EXPERIMENT 4

Aim : To study Histogram Equalization and Histogram Matching.

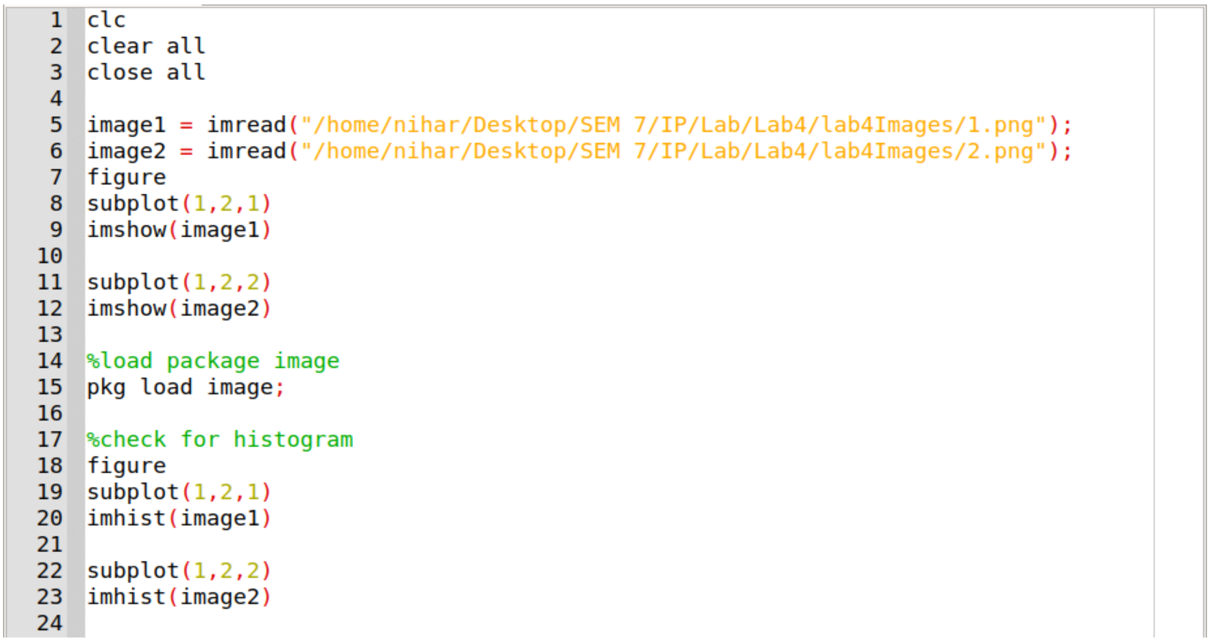
❖ Exercises :

1. **Can two visually different image have same histogram? If yes, synthesize two grayscale images which are visually different but having same histogram and also show the histogram. If no, justify your answer.**

→ **Solution :-**

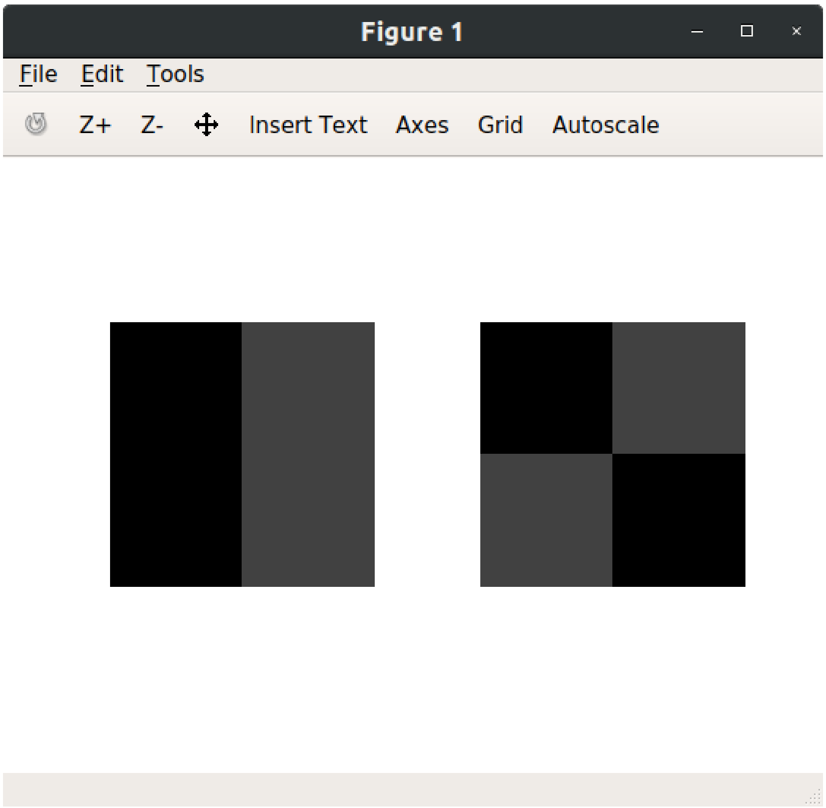
- Histogram is not a unique representation of an image.
- Histogram is a graph of gray value vs frequency of occurrence of gray value. It depends on the probability or frequency of gray value.
- So ,no matter how the gray values are distributed over the image, if the frequency of occurrence of gray value is not changed, the histogram will not change .Therefore, Histogram is not unique representation of images.
- That means it is possible that two or more different images can have same Histogram.

Code :

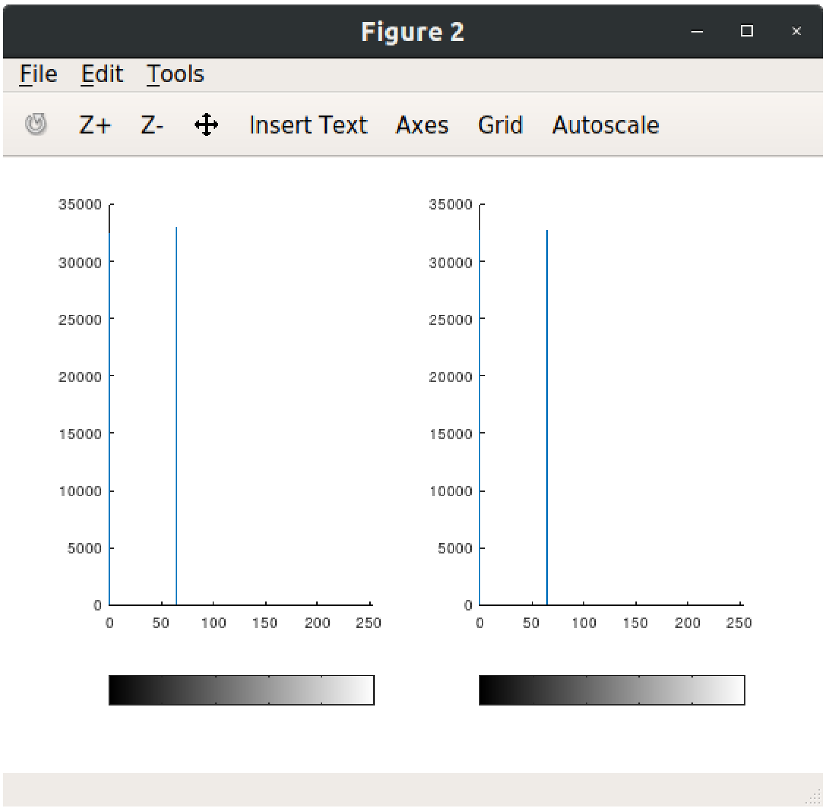


```
clc
clear all
close all

image1 = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/1.png");
image2 = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/2.png");
figure
subplot(1,2,1)
imshow(image1)

subplot(1,2,2)
imshow(image2)

%load package image
pkg load image;

%check for histogram
figure
subplot(1,2,1)
imhist(image1)

subplot(1,2,2)
imhist(image2)

```

Input Image :



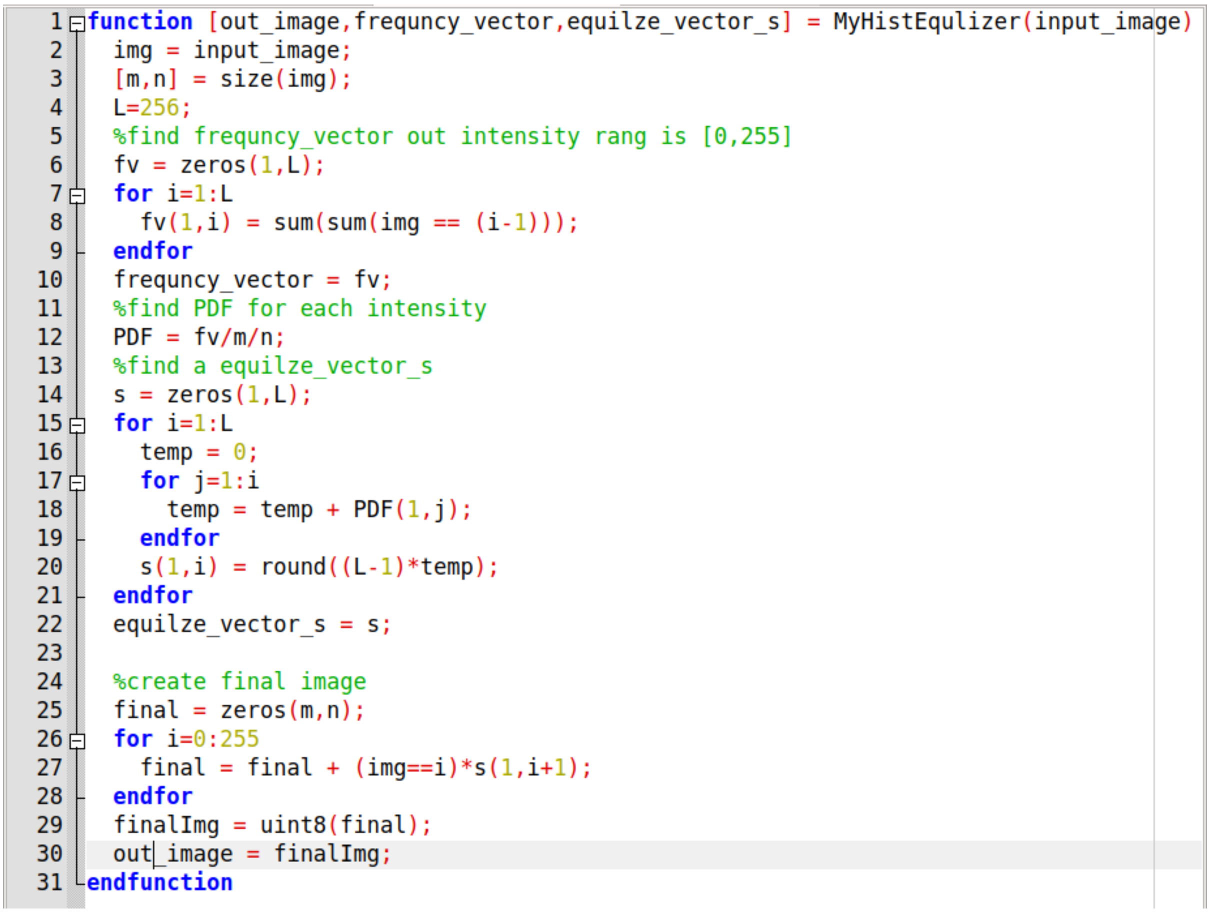
Output Image :



2. Take your color photograph taken in dark. Equalize it’s histogram.

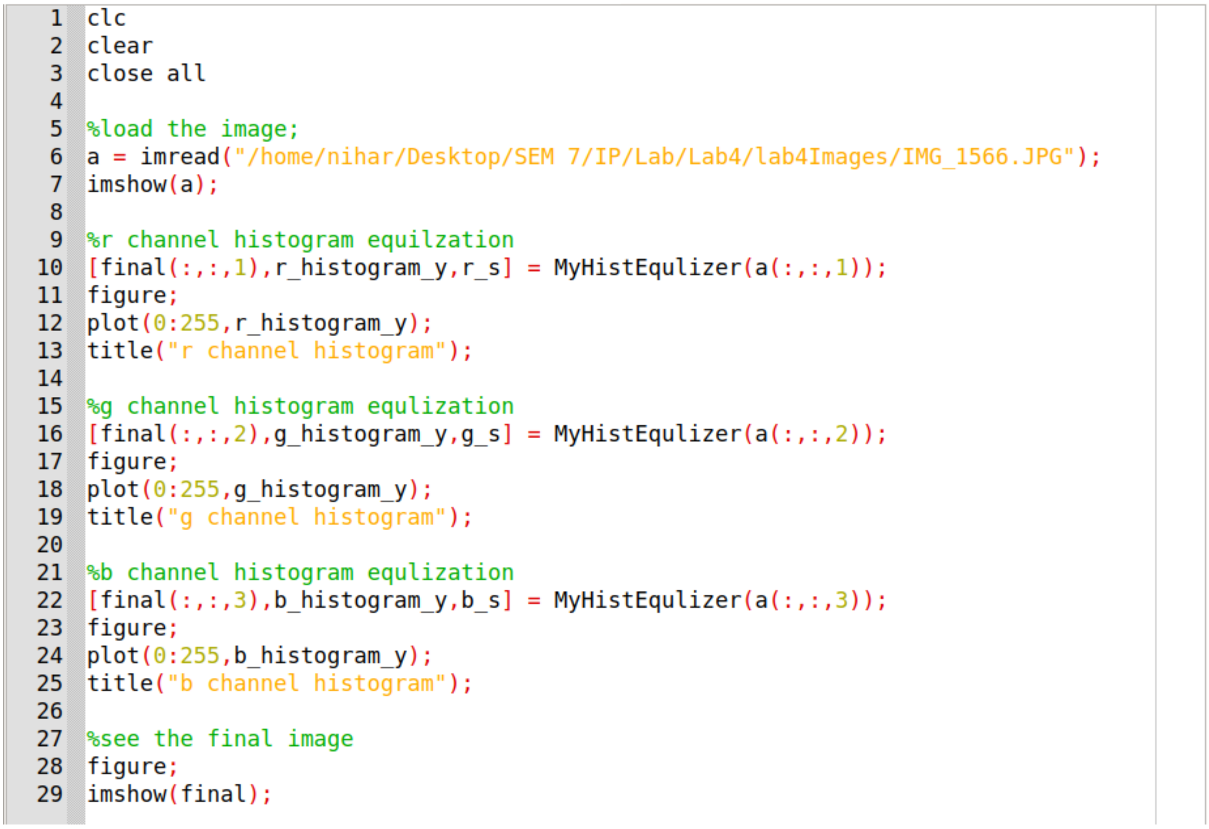
→ **Solution :-**

MyHistEqualizer Function :



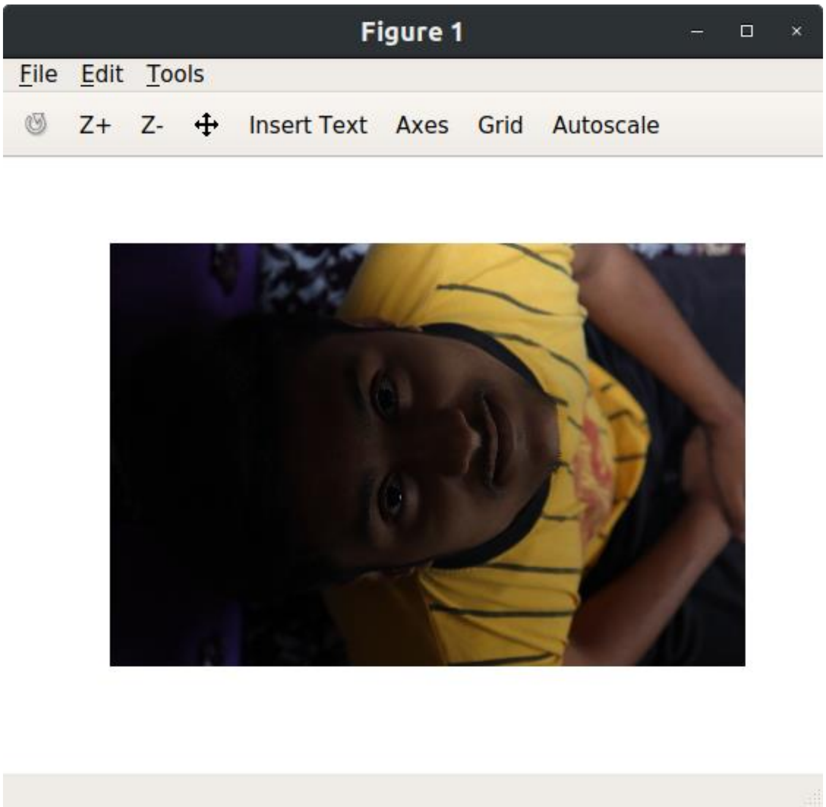
```
function [out_image,frequncy_vector,equilze_vector_s] = MyHistEqulizer(input_image)
  img = input_image;
  [m,n] = size(img);
  L=256;
  %find frequncy_vector out intensity rang is [0,255]
  fv = zeros(1,L);
  for i=1:L
    fv(1,i) = sum(sum(img == (i-1)));
  endfor
  frequncy_vector = fv;
  %find PDF for each intensity
  PDF = fv/m/n;
  %find a equilze_vector_s
  s = zeros(1,L);
  for i=1:L
    temp = 0;
    for j=1:i
      temp = temp + PDF(1,j);
    endfor
    s(1,i) = round((L-1)*temp);
  endfor
  equilze_vector_s = s;

  %create final image
  final = zeros(m,n);
  for i=0:255
    final = final + (img==i)*s(1,i+1);
  endfor
  finalImg = uint8(final);
  out_image = finalImg;
endfunction
```

Code :

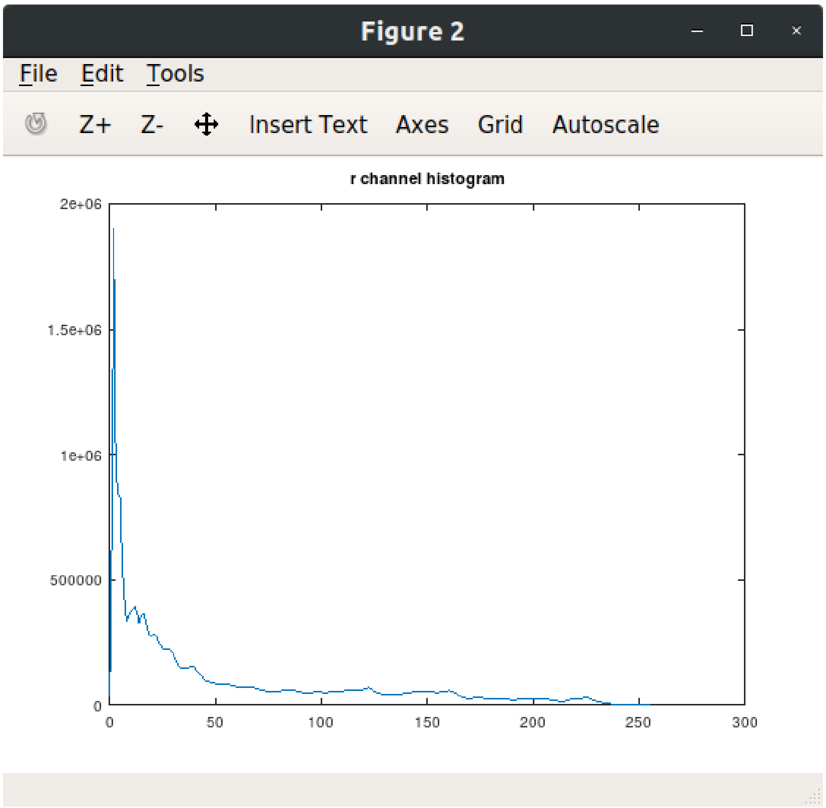


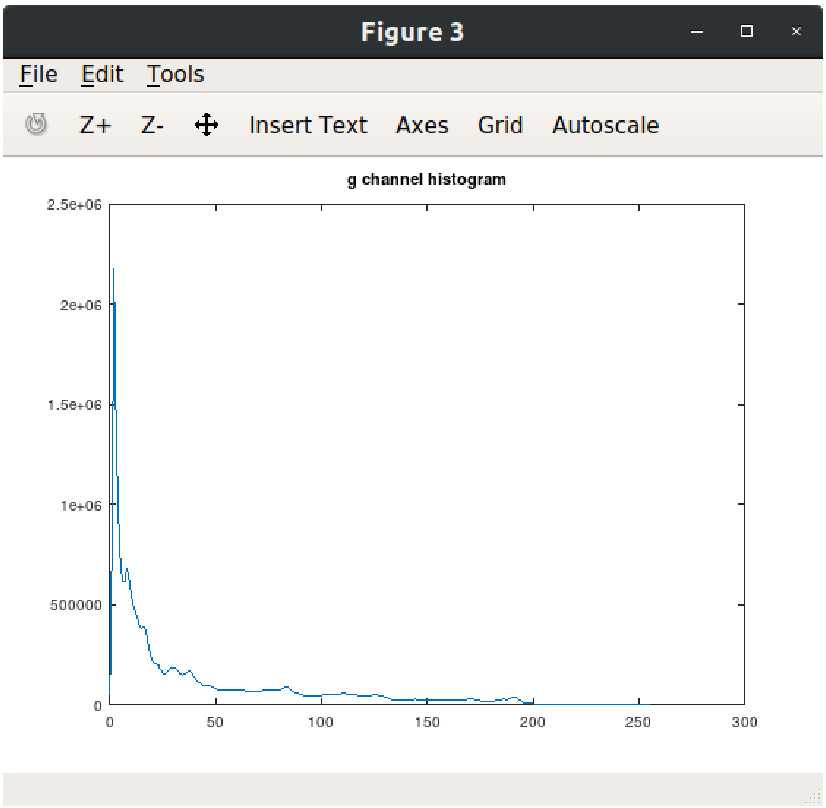
```
clc
clear
close all

%load the image;
a = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/IMG_1566.JPG");
imshow(a);

%r channel histogram equilzation
[final(:,:,1),r_histogram_y,r_s] = MyHistEqulizer(a(:,:,1));
figure;
plot(0:255,r_histogram_y);
title("r channel histogram");

%g channel histogram equilzation
[final(:,:,2),g_histogram_y,g_s] = MyHistEqulizer(a(:,:,2));
figure;
plot(0:255,g_histogram_y);
title("g channel histogram");

%b channel histogram equilzation
[final(:,:,3),b_histogram_y,b_s] = MyHistEqulizer(a(:,:,3));
figure;
plot(0:255,b_histogram_y);
title("b channel histogram");

%see the final image
figure;
imshow(final);
```

Input Image :



Output Images :





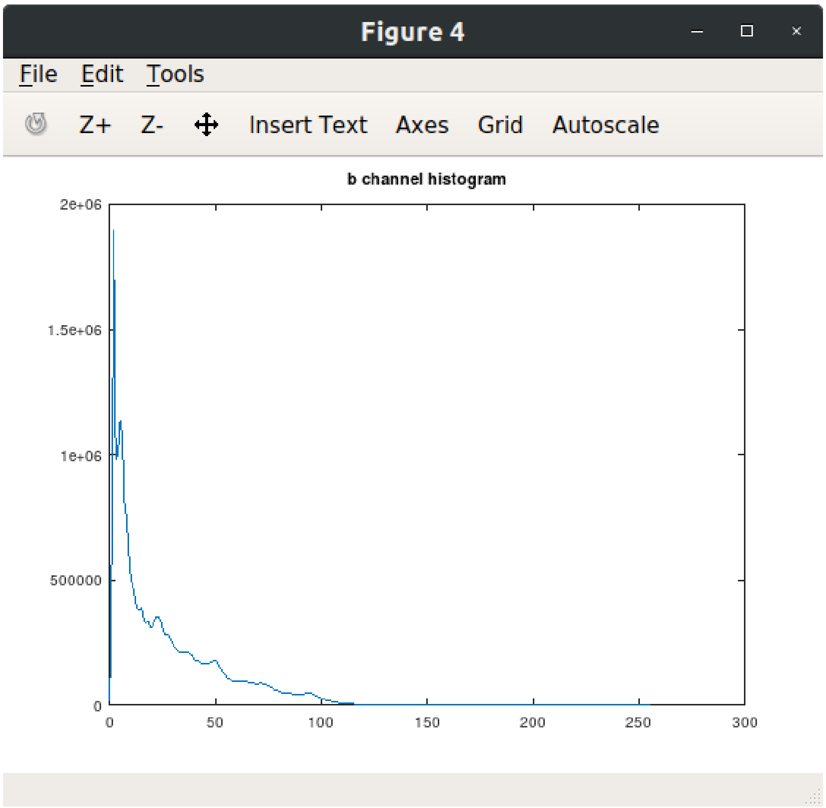


Figure 4
File
Edit
Tools
Z+
Z-
Insert Text
Axes
Grid
Autoscale
b channel histogram
2e+06
1.5e+06
1e+06
500000
0
0
50
100
150
200
250
300

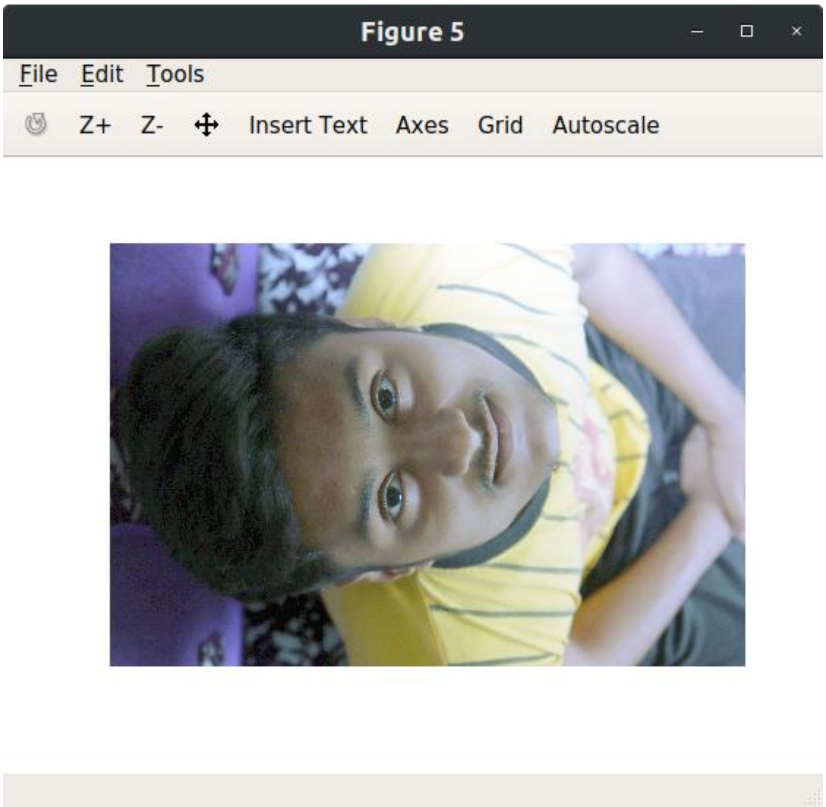


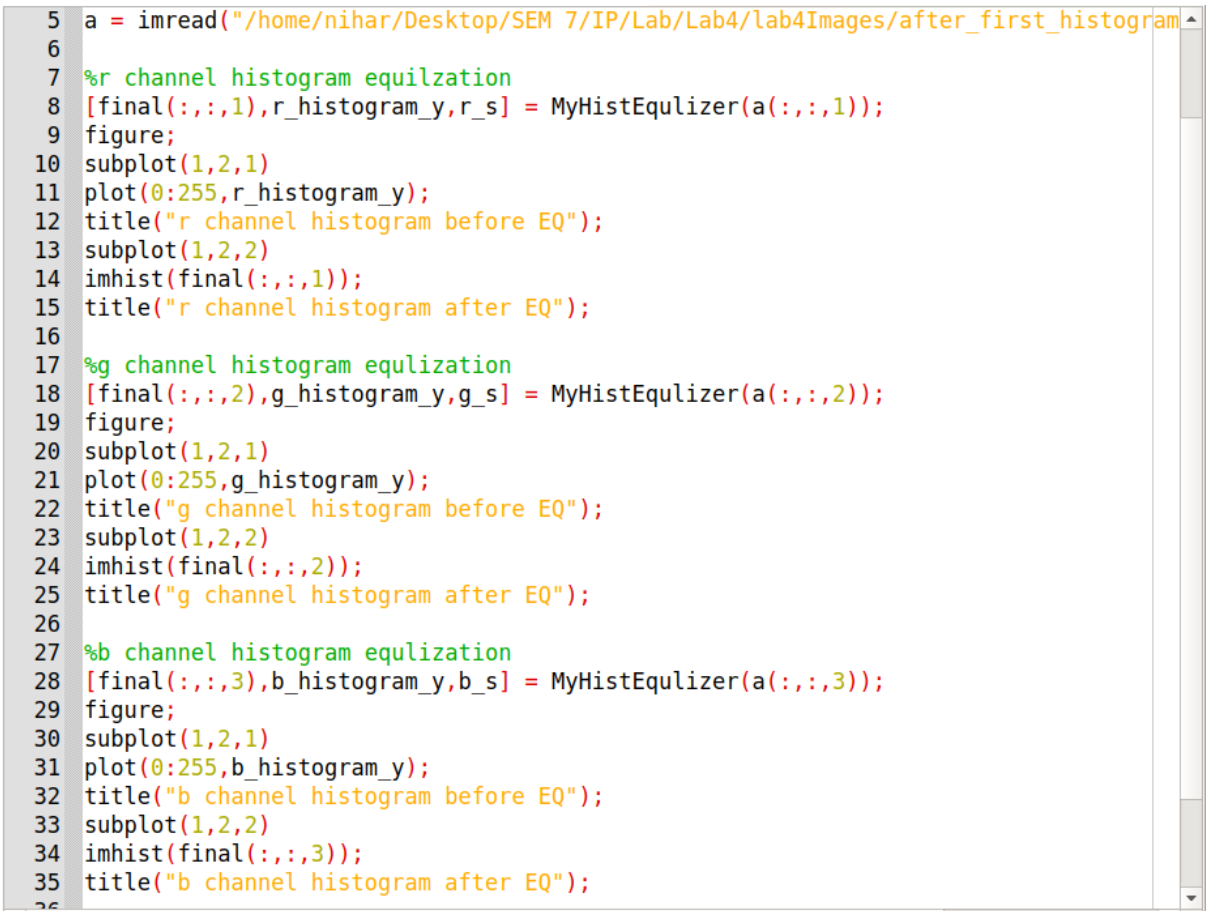
Figure 5
File
Edit
Tools
Z+
Z-
Insert Text
Axes
Grid
Autoscale

3. **Perform histogram equalization of equalized image obtained. Is second pass of the histogram equalization process useful? Justify your answer.**

→ **Solution :-**

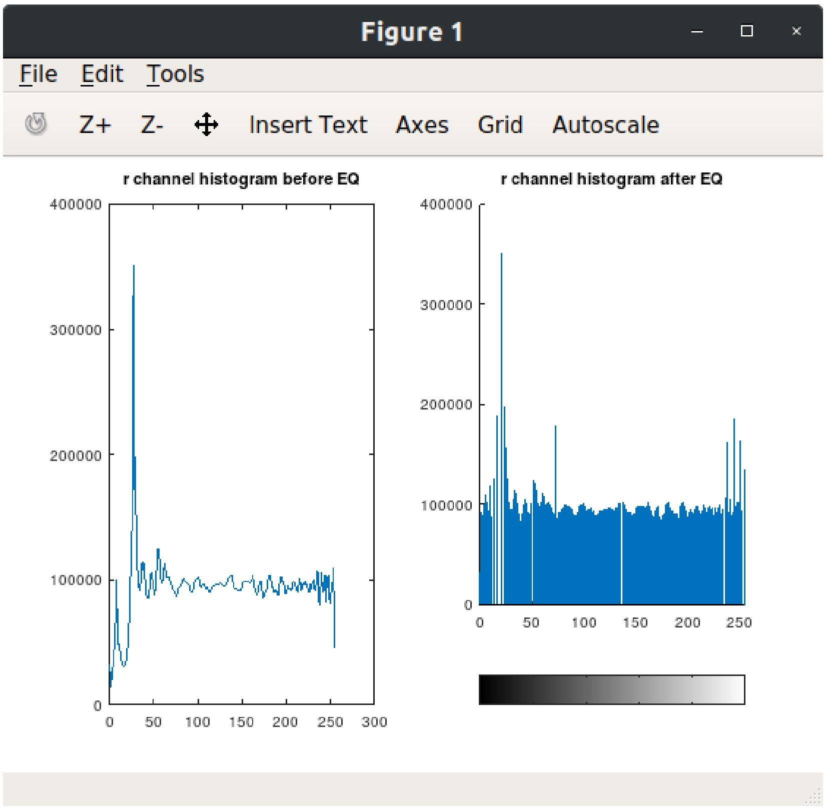
- If histogram equalization is applied twice, there is no change. This is because a histogram equalization is idempotent.

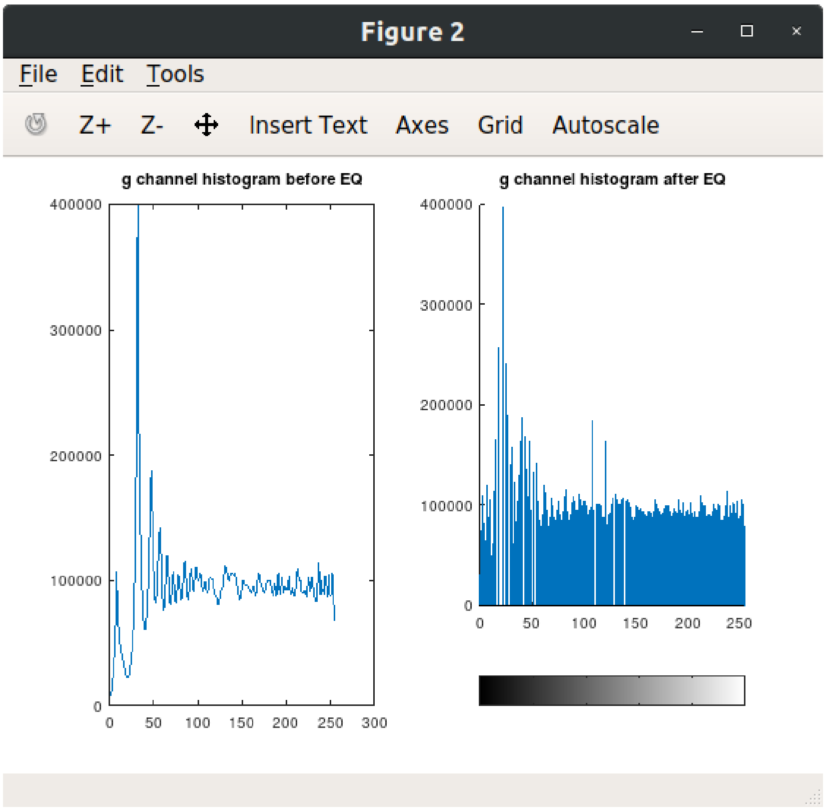
Code :

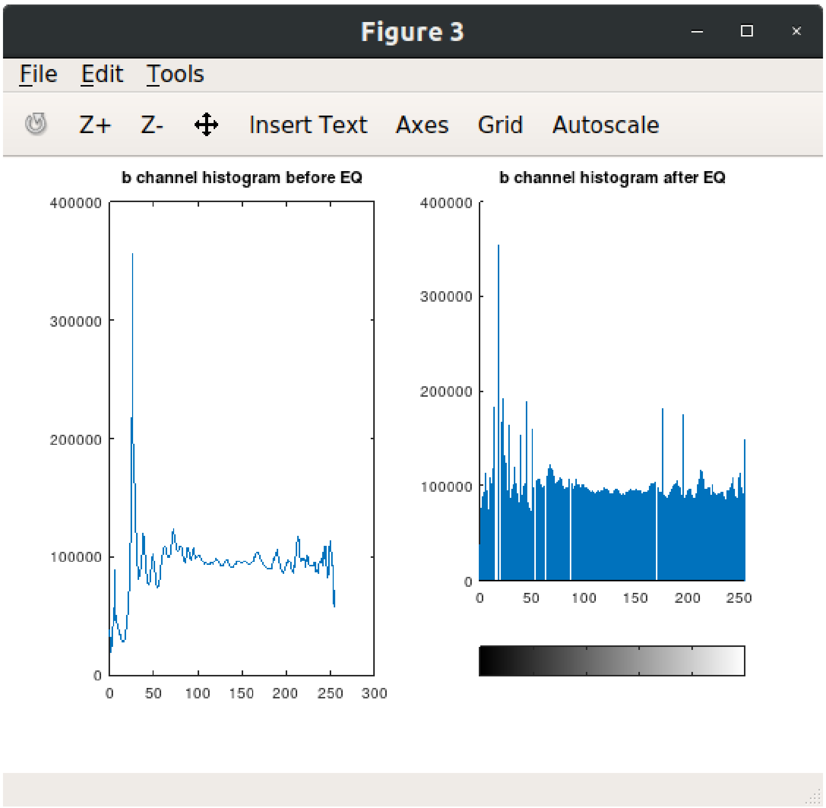


```
a = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/after_first_histogram

%r channel histogram equilzation
[final(:,:,1),r_histogram_y,r_s] = MyHistEqulizer(a(:,:,1));
figure;
subplot(1,2,1)
plot(0:255,r_histogram_y);
title("r channel histogram before EQ");
subplot(1,2,2)
imhist(final(:,:,1));
title("r channel histogram after EQ");

%g channel histogram equilzation
[final(:,:,2),g_histogram_y,g_s] = MyHistEqulizer(a(:,:,2));
figure;
subplot(1,2,1)
plot(0:255,g_histogram_y);
title("g channel histogram before EQ");
subplot(1,2,2)
imhist(final(:,:,2));
title("g channel histogram after EQ");

%b channel histogram equilzation
[final(:,:,3),b_histogram_y,b_s] = MyHistEqulizer(a(:,:,3));
figure;
subplot(1,2,1)
plot(0:255,b_histogram_y);
title("b channel histogram before EQ");
subplot(1,2,2)
imhist(final(:,:,3));
title("b channel histogram after EQ");
```

Comparision of histograms (before and after second equalization) :



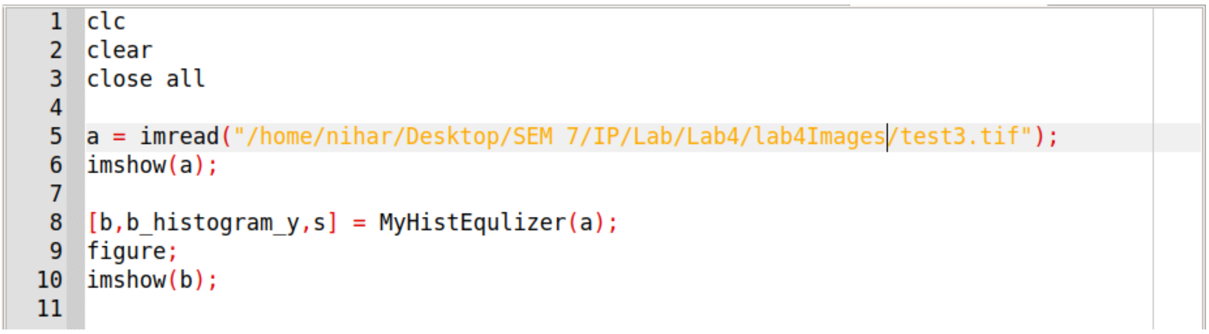




4. Perform histogram equalization for image ‘test3.jpg’

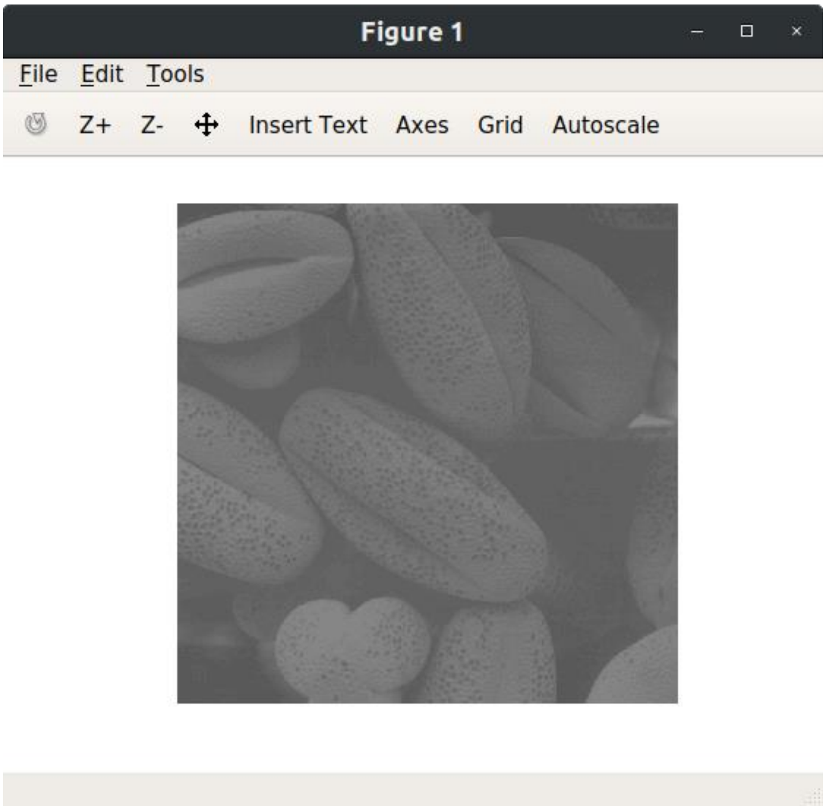
→ **Solution :-**

Code :

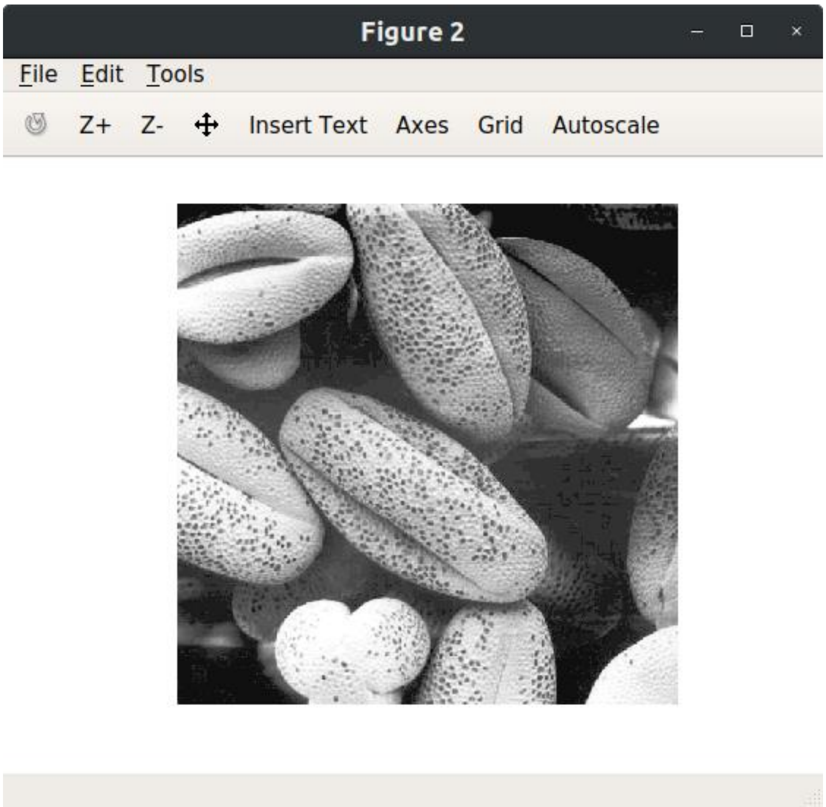


```
clc
clear
close all

a = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/test3.tif");
imshow(a);

[b,b_histogram_y,s] = MyHistEqulizer(a);
figure;
imshow(b);

```

Input Image :



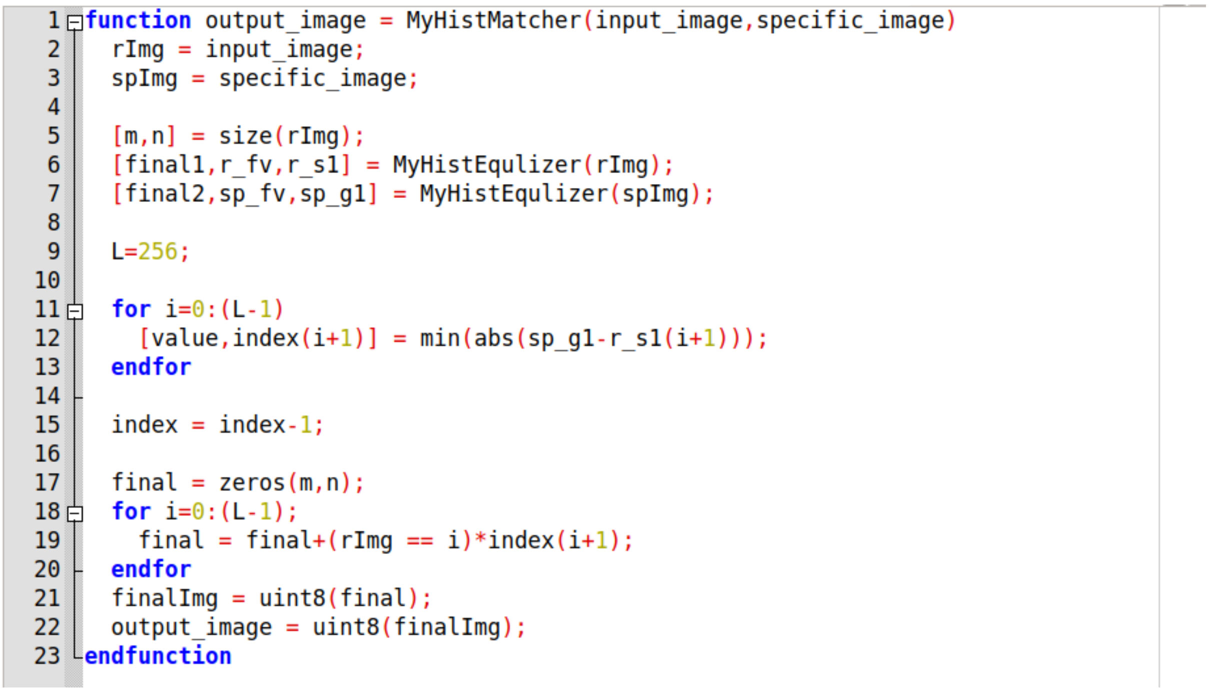
Output Image :



5. **Take any of your photograph, match it’s histogram with the histogram of image ‘test4.jpg’. plot histogram of original image, template and matched image for all three channels.**

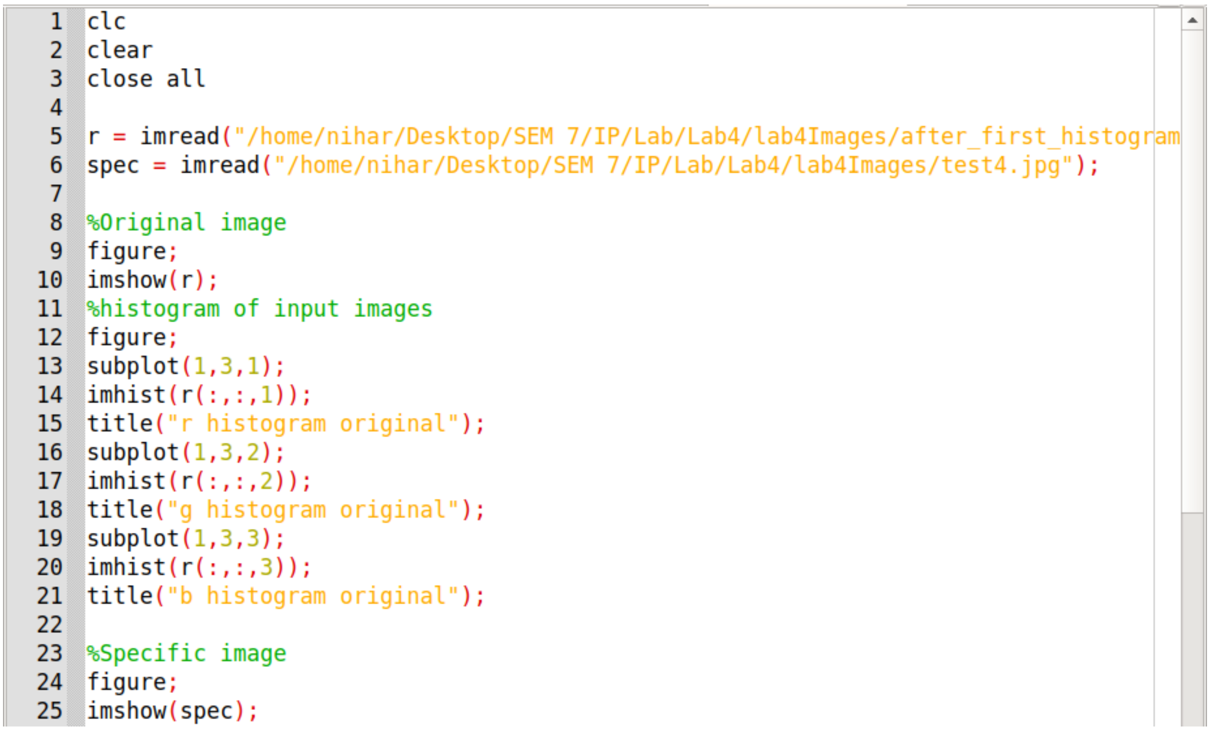
→ **Solution :-**

MyHistMatcher Function :

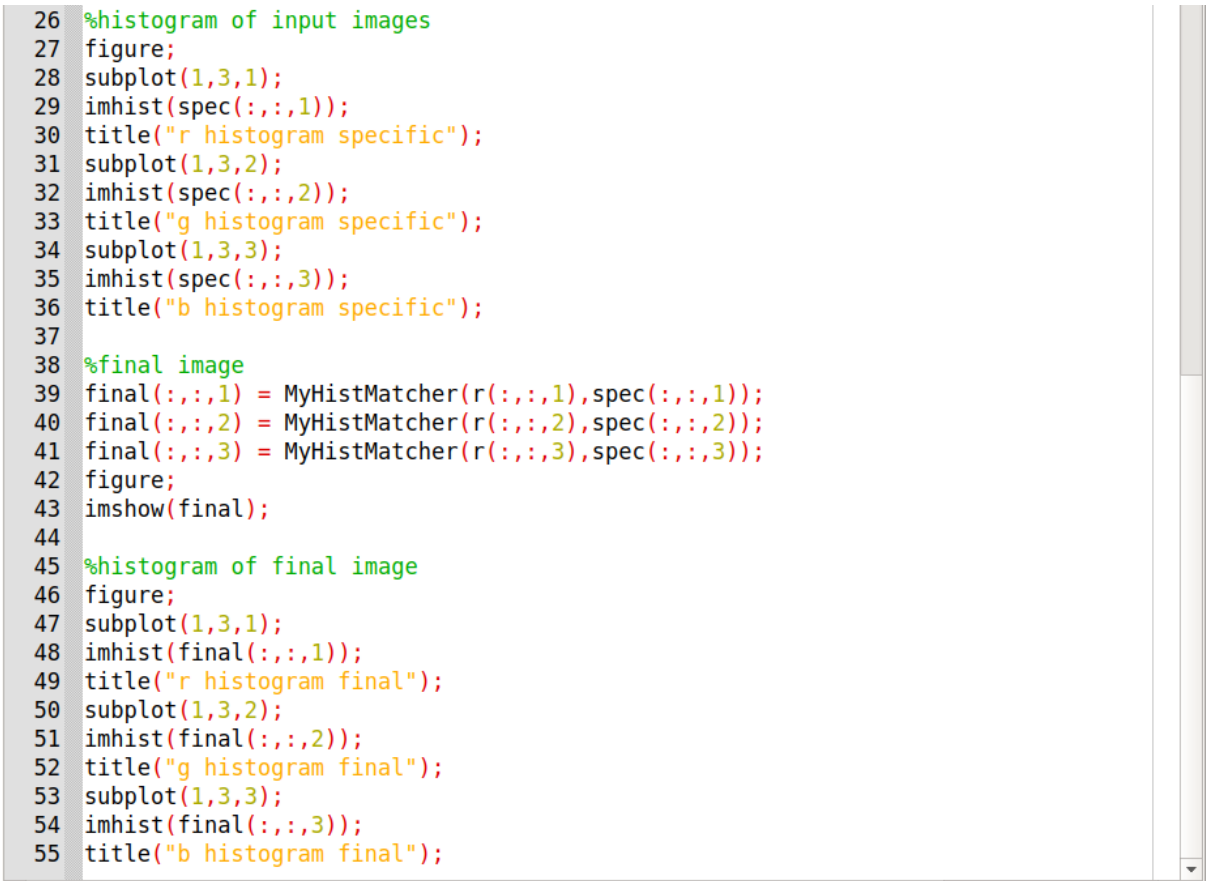


```
function output_image = MyHistMatcher(input_image,specific_image)
  rImg = input_image;
  spImg = specific_image;

  [m,n] = size(rImg);
  [final1,r_fv,r_s1] = MyHistEqulizer(rImg);
  [final2,sp_fv,sp_g1] = MyHistEqulizer(spImg);

  L=256;

  for i=0:(L-1)
    [value,index(i+1)] = min(abs(sp_g1-r_s1(i+1)));
  endfor

  index = index-1;

  final = zeros(m,n);
  for i=0:(L-1);
    final = final+(rImg == i)*index(i+1);
  endfor
  finalImg = uint8(final);
  output_image = uint8(finalImg);
endfunction
```

Code :

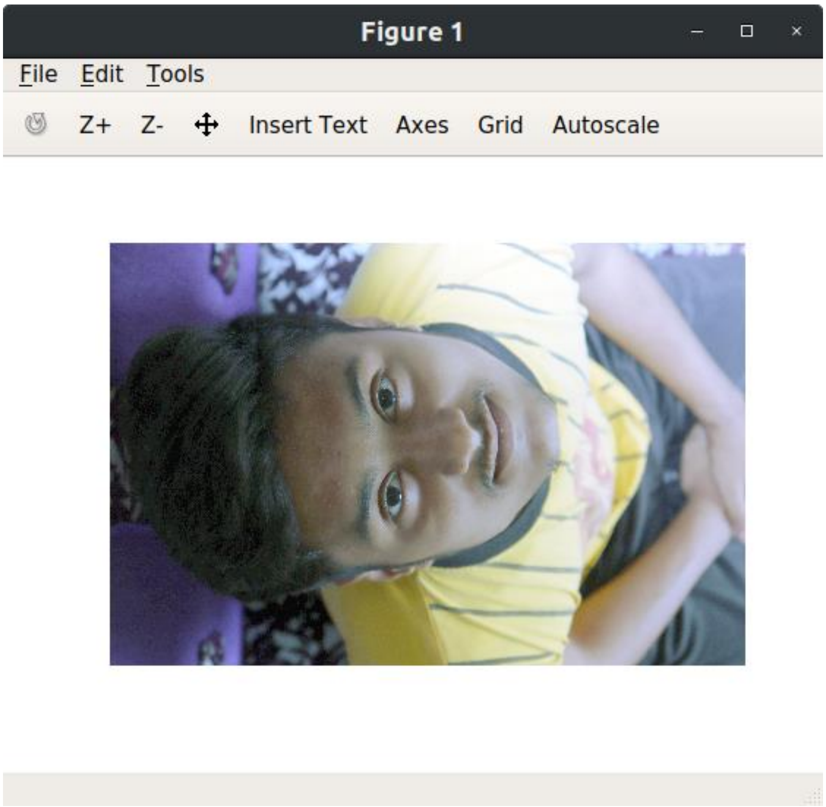


```
clc
clear
close all

r = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/after_first_histogram
spec = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab4/lab4Images/test4.jpg");

%Original image
figure;
imshow(r);
%histogram of input images
figure;
subplot(1,3,1);
imhist(r(:,:,1));
title("r histogram original");
subplot(1,3,2);
imhist(r(:,:,2));
title("g histogram original");
subplot(1,3,3);
imhist(r(:,:,3));
title("b histogram original");

%Specific image
figure;
imshow(spec);
```

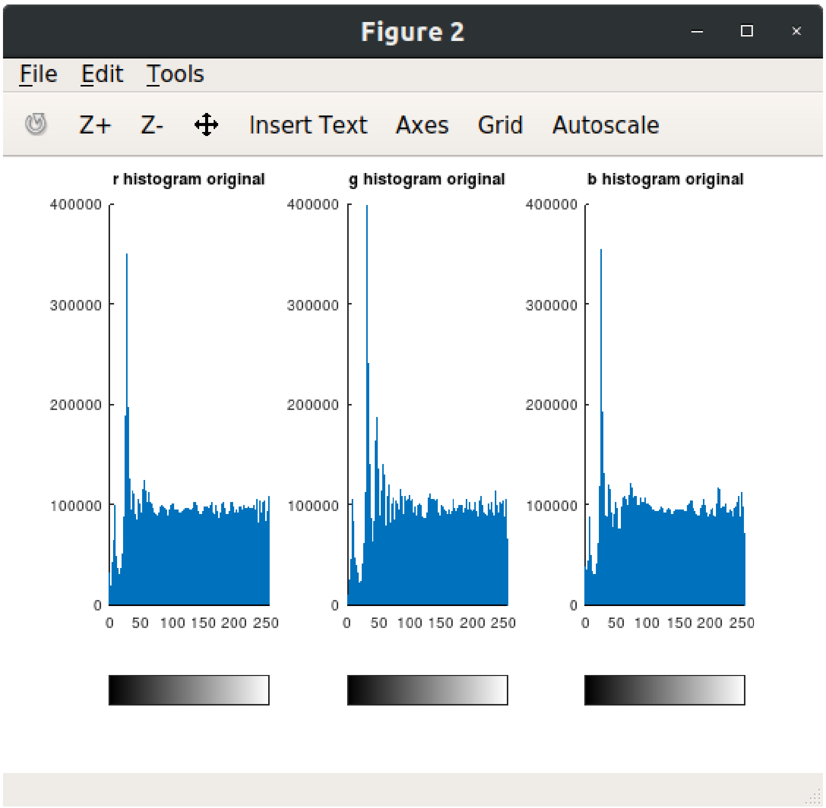


```
%histogram of input images
figure;
subplot(1,3,1);
imhist(spec(:,:,1));
title("r histogram specific");
subplot(1,3,2);
imhist(spec(:,:,2));
title("g histogram specific");
subplot(1,3,3);
imhist(spec(:,:,3));
title("b histogram specific");

%final image
final(:,:,1) = MyHistMatcher(r(:,:,1),spec(:,:,1));
final(:,:,2) = MyHistMatcher(r(:,:,2),spec(:,:,2));
final(:,:,3) = MyHistMatcher(r(:,:,3),spec(:,:,3));
figure;
imshow(final);

%histogram of final image
figure;
subplot(1,3,1);
imhist(final(:,:,1));
title("r histogram final");
subplot(1,3,2);
imhist(final(:,:,2));
title("g histogram final");
subplot(1,3,3);
imhist(final(:,:,3));
title("b histogram final");
```

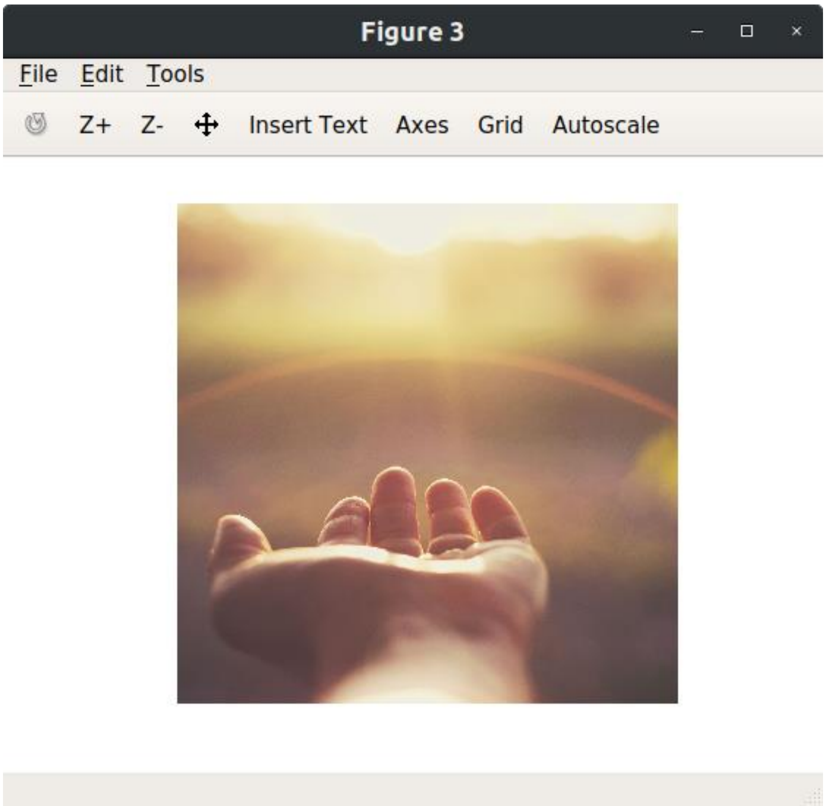
Input Image :



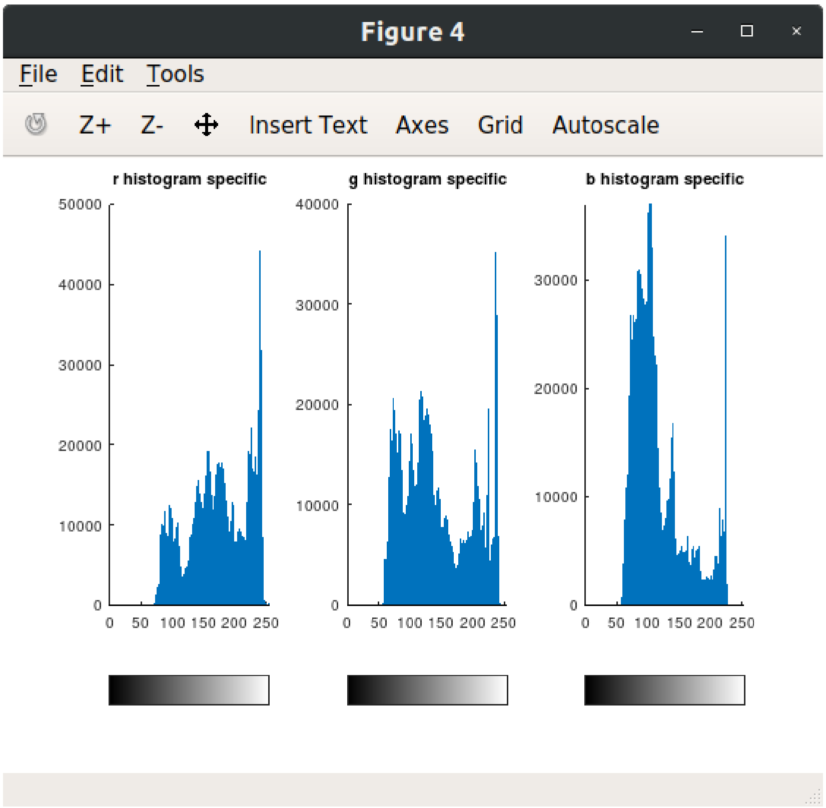
Histogram of Input Image :



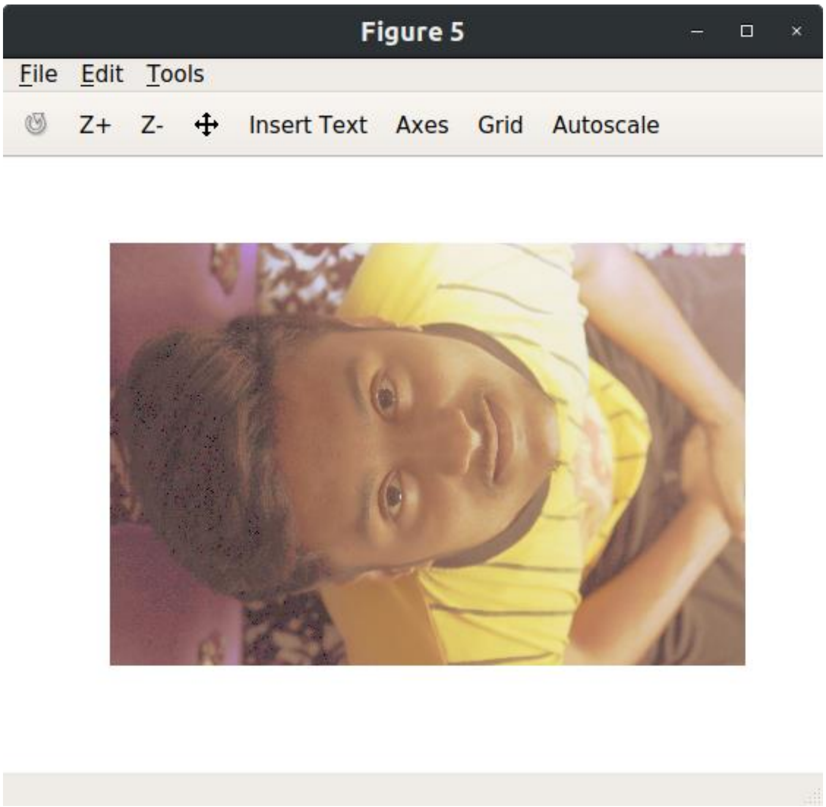
Specified Image :



Histogram of specified Image :



Output Image :



Histogram of Output Image :

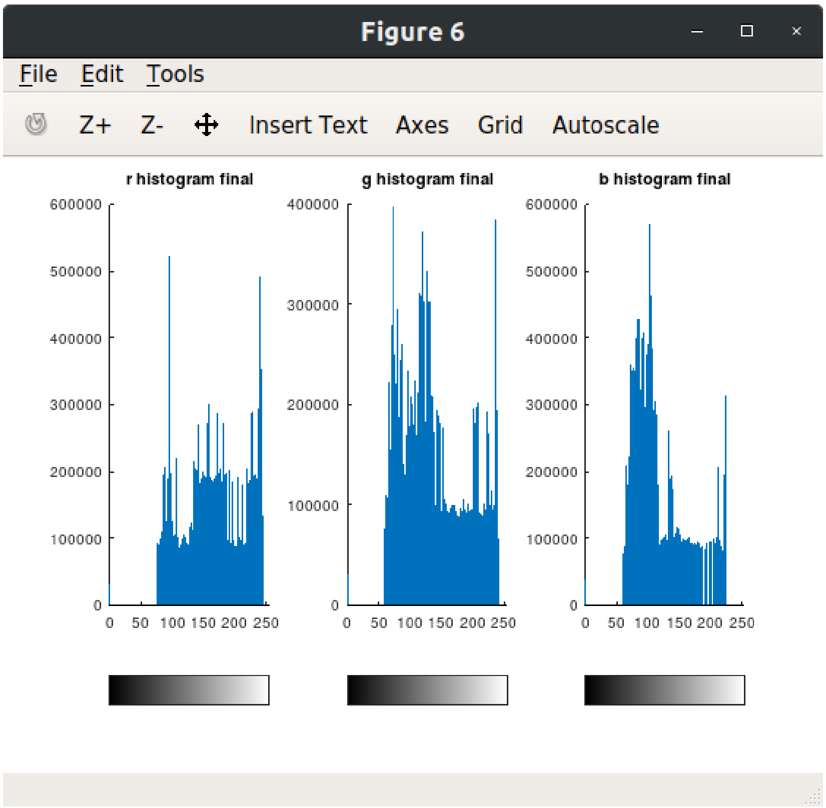


Figure 6
File Edit Tools
Z+ Z- Insert Text Axes Grid Autoscale
r histogram final
g histogram final
b histogram final