

```

%date - 19/01/2026
%created by - Siddharth Khandelwal;
clc;
clear;
close all;

% reading the input image
I = imread('input_image_unequalized.jpg');

% if image is RGB then convert it into grayscale
if size(I,3) == 3
    I = rgb2gray(I);
end

% converting image to uint8 format
I = uint8(I);

% getting size of the image
[M, N] = size(I);

% STEP 1: finding histogram of the image
hist_count = zeros(256,1);

for i = 1:M
    for j = 1:N
        intensity = I(i,j);
        hist_count(intensity + 1) = hist_count(intensity + 1) + 1;
    end
end

% STEP 2: calculating probability density function (PDF)
pdf = hist_count / (M * N);

% STEP 3: calculating cumulative distribution function (CDF)
cdf = zeros(256,1);
cdf(1) = pdf(1);

for k = 2:256
    cdf(k) = cdf(k-1) + pdf(k);
end

% STEP 4: normalizing the CDF values
cdf_min = min(cdf(cdf > 0));
cdf_norm = round((cdf - cdf_min) / (1 - cdf_min) * 255);

% STEP 5: mapping old pixel values to new values
Ieq = zeros(M,N, 'uint8');

for i = 1:M
    for j = 1:N
        Ieq(i,j) = cdf_norm(I(i,j) + 1);
    end
end

% displaying original image
figure;
imshow(I);
title('Before Histogram Equalization');

% displaying histogram and CDF of original image
figure;
bar(0:255, hist_count, 'r');
hold on;
plot(0:255, cdf * max(hist_count), 'k', 'LineWidth', 2);
hold off;
title('Histogram(red bars) and CDF (Original Image)black curve');

% displaying equalized image
figure;
imshow(Ieq);
title('After Histogram Equalization');

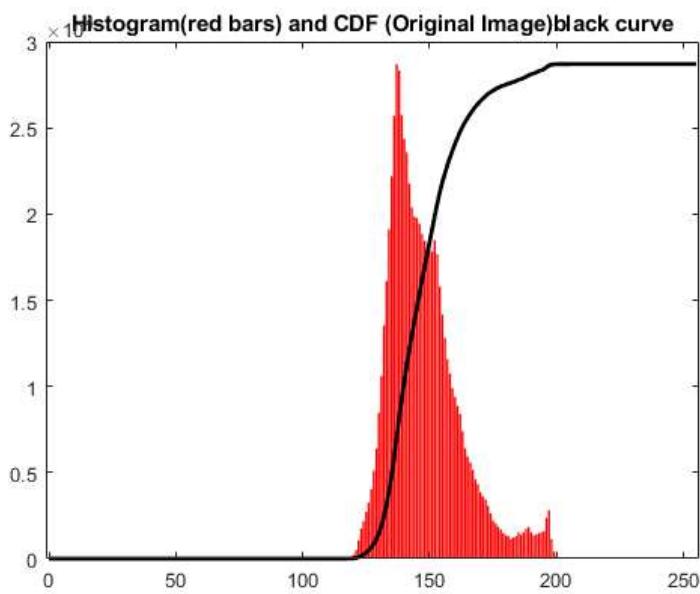
% calculating histogram of equalized image
hist_eq = zeros(256,1);
for i = 1:M
    for j = 1:N
        hist_eq(Ieq(i,j) + 1) = hist_eq(Ieq(i,j) + 1) + 1;
    end
end

```

```
% calculating CDF of equalized image
cdf_eq = cumsum(hist_eq) / (M*N);

% displaying histogram and CDF of equalized image
figure;
bar(0:255, hist_eq, 'r');
hold on;
plot(0:255, cdf_eq * max(hist_eq), 'k', 'LineWidth', 2);
hold off;
title('Histogram(red bars) and CDF (Equalized Image)(black curve)');
```

Before Histogram Equalization



After Histogram Equalization

