

---

```
clc;
clear all;
close all;

% Read and convert image to grayscale
input_image = imread('my_pic.jpg'); % Replace with your image path
grayscale_image = rgb2gray(input_image);

% Calculate histogram and CDF
[pixel_counts, intensity_bins] = imhist(grayscale_image);
pixel_pdf = pixel_counts / numel(grayscale_image);
pixel_cdf = cumsum(pixel_pdf);

% Perform histogram equalization
equalized_image = histeq(grayscale_image);
[equalized_pixel_counts, equalized_bins] = imhist(equalized_image);
equalized_pdf = equalized_pixel_counts / numel(equalized_image);
equalized_cdf = cumsum(equalized_pdf);

% Create subplots for visualization
figure('Position', [100 100 1200 800]);

% Original and equalized images
subplot(2, 3, 1);
imshow(grayscale_image);
title('Original Grayscale Image');

subplot(2, 3, 4);
imshow(equalized_image);
title('Histogram Equalized Image');

% Original histogram and CDF
subplot(2, 3, 2);
bar(intensity_bins, pixel_counts, 'FaceColor', [0.4 0.4 0.4]);
title('Original Histogram');
xlabel('Pixel Intensity (v)');
ylabel('Frequency');
grid on;

subplot(2, 3, 3);
plot(intensity_bins, pixel_cdf, 'LineWidth', 2);
title('Original CDF');
xlabel('Pixel Intensity (v)');
ylabel('CDF(v)');
grid on;
ylim([0 1]);

% Equalized histogram and CDF
subplot(2, 3, 5);
bar(equalized_bins, equalized_pixel_counts, 'FaceColor', [0.4 0.4 0.4]);
title('Equalized Histogram');
xlabel('Pixel Intensity (h(v))');
```

---

---

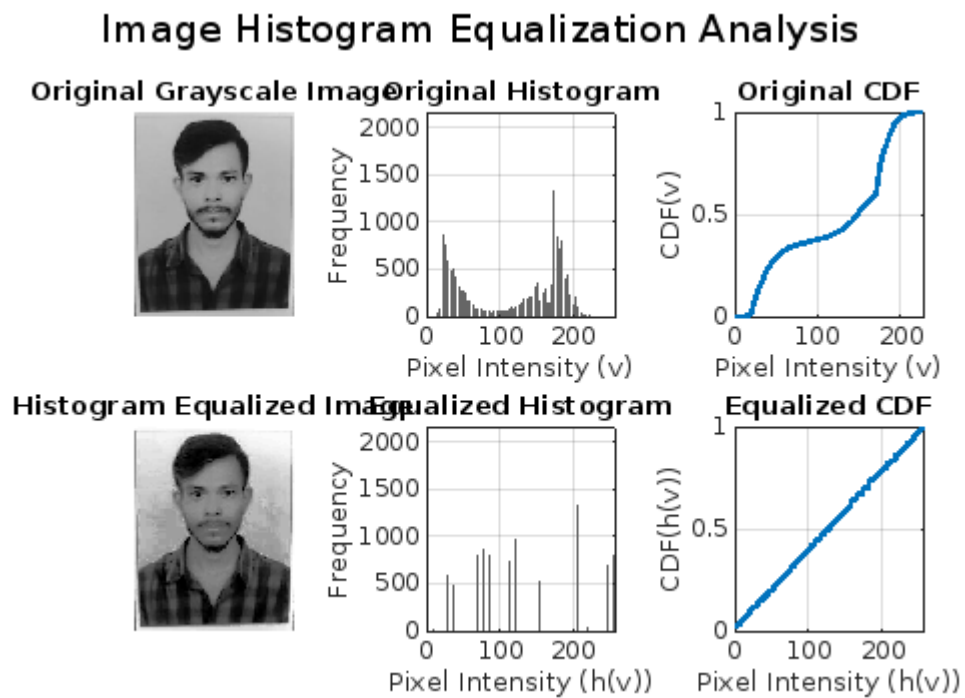
```

ylabel('Frequency');
grid on;

subplot(2, 3, 6);
plot(equalized_bins, equalized_cdf, 'LineWidth', 2);
title('Equalized CDF');
xlabel('Pixel Intensity (h(v))');
ylabel('CDF(h(v))');
grid on;
ylim([0 1]);

% Adjust spacing between subplots
sgtitle('Image Histogram Equalization Analysis', 'FontSize', 14);

```



*Published with MATLAB® R2024b*