

---

```
% A Program for Histogram Equalization Without histeq() or built-in functions
clc;
close all;
clear all;

% Read the image
I = imread('gratisography-cool-cat-800x525.jpg');

% Convert to grayscale if the image is RGB
if size(I, 3) == 3
    I = rgb2gray(I);
end

% Get the dimensions of the image
[r, c] = size(I);
no_of_pixels = r * c; % Total number of pixels

% Step 1: Calculate the histogram (PDF)
h = zeros(1, 256); % Initialize histogram array
for i = 1:r
    for j = 1:c
        intensity = I(i, j); % Pixel intensity
        h(intensity + 1) = h(intensity + 1) + 1; % Increment intensity count
    end
end

% Step 2: Normalize the histogram to get PDF
pdf = zeros(1, 256); % Initialize PDF array
for i = 1:256
    pdf(i) = h(i) / no_of_pixels; % Normalize by total pixels
end

% Step 3: Compute the cumulative distribution function (CDF)
cdf = zeros(1, 256); % Initialize CDF array
cdf(1) = pdf(1); % First value of CDF is same as the first PDF value
for i = 2:256
    cdf(i) = cdf(i - 1) + pdf(i); % Cumulative sum
end

% Step 4: Scale the CDF to the range [0, 255] (Equalized Mapping)
equalized_mapping = zeros(1, 256); % Initialize mapping array
for i = 1:256
    equalized_mapping(i) = round(cdf(i) * 255); % Scale to 0-255 range
end

% Step 5: Apply the equalized mapping to the image
I_equalized = zeros(r, c, 'uint8'); % Preallocate for equalized image
for i = 1:r
    for j = 1:c
        I_equalized(i, j) = equalized_mapping(I(i, j) + 1); % Map old
intensities
    end
end
```

---

---

```

end

% Step 6: Calculate the histogram of the equalized image
h_equalized = zeros(1, 256); % Initialize histogram for equalized image
for i = 1:r
    for j = 1:c
        intensity = I_equalized(i, j); % Pixel intensity in equalized image
        h_equalized(intensity + 1) = h_equalized(intensity + 1) + 1; %
Increment
    end
end

% Normalize the equalized histogram
pdf_equalized = zeros(1, 256);
for i = 1:256
    pdf_equalized(i) = h_equalized(i) / no_of_pixels;
end

% Compute CDF for equalized image
cdf_equalized = zeros(1, 256);
cdf_equalized(1) = pdf_equalized(1);
for i = 2:256
    cdf_equalized(i) = cdf_equalized(i - 1) + pdf_equalized(i);
end

% Step 7: Display Results
figure;

% Original Image
subplot(2, 2, 1);
imshow(I);
title('Original Image');

% Histogram Equalized Image
subplot(2, 2, 2);
imshow(I_equalized);
title('Equalized Image');

% Histogram (PDF) Comparison
subplot(2, 2, 3);
stem(0:255, pdf, 'b', 'Marker', 'none', 'LineWidth', 1.5); hold on;
stem(0:255, pdf_equalized, 'r', 'Marker', 'none', 'LineWidth', 1.5);
xlabel('Intensity Levels');
ylabel('Normalized Frequency');
title(' Original (Blue) vs. Equalized (Red)');
grid on;

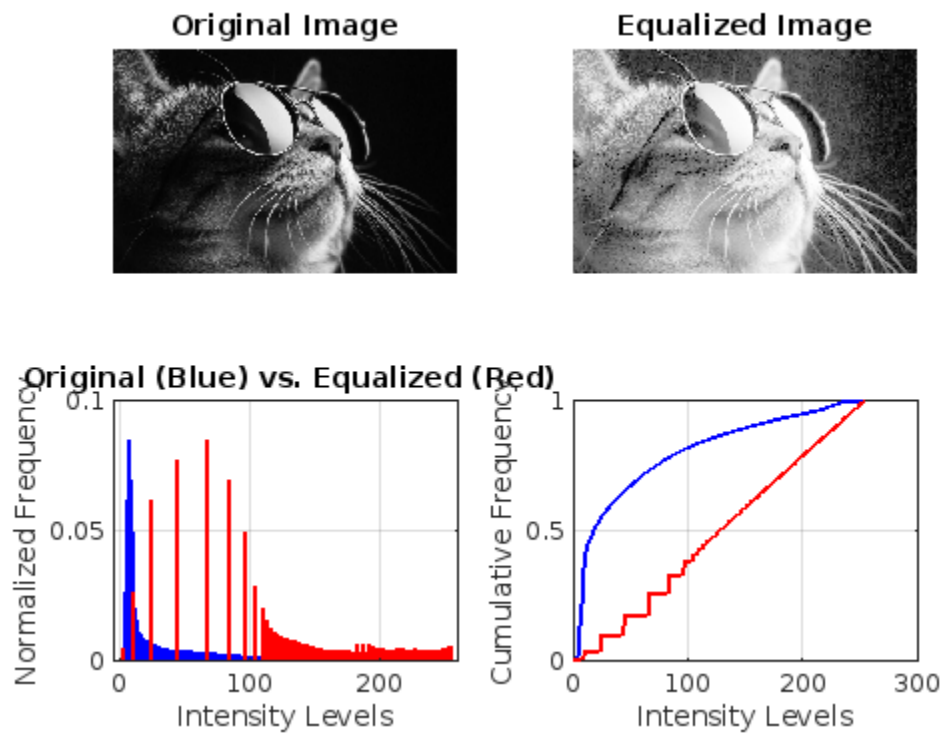
% Cumulative Histogram (CDF) Comparison
subplot(2, 2, 4);
plot(0:255, cdf, 'b', 'LineWidth', 1.5); hold on;
plot(0:255, cdf_equalized, 'r', 'LineWidth', 1.5);
xlabel('Intensity Levels');
ylabel('Cumulative Frequency');

```

---

---

```
%title(' Original (Blue) vs. Equalized (Red)');  
grid on;
```



*Published with MATLAB® R2024b*