LAPORAN PRAKTIKUM PENGOLAHAN CITRA DIGITAL
8. OTHER HISTOGRAM MODIFICATION TECHNIQUES

Disusun oleh :

Nama : Valencio Arjun

NPM : 2327250049

Kelas : IF41

PROGRAM STUDI INFORMATIKA

FAKULTAS ILMU KOMPUTER DAN REKAYASA

UNIVERSITAS MULTI DATA PALEMBANG

2024

TUTORIAL : OTHER HISTOGRAM MODIFICATION TECHNIQUES

Goal

The goal of this tutorial is to learn how to perform other common histogram modification operations.

Objectives

- Learn how to adjust brightness of an image by histogram sliding.
- Learn how to use the `imadjust` function.
- Learn how to use the `stretchlim` function.
- Explore adjusting image contrast through histogram stretching (also known as input cropping).
- Learn how to adjust image contrast with histogram shrinking (also known as output cropping).

Procedure

Histogram sliding is the process of adding or subtracting a constant brightness value to all pixels in the image. When implementing histogram sliding, we must make sure that pixel values do not go outside the boundaries of the gray scale. Therefore, any pixels that result in values greater than 1 after adjustment will be set to 1. Likewise, any pixels resulting in values less than zero after adjustment will be set to 0.

1. Display original image and prepare `subplot`.

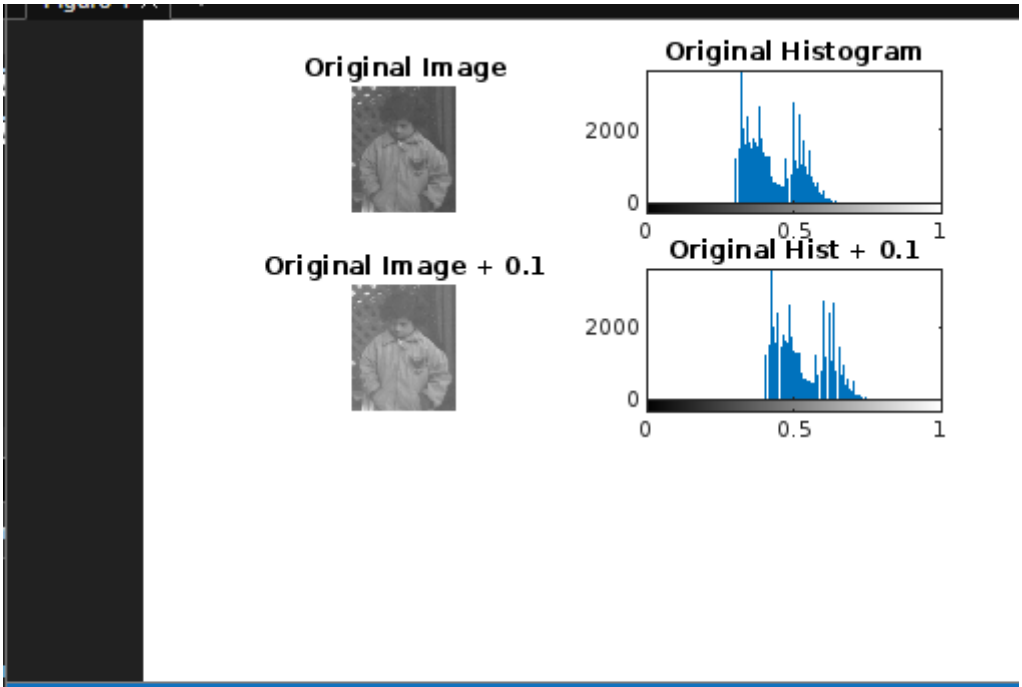
```
J = imread('pout.tif');
I = im2double(J);
clear J
figure, subplot(3,2,1), imshow(I), title('Original Image')
subplot(3,2,2), imhist(I), axis tight, ...
    title('Original Histogram')
```

Command Window

```
>> J = imread('pout.tif');
I = im2double(J);
clear J
figure,subplot(3,2,1),imshow(I),title('Original Image')
subplot(3,2,2), imhist(I), axis tight, title('Original Histogram')
>> %valencio arjun
```

2. Obtain a brighter version of the input image by adding 0.1 to each pixel.

```
const = 0.1;
I2 = I + const;
subplot(3,2,3), imshow(I2), title('Original Image + 0.1')
subplot(3,2,4), imhist(I2), axis tight, ...
    title('Original Hist + 0.1')
```




```
Command Window
subplot(3,2,2), imhist(I), axis tight, title('Original Histogram')
>> const = 0.1;
I2 = I + const;
subplot(3,2,3), imshow(I2), title('Original Image + 0.1')
subplot(3,2,4), imhist(I2), axis tight, title('Original Hist + 0.1')
>> %valencio arjun
```

Question 1 How did the histogram change after the adjustment?

the histogram shifted to the right, toward higher intensity values.

3. Produce another brighter image by adding 0.5 to original image.

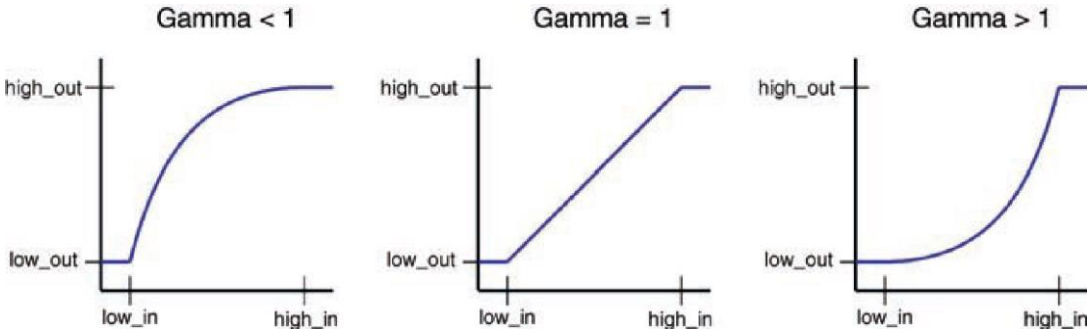
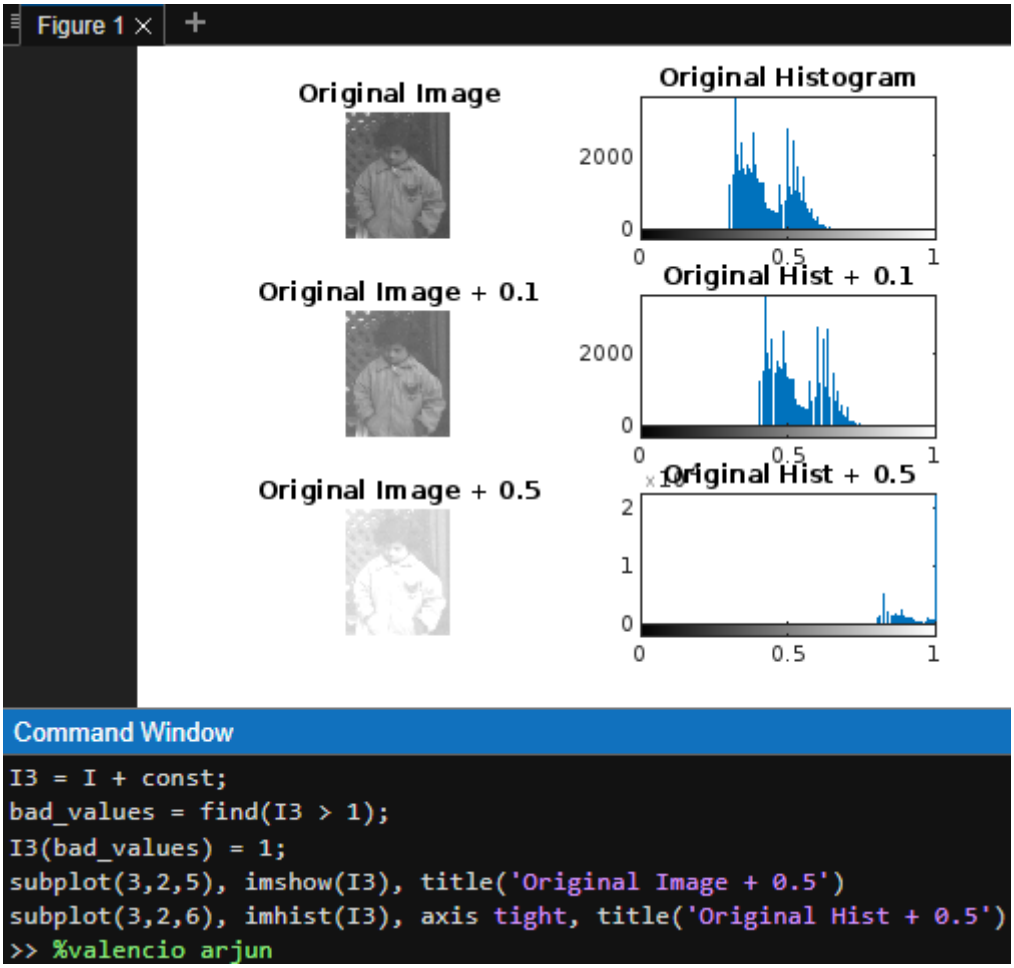


FIGURE 8.1 Gamma transformations for different values of gamma

```
const = 0.5;
I3 = I + const;
bad_values = find(I3 > 1);
I3(bad_values) = 1;
subplot(3,2,5), imshow(I3), title('Original Image + 0.5')
subplot(3,2,6), imhist(I3), axis tight, ...
    title('Original Hist + 0.5')
```



Question 2 What does the variable `bad_values` contain?

contains the linear indices of all pixels in `I3` whose intensity values are greater than 1.

Question 3 Why does the third plot show such an excessive number of pixels with a value of 1?

Because a large number of pixel values in I3 exceeded 1 after the + 0.5 addition and were then manually clipped (set) to 1.

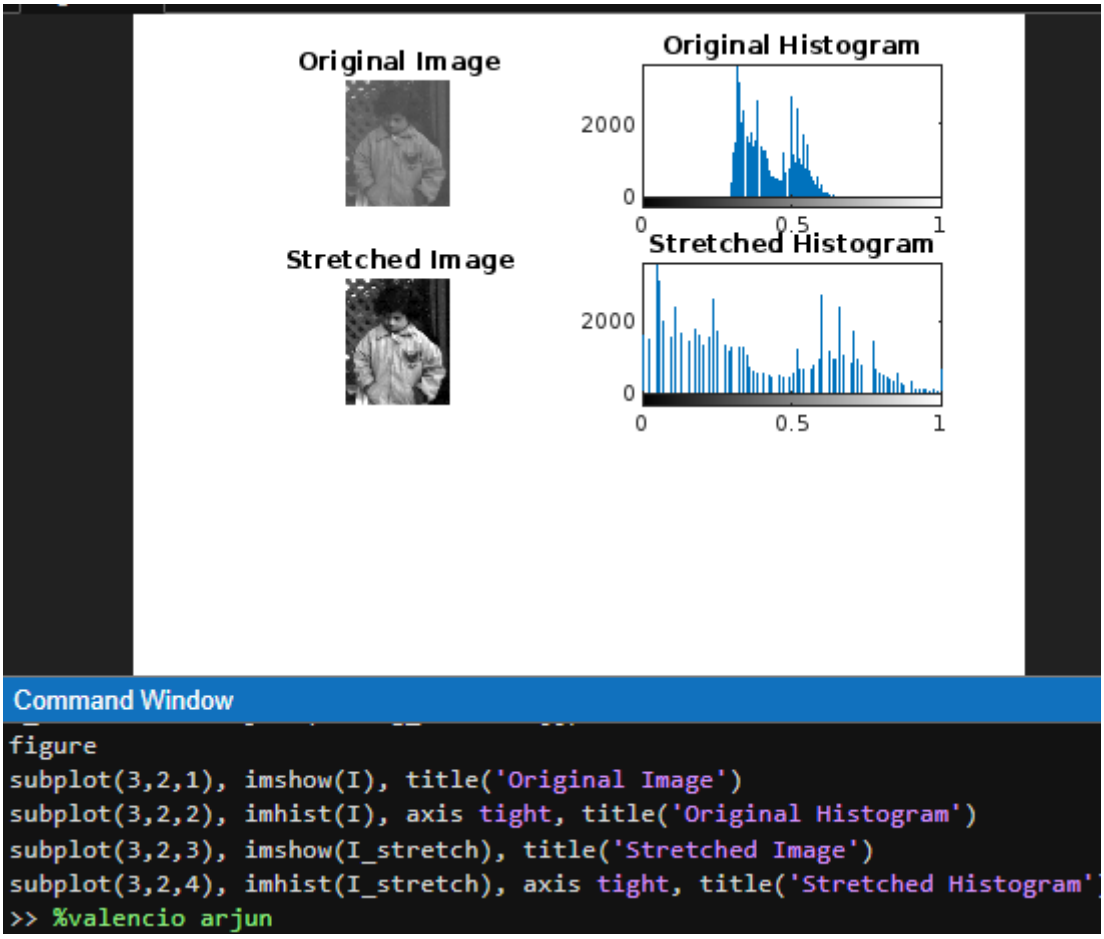
The brightness of an image can also be modified using the `imadd` function, which takes care of truncating and rounding off values outside the desired range in the output image. Histogram stretching and shrinking can be achieved through use of the `imadjust` function. The syntax for the function is as follows:

```
J = imadjust(I,[low_in; high_in],[low_out; high_out], gamma)
```

Figure 8.1 illustrates what the transformation functions look like when different values of gamma are used. As we already know from Chapter 9, the value of gamma is the exponent in power law transformation. Any values below `low_in` and above `high_in` are clipped or simply mapped to `low_out` and `high_out`, respectively. Only values in between these limits are affected by the curve. Gamma values less than 1 create a weighted curve toward the brighter range, and gamma values greater than 1 weight toward the darker region. The default value of gamma is 1. Let us explore how to use `imadjust` to perform histogram stretching.

4. Close any open figures.
5. Execute the following code to see histogram stretching on the `pout` image, which is already loaded in variable `I`.

```
img_limits = stretchlim(I);
I_stretch = imadjust(I,img_limits,[]);
figure
subplot(3,2,1), imshow(I), title('Original Image')
subplot(3,2,2), imhist(I), axis tight, ...
    title('Original Histogram')
subplot(3,2,3), imshow(I_stretch), ...
    title('Stretched Image')
subplot(3,2,4), imhist(I_stretch), axis tight, ...
    title('Stretched Histogram')
```



Question 4 How did the histogram change after the adjustment?

the histogram of the image becomes broader and more spread out, often stretching across the full intensity range (from 0 to 1)

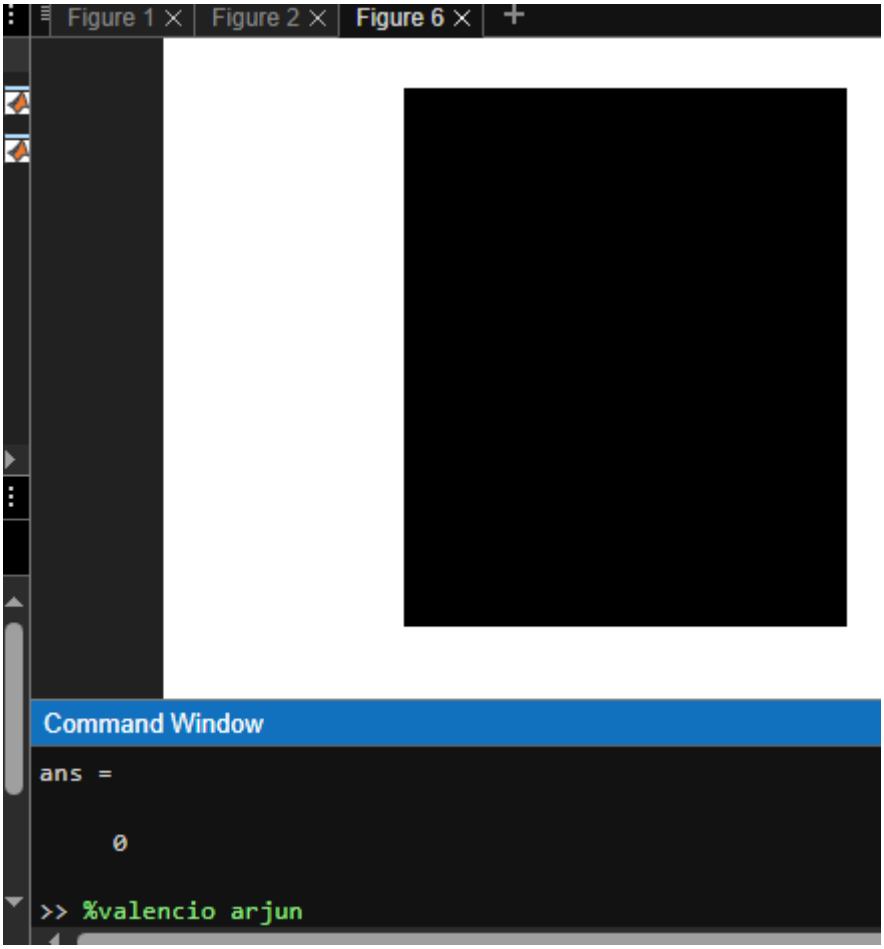
Question 5 What is the purpose of using the `stretchlim` function?

The stretchlim function computes intensity limits that exclude a small percentage of the darkest and brightest pixels typically the lower 1% and upper 1% to avoid the influence of outliers during contrast stretching.

In the previous step, we specified the `low_in, high_in, low_out`, and `high_out` parameters when calling the `imadjust` function when in fact the default operation is histogram stretching—meaning these parameters are not necessary to perform histogram stretching. Notice in the next step how just calling the function and only specifying the image as its parameter will give the same results.

6. Perform histogram stretching with `imadjust` using default parameters and confirm that the results are identical to the ones obtained before.

```
I_stretch2 = imadjust(I);
subplot(3,2,5), imshow(I_stretch2), ...
    title('Stretched Image')
subplot(3,2,6), imhist(I_stretch2), axis tight, ...
    title('Stretched Histogram')
I_stretch_diff = imabsdiff(I_stretch, I_stretch2);
figure, imshow(I_stretch_diff,[])
min(I_stretch_diff(:))
max(I_stretch_diff(:))
```



Question 6 How does the difference image look?

the difference image typically appears mostly dark or nearly black, indicating that the pixel differences are very small or zero across most of the image.

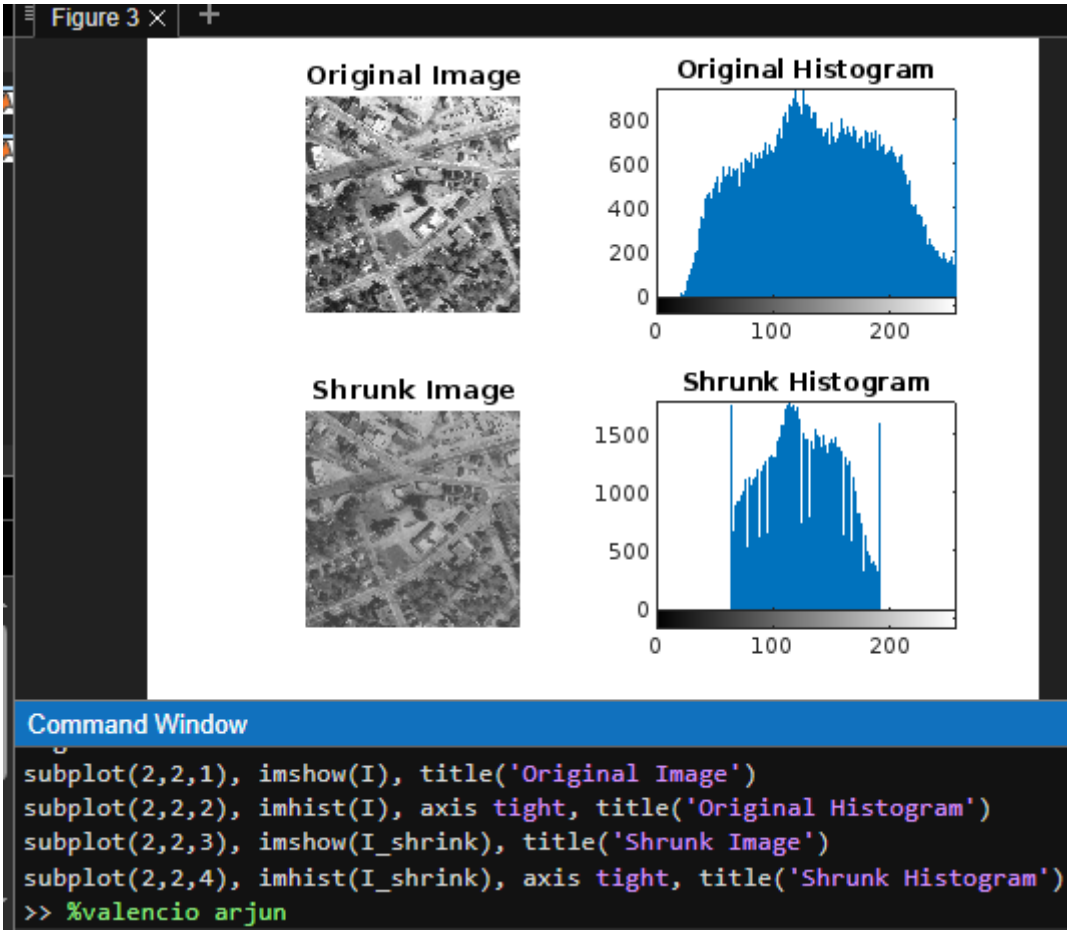
Question 7 What is the purpose of inspecting its maximum and minimum values?

Inspecting the minimum and maximum values of the difference image helps you quantify how different the two stretched images are.

To shrink an image histogram, we must specify the parameters explicitly.

7. Close any open figures and clear all workspace variables.
8. Execute the following code to see the result of histogram shrinking.

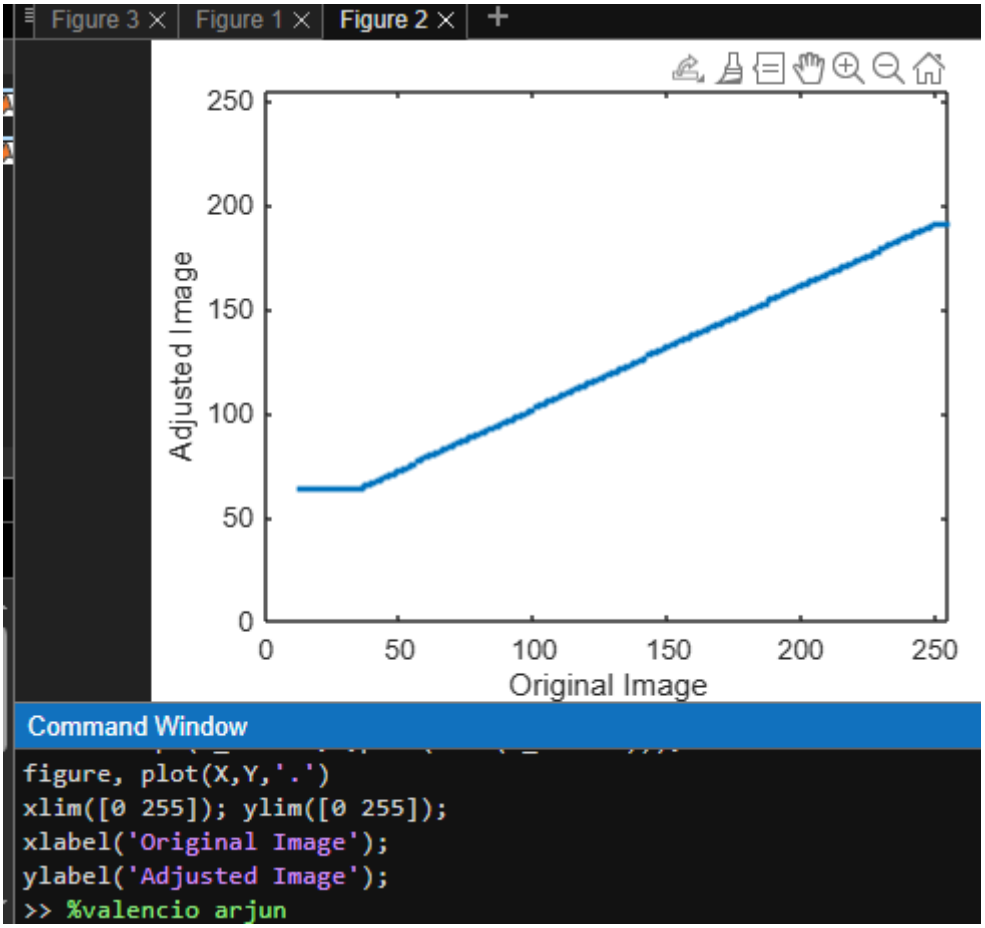
```
I = imread('westconcordorthophoto.png');
I_shrink = imadjust(I,stretchlim(I),[0.25 0.75]);

figure
subplot(2,2,1), imshow(I), title('Original Image')
subplot(2,2,2), imhist(I), axis tight, ...
    title('Original Histogram')
subplot(2,2,3), imshow(I_shrink), ...
    title('Shrunk Image')
subplot(2,2,4), imhist(I_shrink), axis tight, ...
    title('Shrunk Histogram')
```



When we use other techniques to adjust the histogram of an image, we have a means to view the transformation function (i.e., the `histeq` function will return the transformation function as an output parameter if requested). There is no built-in technique for viewing a transformation function when performing histogram sliding, stretching, or shrinking, but we can achieve a visual representation of the transformation function by using the `plot` function. To do so, we specify the original image as the X values and the adjusted image as the Y values.

9. Display the transformation function for the adjustment performed in the previous step.

```
X = reshape(I,1,prod(size(I)));
Y = reshape(I_shrink,1,prod(size(I_shrink)));
figure, plot(X,Y,'.')
xlim([0 255]); ylim([0 255]);
xlabel('Original Image');
ylabel('Adjusted Image');
```



Question 8 What do the above first two statements in the code do?

These two lines convert the 2D grayscale images I and I_shrink into 1D row vectors.

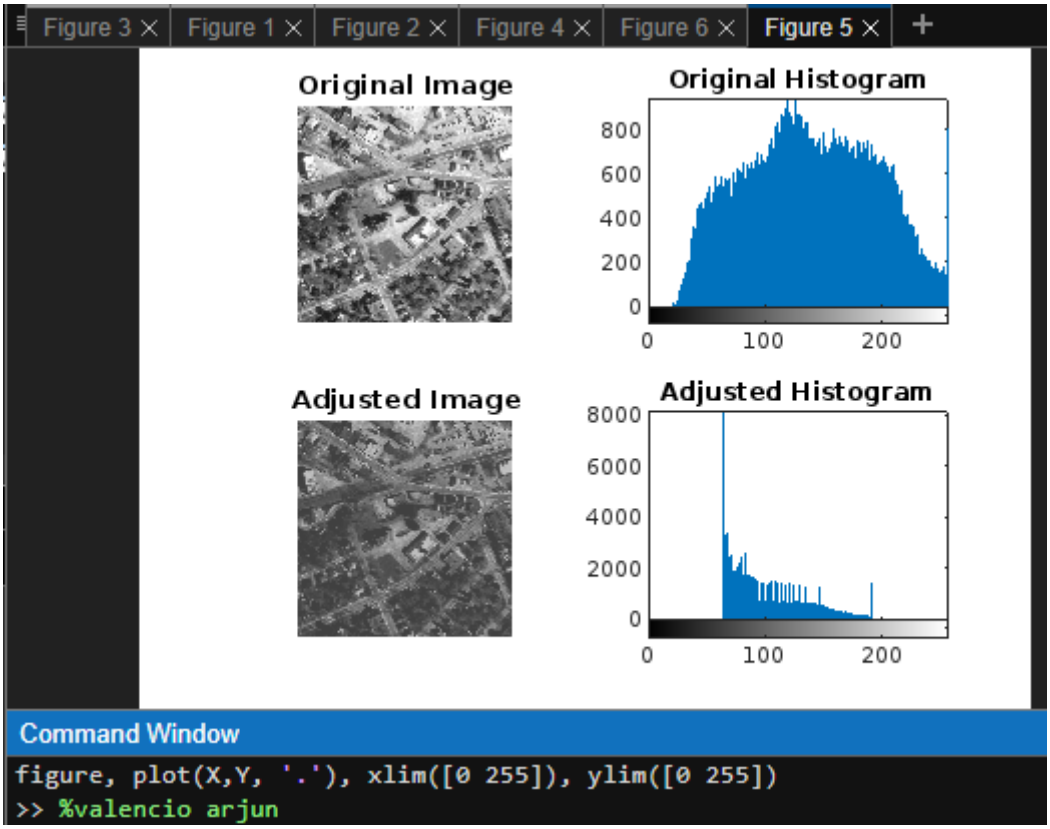
Question 9 What does the `xlabel` and `ylabel` functions do?

xlabel('Original Image') sets the label of the x-axis to "Original Image". ylabel ('Adjusted Image') sets the label of the y-axis to "Adjusted Image".

As noted earlier, gamma values other than 1 will specify the shape of the curve, toward either the bright or the dark region.

10. Close any open figures.
11. Perform histogram shrinking with a gamma value of 2.

```
I_shrink = imadjust(I,stretchlim(I),[0.25 0.75],2);
X = reshape(I,1,prod(size(I)));
Y = reshape(I_shrink,1,prod(size(I_shrink)));
figure
subplot(2,2,1), imshow(I), title('Original Image')
subplot(2,2,2), imhist(I), axis tight, ...
    title('Original Histogram')
subplot(2,2,3), imshow(I_shrink), title('Adjusted Image')
subplot(2,2,4), imhist(I_shrink), axis tight, ...
    title('Adjusted Histogram')
figure, plot(X,Y,'.'), xlim([0 255]), ylim([0 255])
```



Question 10 The transformation function plot displays a gap from 0 to 12 (on the X axis) where there are no points. Why is this so?

because there are no pixels in the original image I with intensity values between 0 and 12.