

---

# Table of Contents

Fresh Start .....	1
Specification .....	1
Code .....	2

## Fresh Start

Author: Evan Krimpenfort

Class: ECE 563-01

Purpose: Histogram equalization through the lens of linear and nonlinear transforms.

```
clc; clear all; close all;
```

## Specification

For this project assignment, you are to use a linear contrast stretch and also use histogram equalization<sup>1</sup> to improve the display of a grayscale image with a poor histogram. You are to write and execute a well-commented MATLAB script file to perform the following tasks on the image “Unequalized\_Hawkes\_Bay\_NZ.jpg” found on Isidore under Resources/Data.

1. Load the image with `x = imread()`. Extract green and make double: `x = double(x(:,:,2))`.
2. Display the grayscale image `x` with `imshow(x, [0,255])` in a new figure window. Add an informative title to the figure with the `title()` command.
3. Display the histogram of this original image in a new figure window using `hist(x(:), 256)`. Add an informative title to the figure.
4. Create and apply a linear transformation that better spreads the histogram across the `[0,255]` dynamic range that is to be displayed. Save the modified image to a new variable `y`. Refer to the notes for methods that use the `mean()` and `std()` for help.
5. Display the linearly transformed image with `imshow(y, [0,255])` in a new figure window. Add an informative title.
6. Display the histogram of the linearly transformed image in a new figure window using `hist(y(:), 256)`. Add an informative title.
7. Create a new figure that shows the linear transformation mapping of input values (on the horizontal axis) to output values (on the vertical axis). Label the axes appropriately and give the plot an informative title. A simple way to do this is to use a command like this: `plot(x(:), y(:), 'b.')`, where `x` is the input image and `y` is the output image.
8. Now apply histogram equalization to the ORIGINAL image and save this with a new variable name `z`. Use the function I provided `z = myhisteq( x, 256 )`.
9. Display the linearly transformed image with `imshow(z, [0,255])` in a new figure window. Add an informative title to the figure with the `title()` command.
10. Display the histogram of the histogram equalized image in a new figure window using `hist(z(:), 256)`. Add an informative title to the figure with the `title()` command.

---

11. Create a new figure that shows the nonlinear histogram equalization transformation mapping of input values to output values as in Step 7 (except using  $x$  and  $z$ ).

12. Publish your script file to a .pdf file and upload the .pdf file to Isidore. Upload only this .pdf file, do not upload any images. The script must be well commented and include line comments (%) and multiple cells (%%).

## Code

```
% Start of hmwk3.m

% 1. read, extract the green, and make x a double.
x = imread('Unequalized_Hawkes_Bay_NZ.jpg');
x2 = double(x(:,:,2));

% 2. Display the grayscale image.
figure(1);
imshow(x2, [0, 255]); drawnow;
title('original image');

% 3. Display the histogram of x2 and give a title.
figure(2);
histogram(x2(:), 256); drawnow;
title('histogram of the original image');

% 4. apply the linear transformation. used both the mean() and std()
% funtion. mess with the gain and bias.
y = (x2 - mean(x2(:))) / std(x2(:));
y = (y * 64) + 127;

% 5. display the linearized image.
figure(3);
imshow(y, [0, 255]); drawnow;
title('linearized image');

% 6. display the histogram of the linearized image.
figure(4);
histogram(y(:), 256); drawnow;
title('histogram of the linearized image');

% 7. comparison between the input of the image and the output.
figure(5);
plot(x2(:), y(:), 'b. ');
xlabel('input values');
ylabel('output values');
title('linear input v. output');

% 8. histogram equalization with myhisteq(..)
z = myhisteq(x2, 256);

% 9. display the histeq'd image.
figure(6);
imshow(z, [0, 255]); drawnow;
```

---

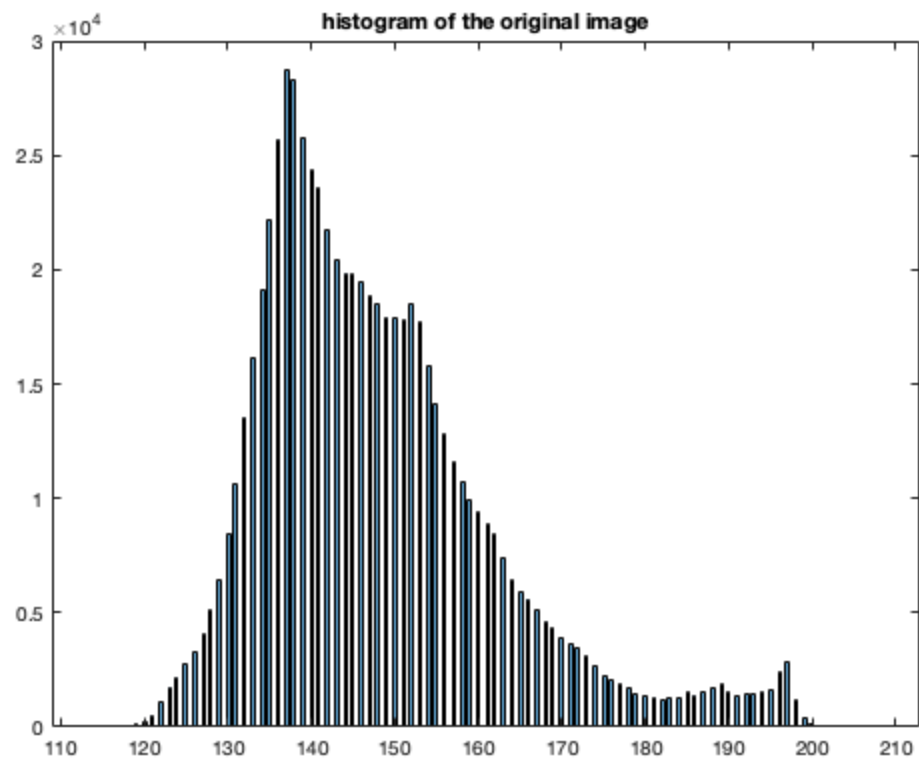
```
title('histeq image');

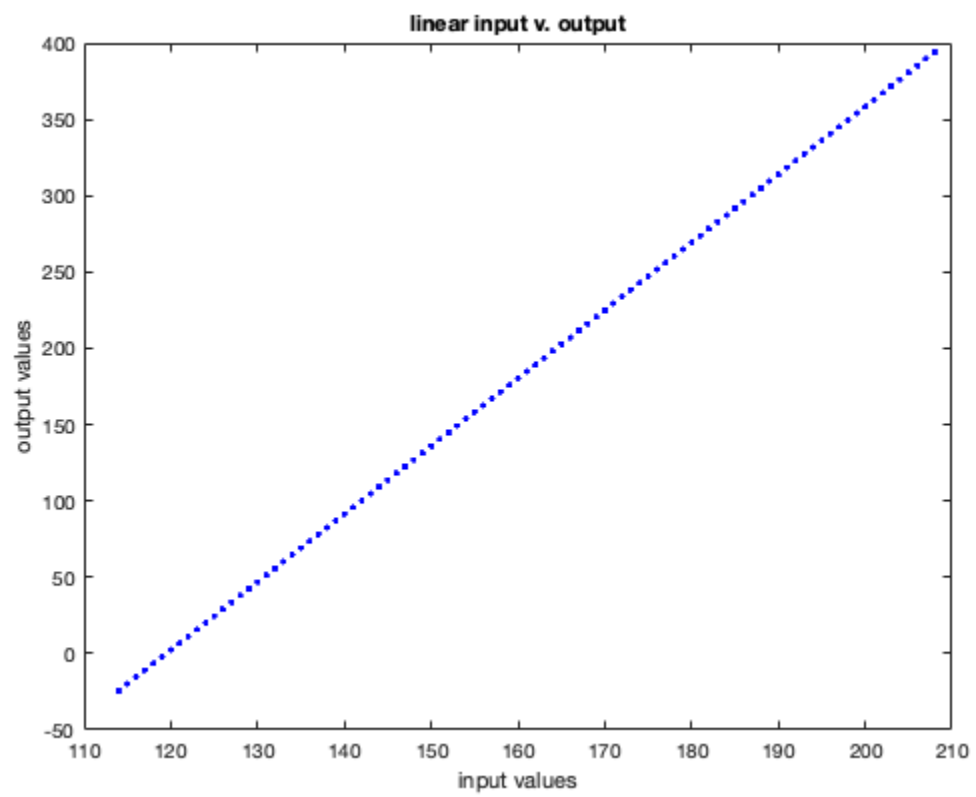
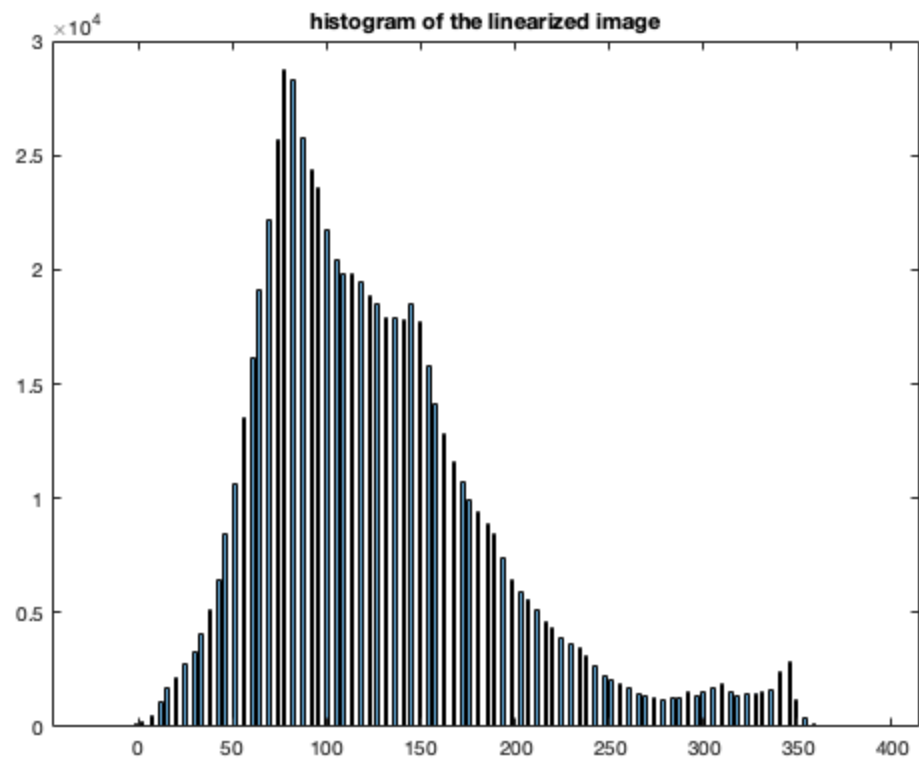
% 10. display the histogram of the histeq'd image
figure(7);
histogram(z(:), 256); drawnow;
title('histogram of the histeq image');

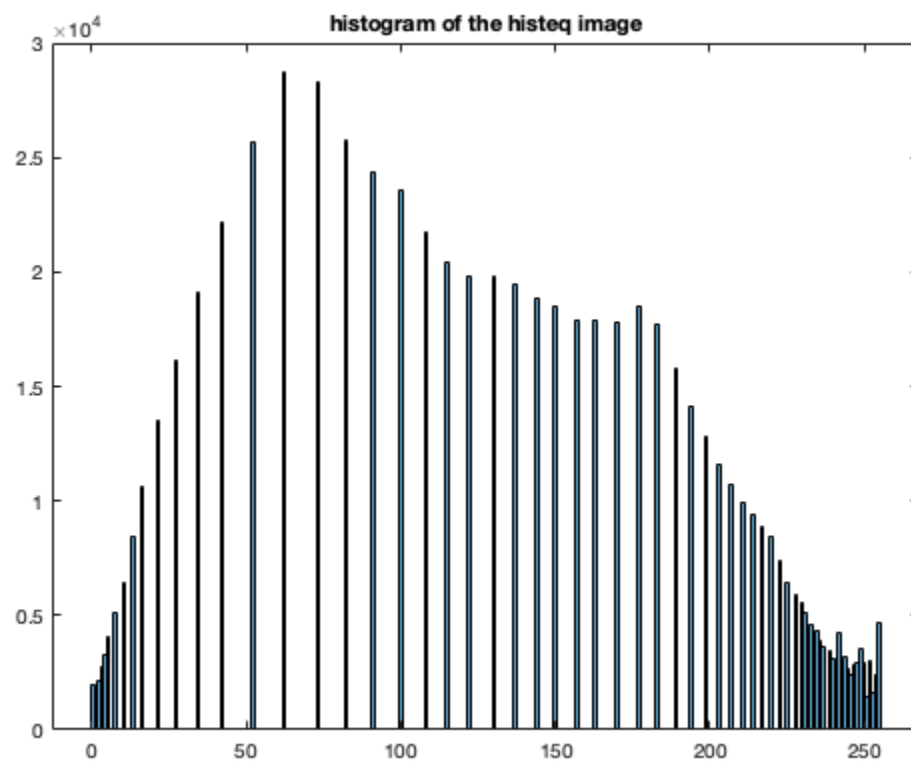
% 11. comparison between the input of the image and the different
      output.
figure(8);
plot(x2(:), z(:), 'b. ');
xlabel('input values');
ylabel('output values');
title('nonlinear input v. output');

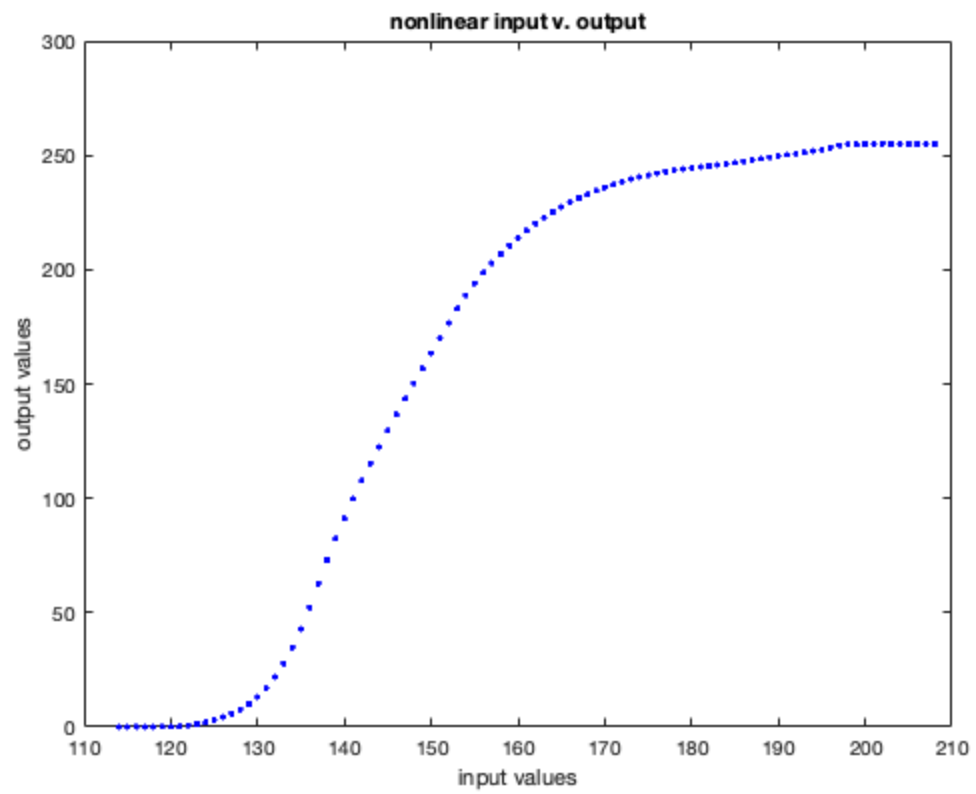
% end of hmwk3.m
```











*Published with MATLAB® R2020a*