

**University of the Punjab**  
**Gujranwala Campus**  
**Department of Information Technology**



**Computer Vision**  
**Assignment**

**Submitted by:**

**Name: Rabia Saleem**

**Roll #BIT21045**

**Section: BSIT (Morning)**

**Semester: 7<sup>th</sup>**

**Submitted to:**

**Ms Fouqia Zafeer**

## EXERCISE:

Write a program which can read an image as an input and do the following automatically.  
Show the results of all steps.

1. Find the type of image: binary, gray or RGB.

```
% Load or use the input image stored in variable A
A = imread('pic1.png');
% Display the input image
figure('Name', 'Original Image');
imshow(A);
title('Original Image');
% Step 1: Identify image type
if islogical(A)
    img_type = 'Binary';
elseif ndims(A) == 3
    img_type = 'RGB';
else
    img_type = 'Grayscale';
end
% Display the image type
disp(['Image type: ', img_type]);
% Convert RGB to grayscale for further processing
if strcmp(img_type, 'RGB')
    img_gray = rgb2gray(A);
elseif strcmp(img_type, 'Grayscale')
    img_gray = A;
else
    img_gray = double(A) * 255; % Convert binary to grayscale
```

End

---

**2.Find the issue in image, over dark, over bright, low contrast, or normal. (Hint: can use histogram).**

**% Step 2: Analyze histogram**

figure('Name', 'Histogram');

histogram(img\_gray, 256);

title('Image Histogram');

**% Determine brightness and contrast issues**

mean\_intensity = mean(img\_gray(:));

contrast = max(img\_gray(:)) - min(img\_gray(:));

if mean\_intensity < 50

issue = 'Over Dark';

elseif mean\_intensity > 200

issue = 'Over Bright';

elseif contrast < 50

issue = 'Low Contrast';

else

issue = 'Normal';

end

**% Display the identified issue**

disp(['Image issue: ', issue]);

---

**3.Resolve the issue if any and show the final image after enhancement.**

**% Step 3: Resolve issues**

enhanced\_img = img\_gray; **% Start with the grayscale image**

switch issue

case 'Over Dark'

enhanced\_img = imadjust(img\_gray, stretchlim(img\_gray), []);

```

case 'Over Bright'
enhanced_img = imadjust(img_gray, [0.2, 1], []);
case 'Low Contrast'
enhanced_img = histeq(img_gray);
end

% Display results
figure('Name', 'Enhanced Image');
imshow(enhanced_img);
title('Enhanced Image');

```

---

#### 4. Test your program on following images

```

% Step 4: Compare original and enhanced images
if ~strcmp(issue, 'Normal')
disp(['Enhancement applied: ', issue]);
figure('Name', 'Comparison');
subplot(1, 2, 1);
imshow(img_gray);
title('Before Enhancement');
subplot(1, 2, 2);
imshow(enhanced_img);
title('After Enhancement');
else
disp('No enhancement needed.');
end

```

---

## RESULTS (MATLAB SCREENSHOTS);

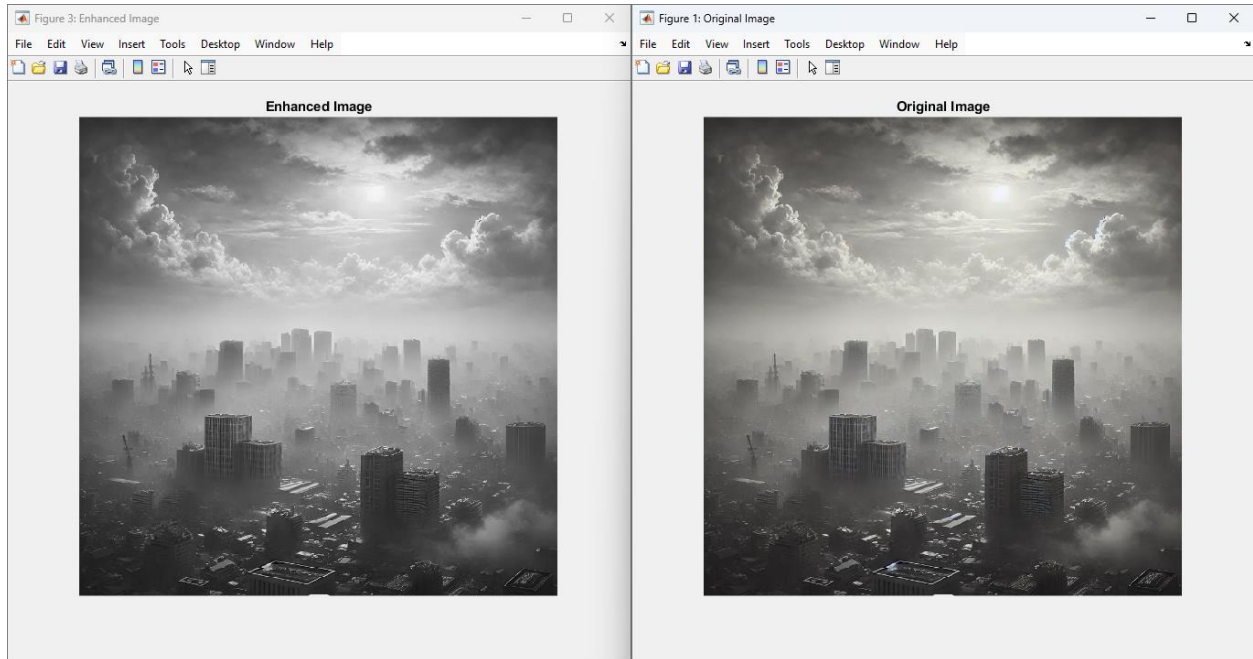


Figure # 1

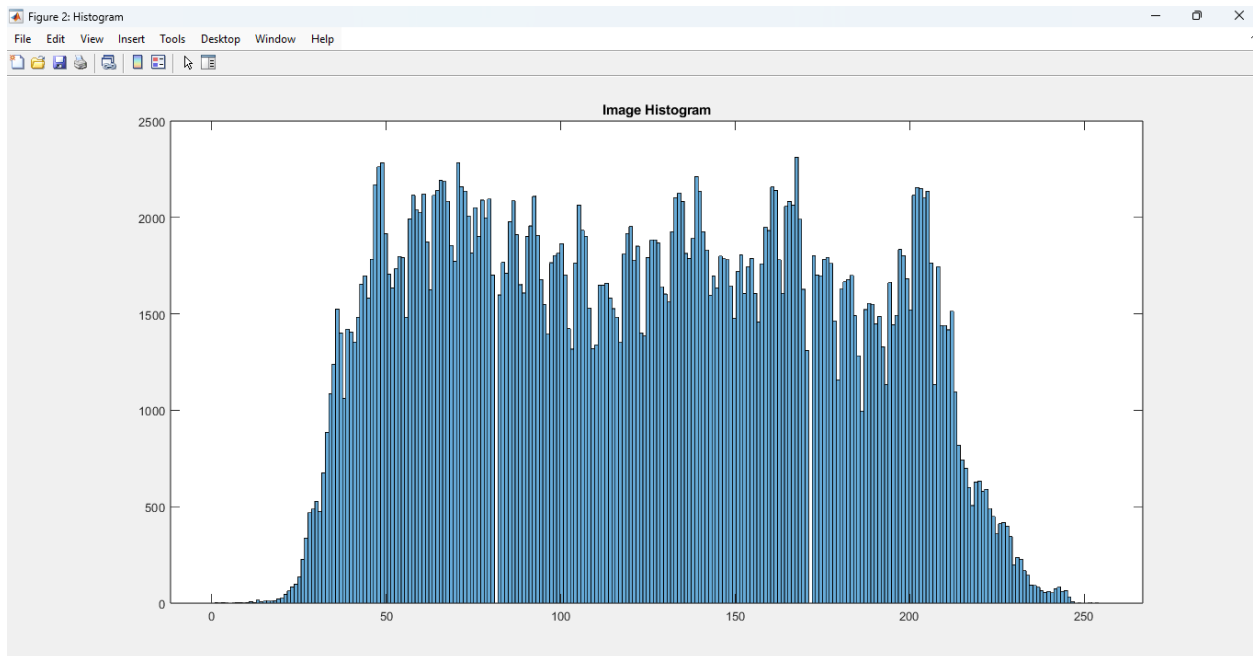


Figure # 2

Image type: RGB  
Image issue: Normal  
No enhancement needed.

Figure # 3








Name ▲	Value	
 A	574x574x3 uint8	
 contrast	253	
 enhanced_img	574x574 uint8	
 img_gray	574x574 uint8	
 img_type	'RGB'	
 issue	'Normal'	
 mean_intensity	123.9652	

Figure # 4