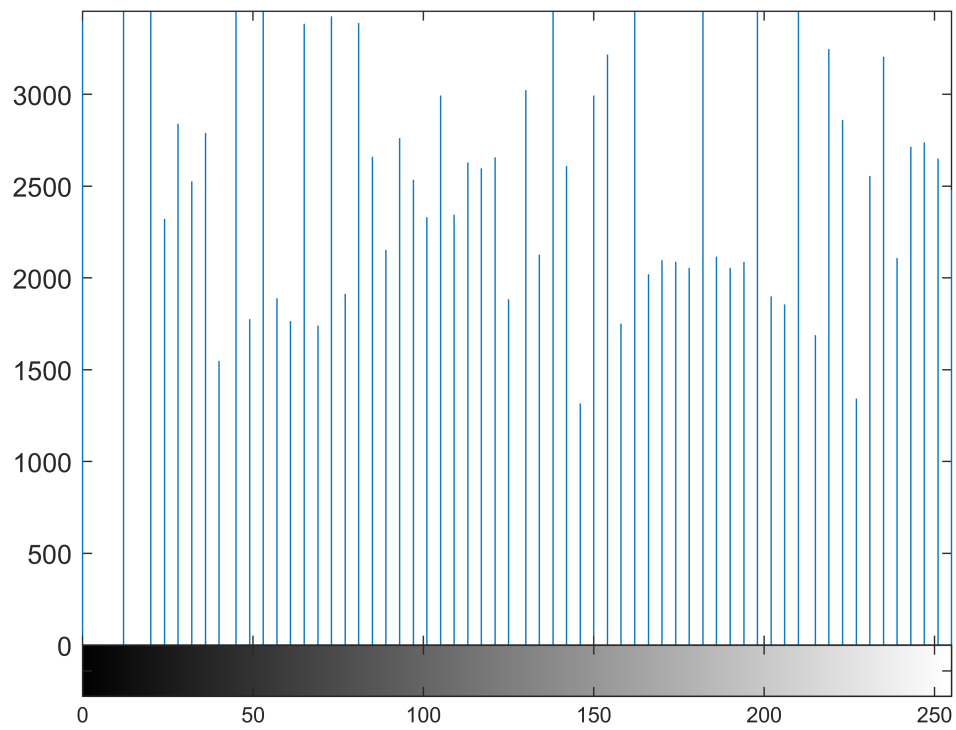
```
figure(1);  
imhist(I);
```



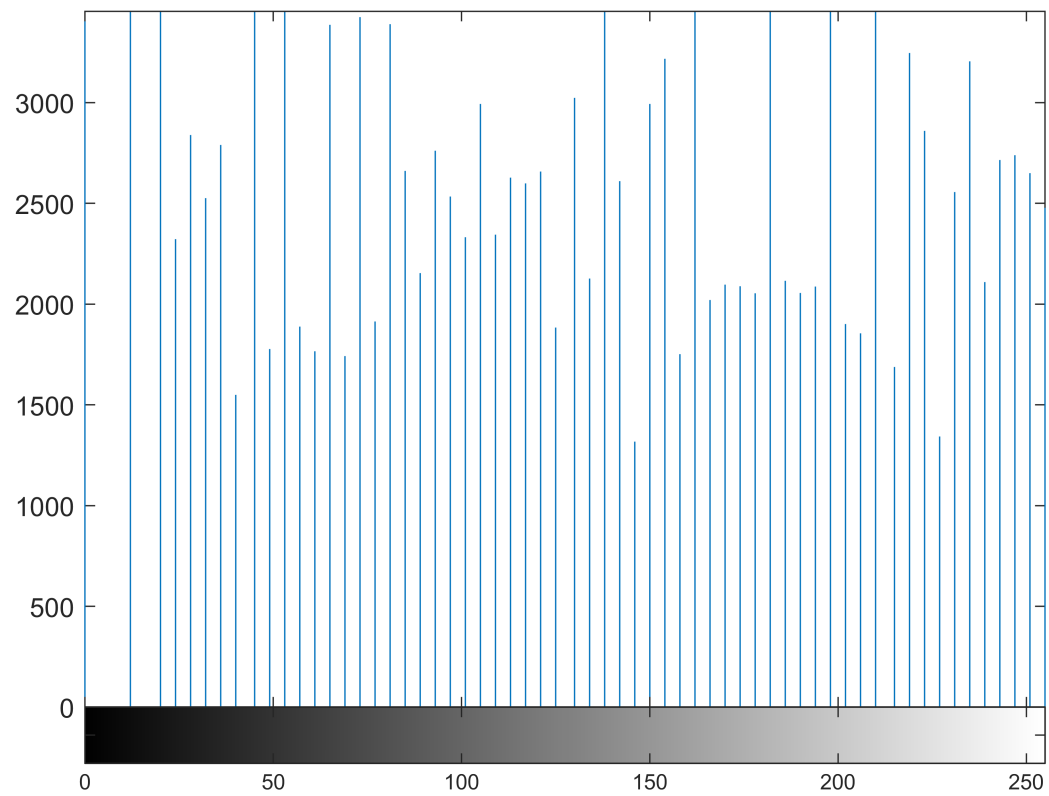
```
figure(2);  
imhist(histeq(I));
```



```
figure(3);  
imhist(histeq(I1));
```



```
figure(4);  
imhist(histeq(Id));
```



Q7 - Histogram Equalization napoleon - Images

Here we show the equalised images.

```
figure(1);  
imshow(I);
```



```
figure(2);  
imshow(histeq(I));
```



```
figure(3);  
imshow(histeq(I1));
```



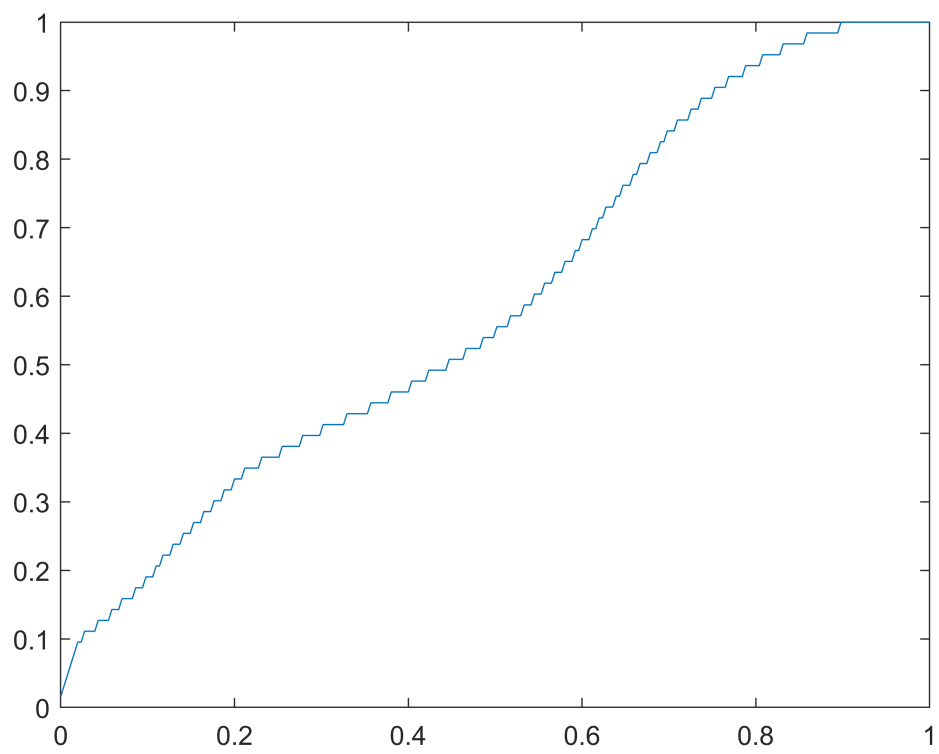

```
figure(4);  
imshow(histeq(Id));
```



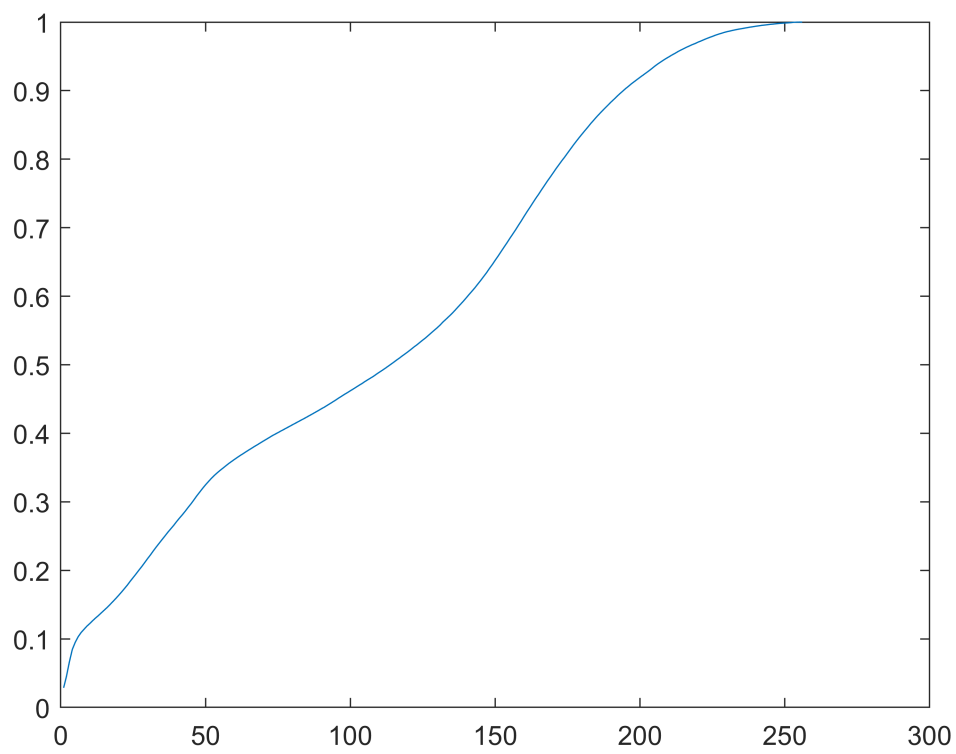
Q7 - Histogram, transform and cumulative histogram - Regular Image

And here is the cumulative histograms. According to the textbook, to equalize a image, you have to take the CDF of the image and use that to transform the pixel values. This is estimated by calculating the cumulative histogram and using that as an approximation for the CDF. If we use the `histeq` function and compare it to the normalied histogram calculated manually, we see that these are the same.

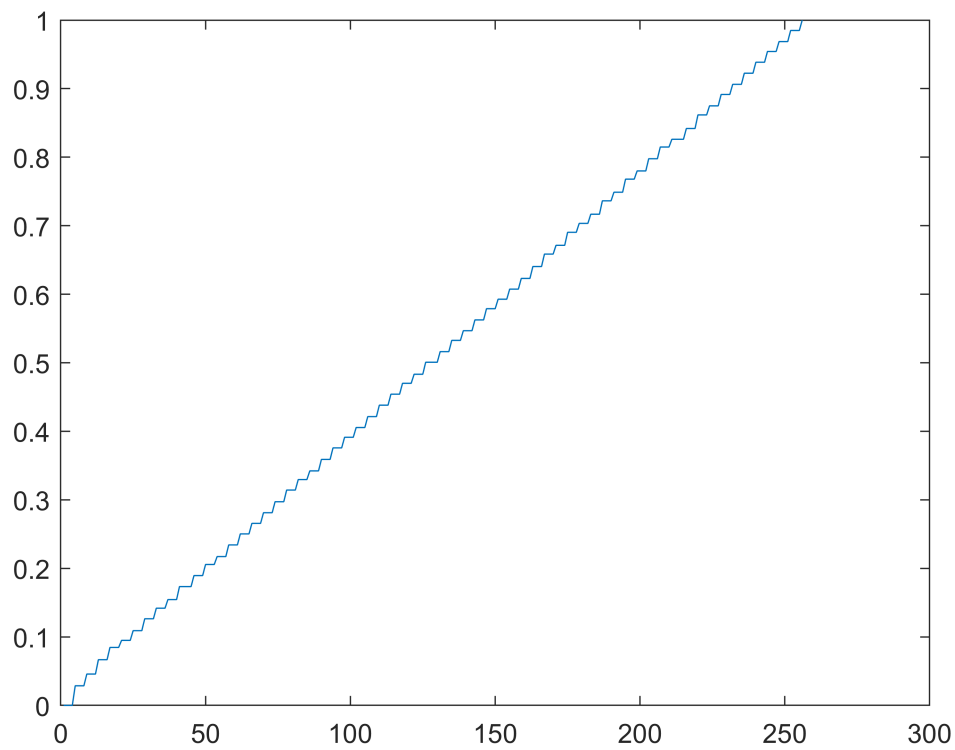
```
[J,T] = histeq(I);  
  
figure  
plot((0:255)/255,T);
```



```
figure
plot(cumsum(imhist(I)) / sum(imhist(I)));
```



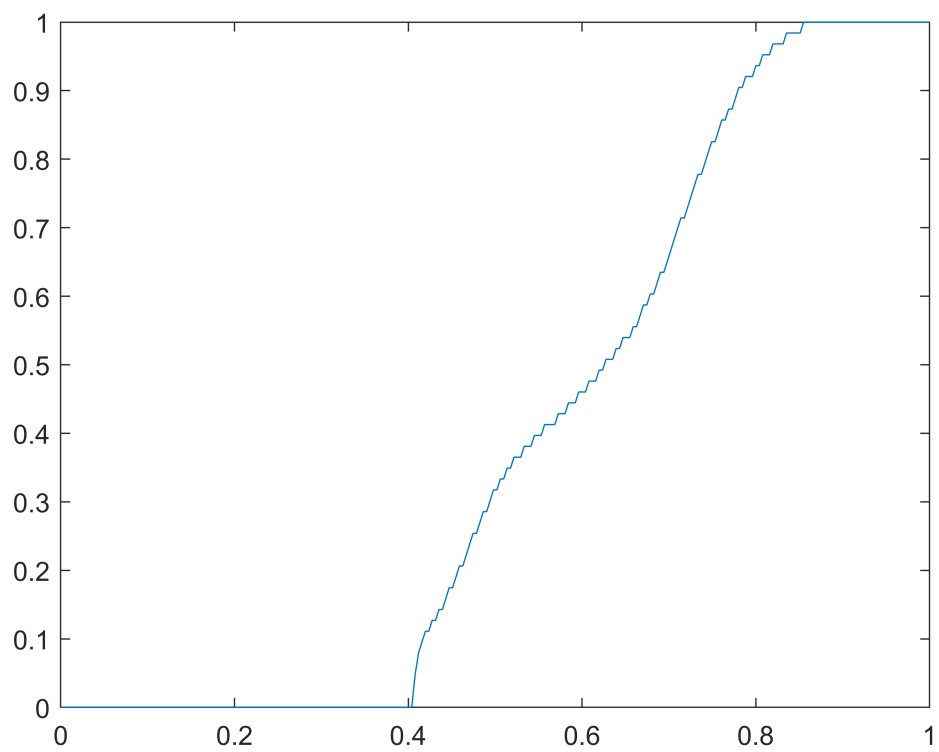
```
figure
plot(cumsum(imhist(J)) / sum(imhist(J)));
```



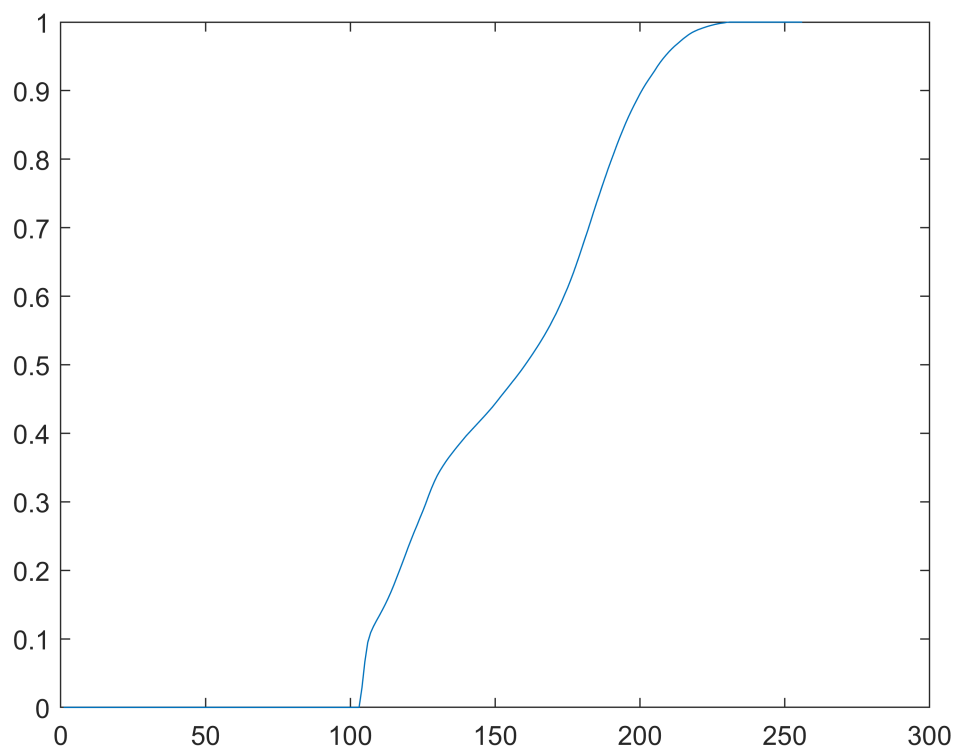
Q7 - Histogram, transform and cumulative histogram - Light

Same as above but for the light image.

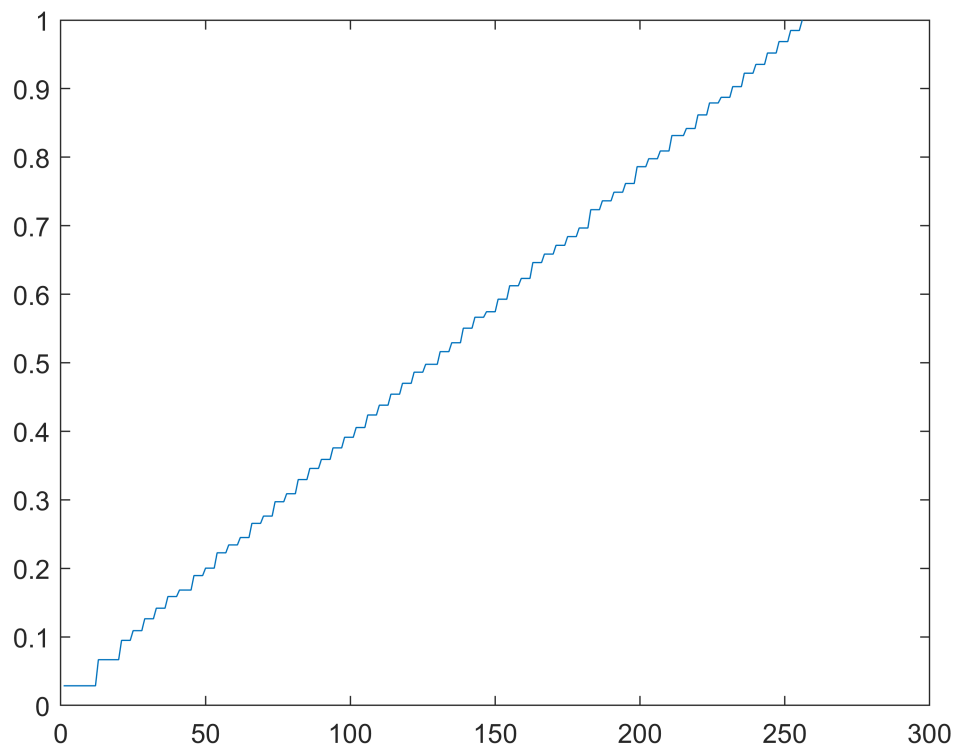
```
[J,T] = histeq(I1);  
  
figure  
plot((0:255)/255,T);
```



```
figure  
plot(cumsum(imhist(I1)) / sum(imhist(I1)));
```



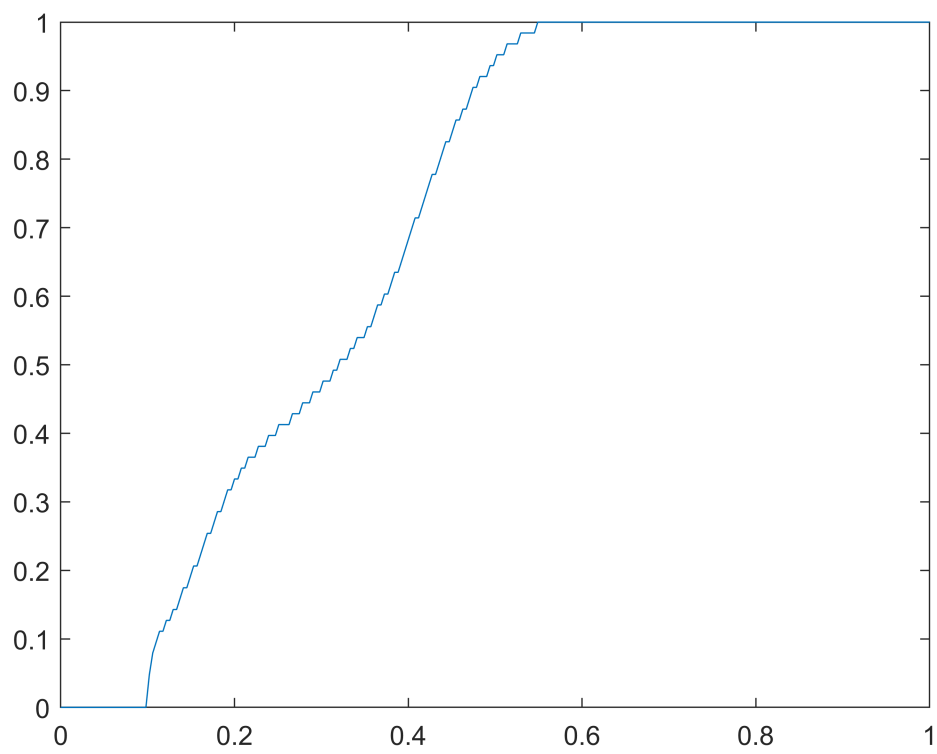
```
figure
plot(cumsum(imhist(J)) / sum(imhist(J)));
```



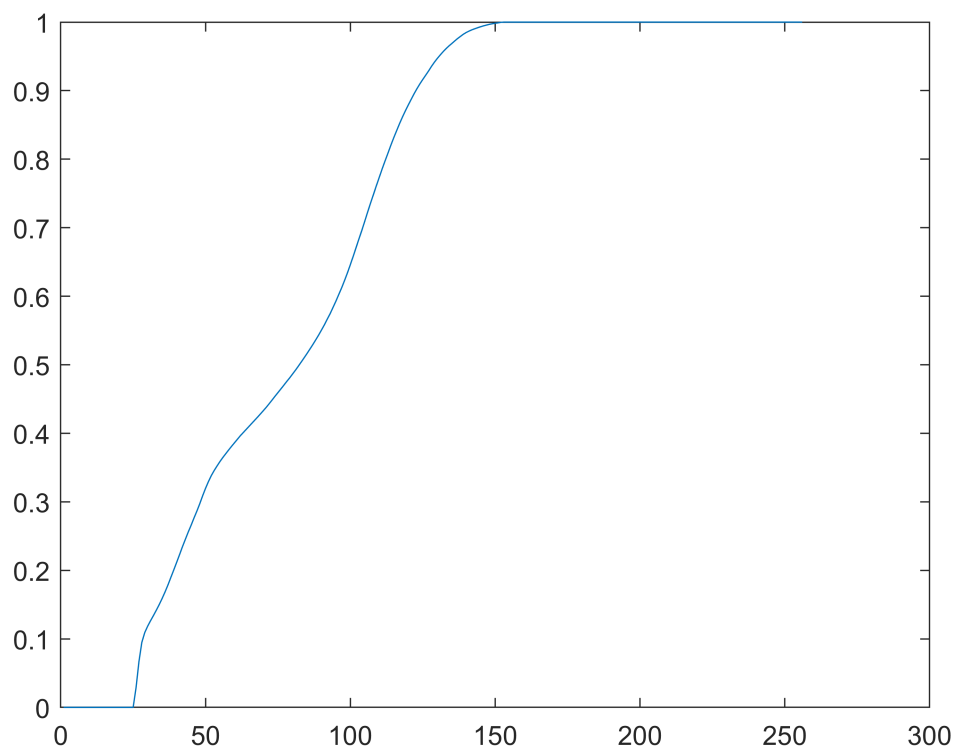
Q7 - Histogram, transform and cumulative histogram - Dark

Same as above but for the dark image.

```
[J,T] = histeq(Id);  
  
figure  
plot((0:255)/255,T);
```

```
figure
plot(cumsum(imhist(Id)) / sum(imhist(Id)));
```



```
figure
plot(cumsum(imhist(J)) / sum(imhist(J)));
```

