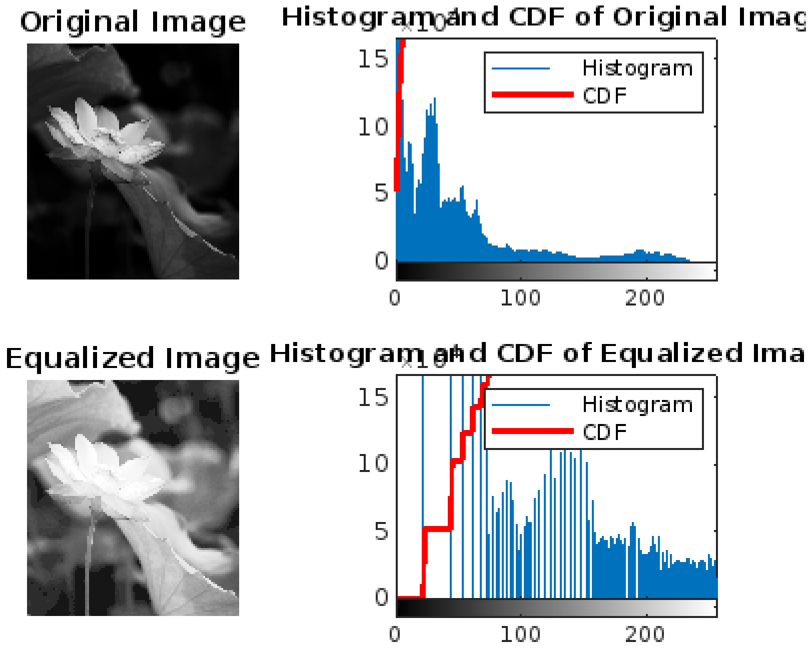
```
 % Prompt the user to select an image file
[fileName, filePath] = uigetfile({'.jpg;.jpeg;.png;.bmp', 'Image Files
(*.jpg, *.jpeg, *.png, *.bmp)'}, 'Select an Image File');
if isequal(fileName, 0)
    disp('No file selected. Exiting...');
    return; % Exit if no file is selected
end
fullFileName = fullfile(filePath, fileName);

% Read the input image
inputImage = imread(fullFileName);
if size(inputImage, 3) == 3
    inputImage = rgb2gray(inputImage); % Convert to grayscale if the image is
RGB
end

% Get the dimensions of the image
[rows, cols] = size(inputImage);

% Calculate the histogram for the original image
histogramOriginal = zeros(256, 1); % Initialize histogram
for i = 1:rows
    for j = 1:cols
        intensity = inputImage(i, j);
        histogramOriginal(intensity + 1) = histogramOriginal(intensity + 1) +
1;
    end
end

% Normalize the histogram to get the PDF for the original image
pdfOriginal = histogramOriginal / (rows * cols);

% Calculate the CDF for the original image
cdfOriginal = cumsum(pdfOriginal);

% Map the intensities to equalized values
equalizedValues = round(cdfOriginal * 255);

% Create the equalized image
equalizedImage = zeros(size(inputImage));
for i = 1:rows
    for j = 1:cols
        equalizedImage(i, j) = equalizedValues(inputImage(i, j) + 1);
    end
end
equalizedImage = uint8(equalizedImage); % Convert to uint8 for display

% Calculate the histogram for the equalized image
histogramEqualized = zeros(256, 1); % Initialize histogram
for i = 1:rows
    for j = 1:cols
        intensity = equalizedImage(i, j);
        histogramEqualized(intensity + 1) = histogramEqualized(intensity + 1)
+ 1;
    end
end

% Normalize the histogram to get the PDF for the equalized image
pdfEqualized = histogramEqualized / (rows * cols);

% Calculate the CDF for the equalized image
cdfEqualized = cumsum(pdfEqualized);

% Display the results
figure;

% Original Image and its histogram
subplot(2, 2, 1);
imshow(inputImage);
title('Original Image');

subplot(2, 2, 2);
imhist(inputImage);
hold on;
plot(cdfOriginal * max(histogramOriginal), 'r', 'LineWidth', 2); % Scale CDF
for visualization
legend('Histogram', 'CDF');
title('Histogram and CDF of Original Image');

% Equalized Image and its histogram
subplot(2, 2, 3);
imshow(equalizedImage);
title('Equalized Image');

subplot(2, 2, 4);
imhist(equalizedImage);
hold on;
plot(cdfEqualized * max(histogramEqualized), 'r', 'LineWidth', 2); % Scale
CDF for visualization
legend('Histogram', 'CDF');
title('Histogram and CDF of Equalized Image');
```



Published with MATLAB® R2024b