

```
% Problem 2: Intensity Transformations: Gamma Mapping, Full-scale Contrast
% Stretch, and Histogram Equalization.
clear all;

input_image = double(imread('books.tif')); % reading the original image. This image is ✓
8 bits/pixel gray-scale
% image.

% Plotting the original image.
figure % figure creates figure graphics objects. figure objects are the individual ✓
windows on the screen
% in which MATLAB displays graphical output.
subplot(2,2,1); % subplot(m, n, p) divides the current figure into an m-by-n grid and ✓
creates axes in the
% position specified by p. Here, m=n=2, p=1.
imshow(uint8(input_image)); % It will display the gray-scale image in the figure and ✓
% it will convert each and every pixel value of the input image into the range of 0 to ✓
255.
title('Original Books Image'); % It will add the specified title for the current plot.

max_val = max(input_image(:)); % It will calculate the maximum value from the input ✓
image. It
% will be further used for image scaling for generating gamma mapped image.
min_val = min(input_image(:)); % It will calculate the minimum value from the input ✓
image. It
% will be further used for image scaling for generating gamma mapped image.
input_image_scaled = (input_image - min_val) / (max_val - min_val); % Here, we are ✓
scaling the pixel
% intensity. It is basically called the normalixation of all the pixel
% values in the image. We are scaling this image from [0, 255] to [0, 1]
% to apply power law.
gamma_transformed_image = (input_image_scaled.^0.5) * 255; % Here, we applied power law ✓
with gamma = 0.5 and
% after that we again need to rescale the image from [0,1] to [0, 255].Now, our gamma ✓
mapped image is ready.

subplot(2,2,2); % subplot(m, n, p) divides the current figure into an m-by-n grid and ✓
creates axes in the
% position specified by p. Here, m=n=p=2.
imshow(uint8(gamma_transformed_image)); % It will display the gray-scale image in the ✓
figure and
% it will convert each and every pixel value of the input image into the range of 0 to ✓
255.
title('Gamma Transformed Image'); % It will add the specified title for the current ✓
plot.

full_scale_contrast_stretched_image = round((2^8-1) * input_image_scaled); % Here, we ✓
created the full scale
% contrast stretched image from the original input scaled image.
subplot(2,2,3); % subplot(m, n, p) divides the current figure into an m-by-n grid and ✓
```

```
creates axes in the
% position specified by p. Here, m=n=2, p=3.
imshow(uint8(full_scale_contrast_stretched_image)); % It will display the gray-scale
image in the figure and
% it will convert each and every pixel value of the input image into the range of 0 to
255.
title('Full Scale Contrast Stretched Transformed Image'); % It will add the specified
title for the current
% plot.

cummulative_histogram_equalized_image = cumsum(hist(reshape(input_image,[],1),
linspace(0,255,256))); % It will
% create a cumulative histogram equalized image. hist() function will
% create a histogram bar chart of the elements in the given vector.
% reshape() method will change the shape of our image.
intermediate_image = cummulative_histogram_equalized_image(uint8(input_image) + 1); %
It will create an
% intermediate image to create final histogram equalized image. uint8() will convert
the value
% of each pixel from 0 to 255.
hei = round((2^8 - 1)*((intermediate_image - min(intermediate_image(:)))/(max
(intermediate_image(:))- ...
    min(intermediate_image(:))))); % It will simply do the scalling of the image to
generate the final histogram
% equalized image.
subplot(2,2,4); % subplot(m, n, p) divides the current figure into an m-by-n grid and
creates axes in the
% position specified by p. Here, m=n=2, p=4.
imshow(uint8(hei)); % It will display the gray-scale image in the figure and
% it will convert each and every pixel value of the input image into the range of 0 to
255.
title('Histogram Equalized Image'); % It will add the specified title for the current
% plot.

figure; % figure creates figure graphics objects. figure objects are the individual
windows on the screen
% in which MATLAB displays graphical output.
original_image_histogram = hist(reshape(input_image,[],1), linspace(0,255,256));
subplot(2,2,1); % subplot(m, n, p) divides the current figure into an m-by-n grid and
creates axes in the
% position specified by p. Here, m=n=2, p=1.
bar(original_image_histogram); % It will create a bar chart of the given image.
title('Histogram of Original Books Image'); % It will add the specified title for the
current
% plot.

gamma_transformed_image_histogram = hist(reshape(gamma_transformed_image,[],1),
linspace(0,255,256));
```

```
subplot(2,2,2); % subplot(m, n, p) divides the current figure into an m-by-n grid and✓
creates axes in the
% position specified by p. Here, m=n=p=2.
bar(gamma_transformed_image_histogram); % It will create a bar chart of the given✓
image.
title('Histogram of Gamma Transformed Image'); % It will add the specified title for✓
the current
% plot.

fscs_image_histogram = hist(reshape(full_scale_contrast_stretched_image,[],1), linspace✓
(0,255,256));
subplot(2,2,3); % subplot(m, n, p) divides the current figure into an m-by-n grid and✓
creates axes in the
% position specified by p. Here, m=n=2, p=3.
bar(fscs_image_histogram); % It will create a bar chart of the given image.
title('Histogram of Full Scale Contrast Stretched Image'); % It will add the specified✓
title for the current
% plot.

hei_image_histogram = hist(reshape(hei,[],1), linspace(0,255,256));
subplot(2,2,4); % subplot(m, n, p) divides the current figure into an m-by-n grid and✓
creates axes in the
% position specified by p. Here, m=n=2, p=4.
bar(hei_image_histogram); % It will create a bar chart of the given image.
title('Histogram of Histogram Equalized Image'); % It will add the specified title for✓
the current
% plot.
```