Brandon Chin
ID(last four digits): 2938 CSc59866 - Capstone 2015

Assignment 1

Writing Assignments

1. How does an image change (e.g., objects' size in the image, field of view, etc.) if the focal length of a pinhole camera is varied?

The smaller the hole, the sharper the image resolution, but at some point, the image edges do get dimmer. A hole that is too small begins to produce diffracted images due to the wave properties of light.

As the focal length is varied, the image size changes linearly, such that as the length increases, the image gets larger, and vice versa. The field of view (FOV) increases quadratically as the focal length shortens.

2. Give an intuitive explanation of the reason why a pinhole camera has an infinite depth of field.

A pinhole camera has an infinite depth of field because, as stated above, the smaller the hole, the sharper the image. The hole is as small as physically possible such that the light rays coming in are minimized, organized, and thus almost everything is in focus.

3. Prove that, in the pinhole camera model, three collinear points (i.e., they lie on a line) in 3D space are imaged into three collinear points on the image plane. You may either use geometric reasoning (with line drawings) or algebra proof (using equations).

Given the points on a straight line are imaged to a straight line, the same is true for collinear points. Note that the area of a triangle determined by three points will be zero if they are collinear. In other words, three collinear points form a matrix whose determinant is zero.

$$Det \begin{vmatrix} X1 & Y1 & Z1 \\ X2 & Y2 & Z2 \\ X3 & Y3 & Z3 \end{vmatrix} = 0$$

Now we can apply the projection of the image to be equal to x = -f(X/Z) and y = -f(Y/Z).

$$Det \begin{vmatrix} (X1/Z1) & (Y1/Z1) & 1 \\ (X2/Z2) & (Y2/Z2) & 1 \\ (X3/Z3) & (Y3/Z3) & 1 \end{vmatrix} = 0$$

This shows that the projected points are collinear and the determinant is zero.

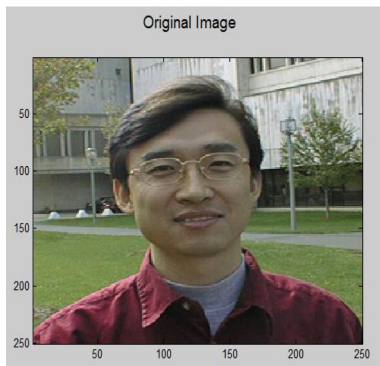
Brandon Chin CSc59866 - Capstone 2015
ID(last four digits): 2938

Programming Assignments

1. Image Formation.

a. Read in a color image C1(x,y) = [R(x,y),G(x,y),B(x,y)] in Windows BMP format and display it.

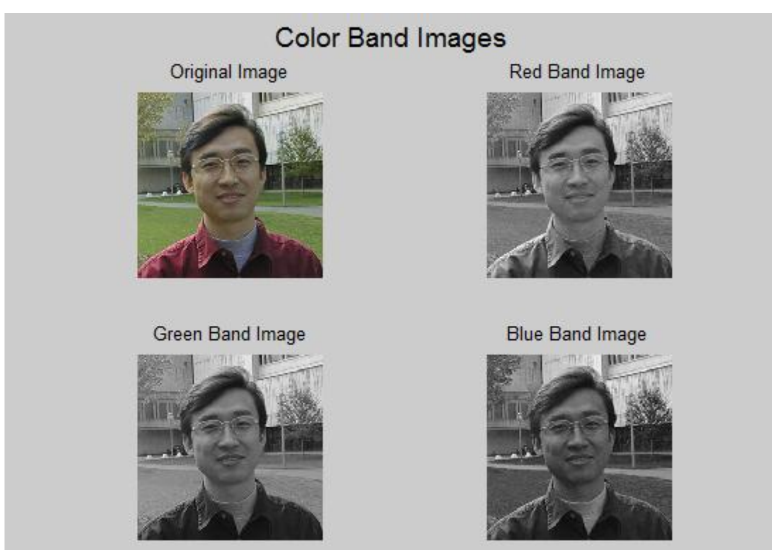
```
InputImage = 'IDPicture.bmp';

C1 = imread(InputImage);    % C1 is an array containing the image data
[ROWS, COLS, CHANNELS] = size(C1);

No1 = figure;
image(C1);  % display the image from array C1
suptitle('Original Image');
```



b. Display the images of the three color components, R(x,y), G(x,y), and B(x,y), separately. You should display three grayscale-like images.

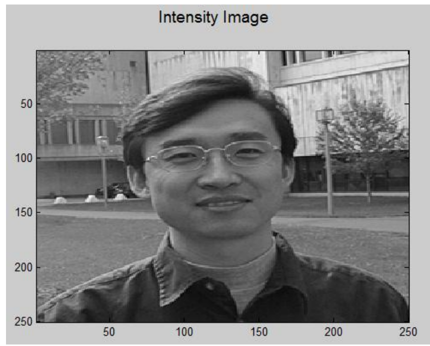
First, we created three blank images for each color band, red, green, and blue.

Then, for each blank image, we extracted the corresponding color band data from the original color image into each of the three bands of the blank image to create a grayscale image.



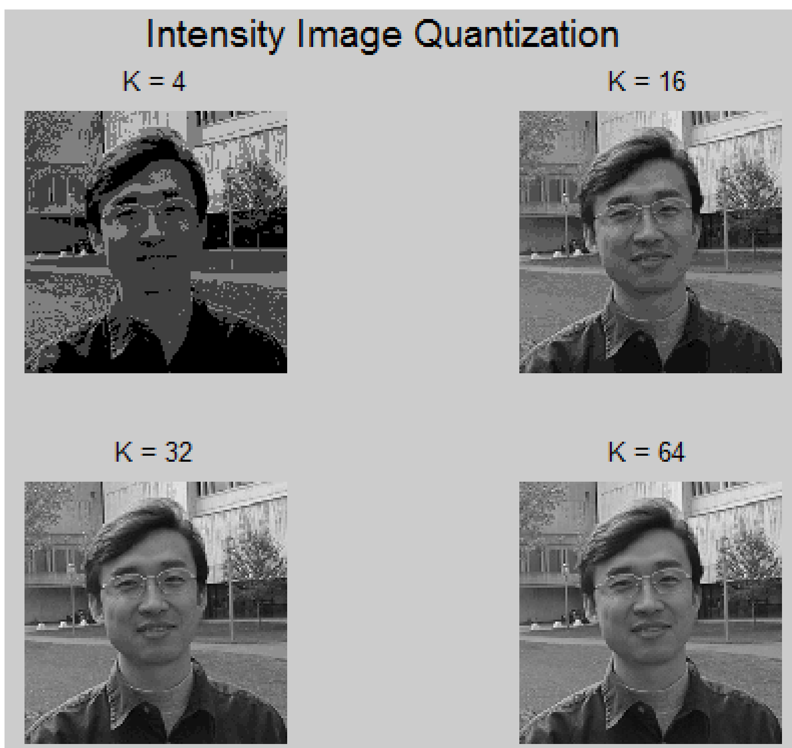
c. Generate an intensity image I(x,y) and display it. You should use the equation I = 0.299R + 0.587G + 0.114B (the NTSC standard for luminance).

To produce this intensity image, we assigned to each band of the image the same value (computed by the above equation), which generated a grayscale-like image.



d. Uniformly quantize the original intensity image into K levels (with K=4,16, 32, 64). Display the four quantized images with four different K levels.

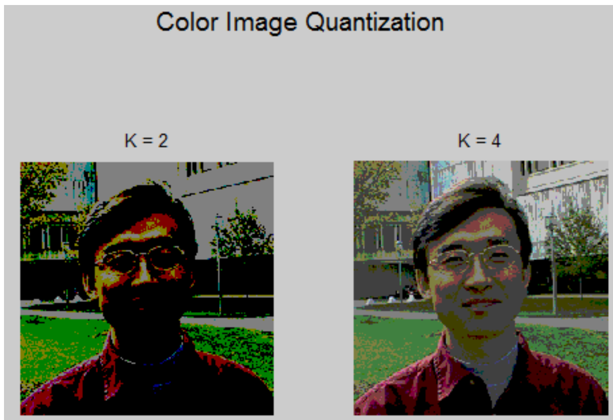
I observed that as the K levels increase, the detail of the image also increased. As K approached 64, it began to look like the original intensity image. The reason for this is because at low K levels, the generator is unable to distinguish all of the pixel values and ends up rounding a value. Analogous to how rounding a number can result in losing precision.



e. Quantize the original three-band color image into K level color images with uniform intervals and display it.

This is similar to the last exercise except now we are producing a color image rather than an intensity image.

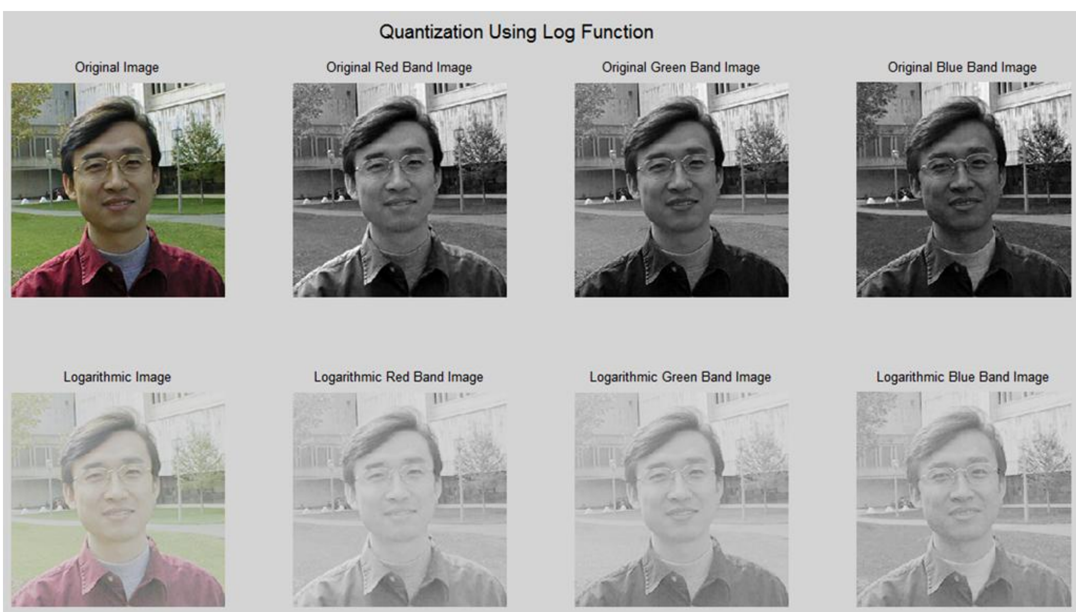
When considering a quantized image, it is usually worse for viewing since the level of detail is decreased, however, for computer processing, the image has a smaller variance of pixel values, and therefore easier to process.



f. Quantize the original three-band color image into a color image with a logarithmic function and display it.

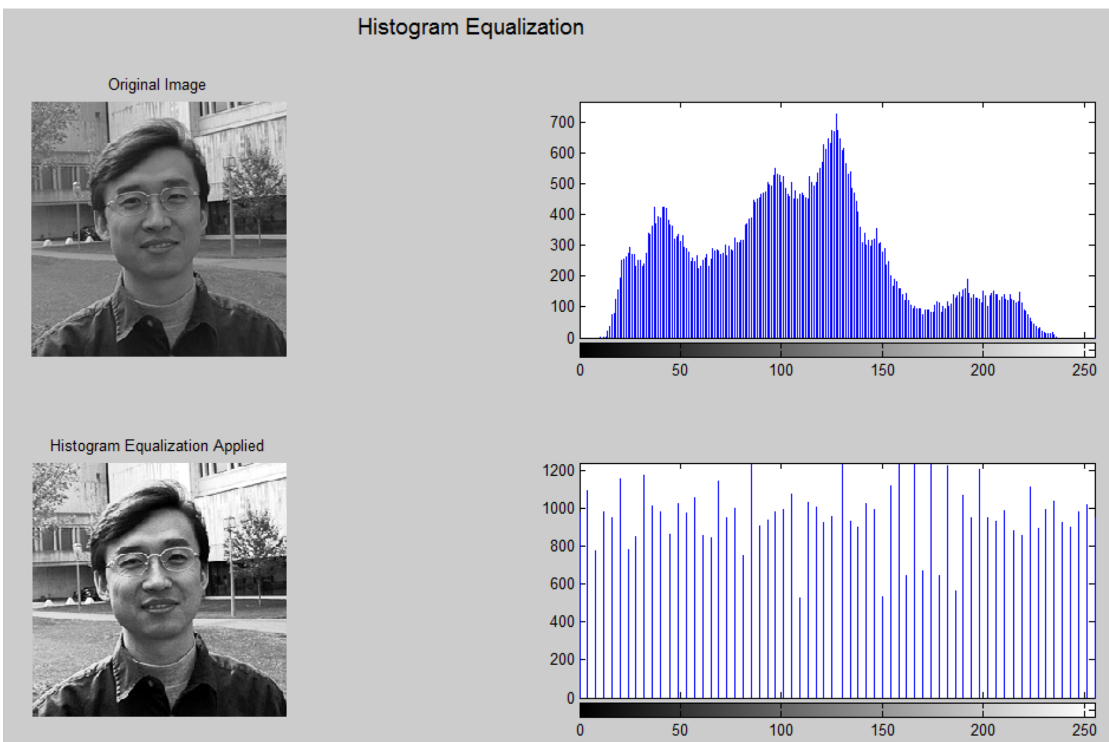
On the top row, we have the original image and its color bands. On the bottom row, we have the same images after a logarithmic function has been applied to it.

The best constant value to scale into the pixel data I found was 0.15. The higher the value of this constant, the darker the image, and the lower the value, the lighter the image.



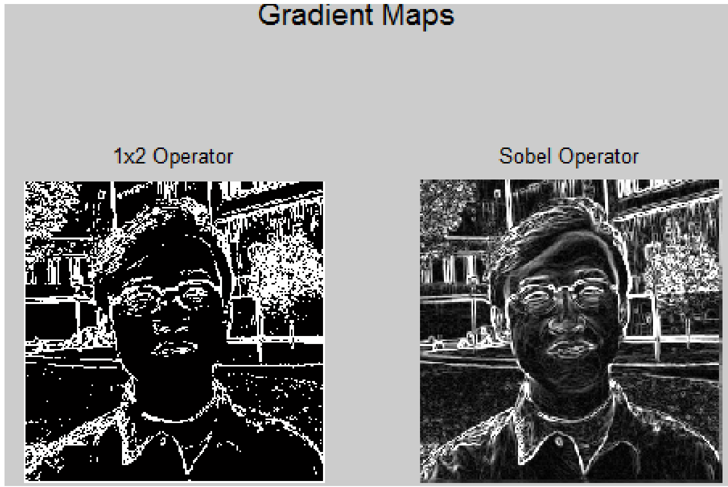
2. Generate the histogram of the intensity image and display it. The perform a number of histogram operations to make the image better for either viewing or processing. Also display the histogram and make observations of the image based on its histogram. What are the general distributions of the intensity image? How many major peaks and valleys does your histogram have? How could you use the histogram to understand, analyze, or segment the image?

This is a threshold image, also known as a binary image. This means the values are distributed at either 0 or 255, and is determined based on some threshold value. The threshold value I chose was 100. This is a simple form of image segmentation.

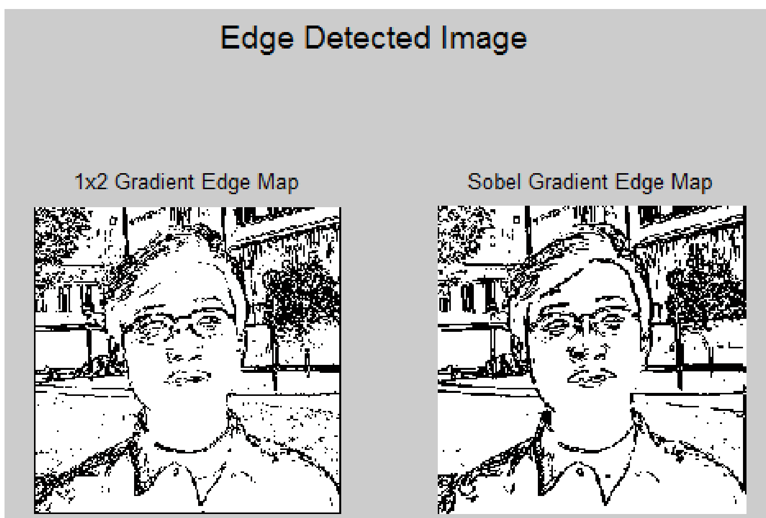


This is a method called histogram equalization, which modifies the contrast of the image using the image's histogram in order to enhance the visual quality of the image. This method works by reading the original image's histogram, and attempting to normalize the distribution of pixel values. The original histogram has about 5 or 6 major peaks and about 4 or 5 major valleys.

3. Apply the 1x2 operator and Sobel operator to your original intensity image and analyze the results of the gradient magnitude images. Generate edge maps of the two gradient maps. An edge map should be a binary image with 1s as edge points and 0s as non-edge points.



These gradient images use edge detection algorithms to produce an image which emphasizes edges.



These are the corresponding edge maps of the above gradient maps. These are binary images which outline the edges of the image based off of some threshold value.