# Lab Report 4 - Image Processing

ECE 3101L Signals and Systems

Section 3

Edward Enriquez

5/15/2021

# Prelab/Background Theory

## Resizing Images

Downsizing an image in MATLAB and in most other softwares is done by averaging a certain number of neighboring pixels. To re-enlarge an image some sort of interpolation must be done. For interpolation methods it is common to map the desired enlarged image pixel position to the smaller input image. Nearest neighbor interpolation simply assigns the value of the mapped pixel’s nearest neighbor to the output pixel. Bilinear interpolation takes the weighted average of a mapped pixel's nearest 2x2 neighborhood and bicubic interpolation takes the weighted average of a mapped pixel's nearest 4x4 neighborhood. Each software calculates its weights differently but most base their weights on row/column of output pixel and width/height scale factor.

## Filtering Images

Image filters make use of specific matrices called kernels. These kernels are convoluted with the image’s pixels to achieve a certain effect. Highpass filters remove low spatial frequencies and make an image sharper while lowpass filters remove high spatial frequencies and make an image blurred. There are also filters which can shift images by single pixels.

## Enhancing Images

The enhancement of an image is done by combining a filtered image with its original version. This lessens the effects of the filtering giving it more aesthetic appeal.

## Histograms/Cumulative Sums

Histograms are a type of graph which tally the amount of pixels based on their intensity. The tally is represented as a bar above the specified intensity. The intensity is represented by an 8 bit number so there are 256 bars in our histograms.

The cumulative sum graph shows the number of pixels which are at or below a certain intensity. These graphs have been normalized so that the total sum of all the pixels is one. This is another way to visualize which intensity is more abundant in the image.

Histograms could be equalized to enhance the contrast and brightness of an image. Equalization transforms the pixel intensity values in a way that redistributes them across the entire intensity range. The process involves calculating a cumulative distribution function (CDF) and mapping the original intensity values to new values based on the CDF. This mapping spreads out the pixel values, making the histogram more uniform and ultimately results in an image with improved contrast.

**Two-Dimensional Discrete Fourier Transform**

Considering an image as a function, the image's domain is the position of a pixel, both its x and y coordinate. The co-domain or 'function value' is the pixel intensity. When a fourier transform is performed, the image is transferred into the frequency domain. The domain is now horizontal and vertical spatial frequency, and the function value represents frequency magnitude. Since the frequency domain is usually in the range of multiple orders of magnitude, it is typical to plot the Fourier transformed function in a log scale.

**Filtering in Frequency Domain (Gaussian Lowpass and Highpass)**

Gaussian filters are an alternative way of filtering an image which is done in the frequency domain. The Gaussian lowpass filter is a 2 dimensional Gaussian distribution. The standard deviation of the distribution ($\sigma$) is related to the cutoff frequency of the filter, specifically, as standard deviation of the distribution increases, so does the cutoff frequency. A high pass filter can be created by subtracting the lowpass Gaussian filter from 1. The image is simply multiplied by the Gaussian distribution to remove the specified frequency content. Then the resulting filtered image can be retrieved by using the inverse Fourier Transform.

# Prelab 11

If an image has a resolution of $300\ \frac{pixels}{inch}$ and is $5''x7''$, then the number of pixels in the image is:

$$300\ \frac{pixels\ lengths}{inch} \times 5\ inch \times 300\ \frac{pixel\ lengths}{inch} \times 7\ inch = 3150000\ pixels$$

The image has 256 $\frac{levels}{pixels}$ or 8 $\frac{bits}{pixel}$ which mean for the image's total number of bits is:

$$3150000\ pixels \times 8\frac{bits}{pixel} = 25200000\ bits$$

With a compression ratio of 2.265:

$$\frac{25200000\ bits}{2.265} = 9600000\ bits$$

The number of 8 bit packets will be:

$$Number\ of\ Packets\ = \frac{9600000\ bits}{8} = 1200000\ packets$$

Accounting for the starting and ending bits of each packet, the total number of bits to send is:

$$N = 1200000\ bits\ \times 10 = 12000000\ bits$$

With a 9.6K baud modem the time needed to transmit the image will be

$$\frac{12000000\ bits}{9600\ baud} = 1250\ seconds$$

With a 56K baud modem the time needed to transmit the image will be

$$\frac{12000000\ bits}{65000\ baud} = 184.615384615\ seconds$$

With a 750K baud modem the time needed to transmit the image will be

$$\frac{12000000\ bits}{750000\ baud} = 16\ seconds$$

With a 100M baud modem the time needed to transmit the image will be

$$\frac{12000000\ bits}{1000000000\ baud} =.12\ seconds$$

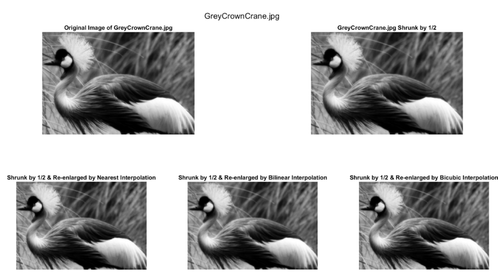
# Lab

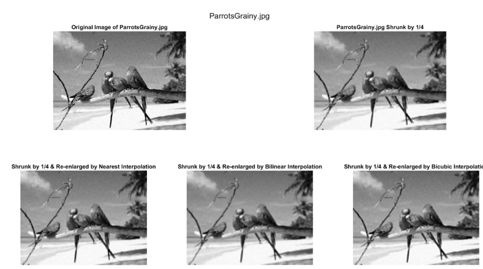
**Lab Procedures**

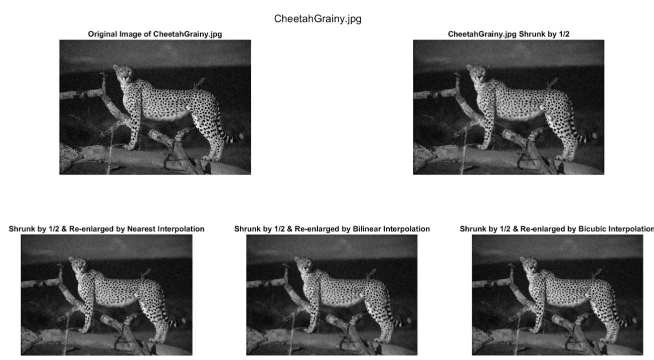
- Downsize each given image by a ½, ¼ and ⅛ . Re-enlarge each downsized image to its original size using nearest neighbor, bilinear, and bicubic interpolation.
- Apply the 9 given image filter kernels to the GreyCrownCrane.jpg and GreyCrownCraneGrainy.jpg
- Plot histograms and cumulative sums for the given images and their equalized counterparts.
- Transfer given images to log scale frequency domain and display.
- Apply Gaussian lowpass and highpass filter to GreyCrownCrane.jpg in frequency domain. Revert back to the original image and display.
- Analyze and Discuss Results

**Lab Results**

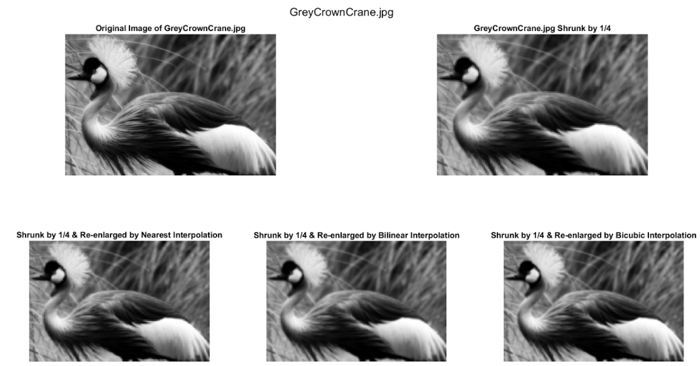
## Part 1: Resizing Image

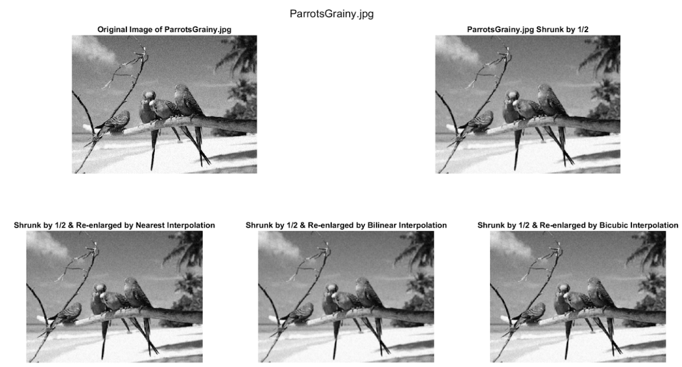


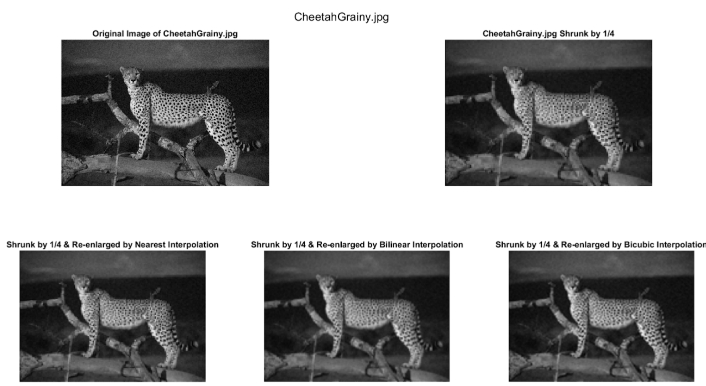




When reduced to half their original size and re-enlarged the visual quality of the images is almost the same as the original image.The images which are being reconstructed do not show any noticeable difference based on the interpolation method. This is because only 1 of every two pixels is being reconstructed using its surrounding pixels which does not lower the quality of image much.

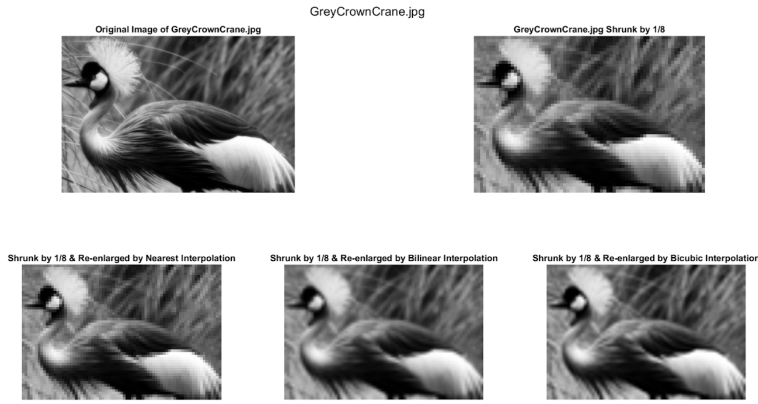


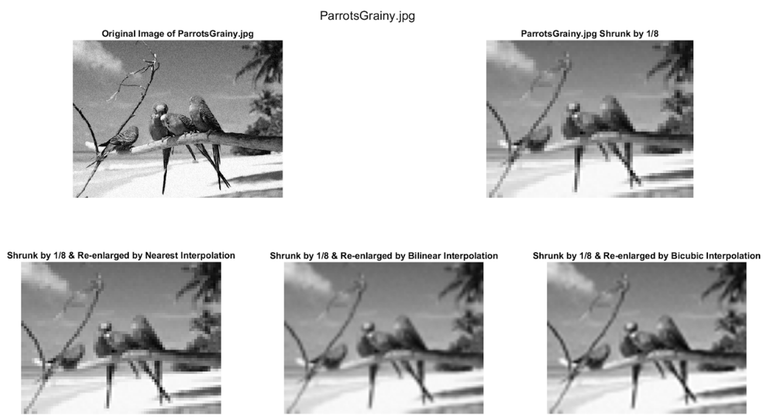



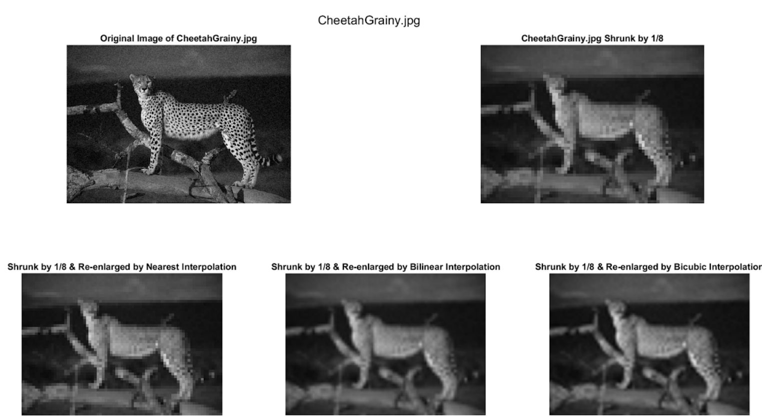




When the images are shrunk to a smaller size the re-enlarged images become more noticeably pixelated or blurred. This is because larger amounts of pixels are losing their original value and being replaced with averages or nearest neighbors. Not only are there larger areas of pixels which do not have their original intensity, but it is more unlikely for a pixel to end up with a shade which it started with because now the neighboring pixels from which the interpolation gets its information from also have slightly different values.

It is also more noticeable now that the nearest neighbor interpolation produces a pixelated image while the other interpolations create a blurred image. This is because the nearest neighbor interpolation method only takes into account the intensity of a specific pixel without taking into account all the other pixels around it which might more accurately represent the true intensity of that pixel. The image is partitions of chunks of pixels of the same intensity rather than a more fluid change in color.



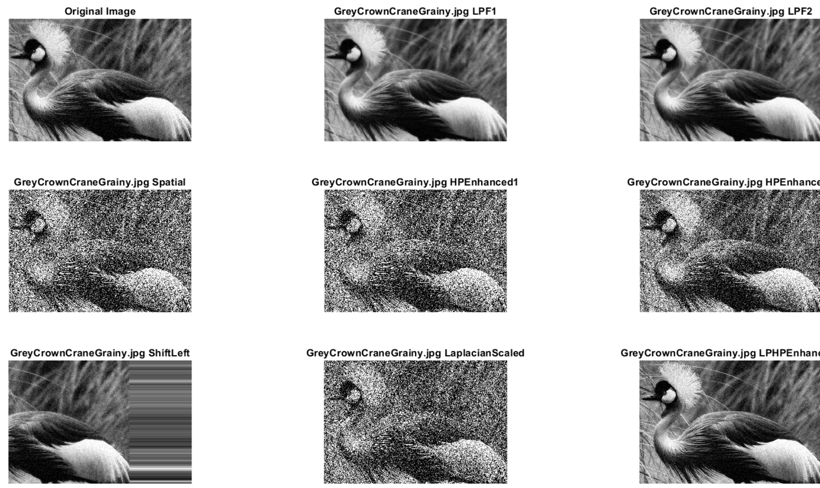




It becomes clear that the shrunk image is identical to the enlarged image using nearest interpolation. The difference is that in the shrunk image, the pixelated chunks are actually single pixels while in the re-enlarged image the pixelated chunks are groups of pixels with the same intensity. They only look the same because they are both made to be the same size on MATLAB.

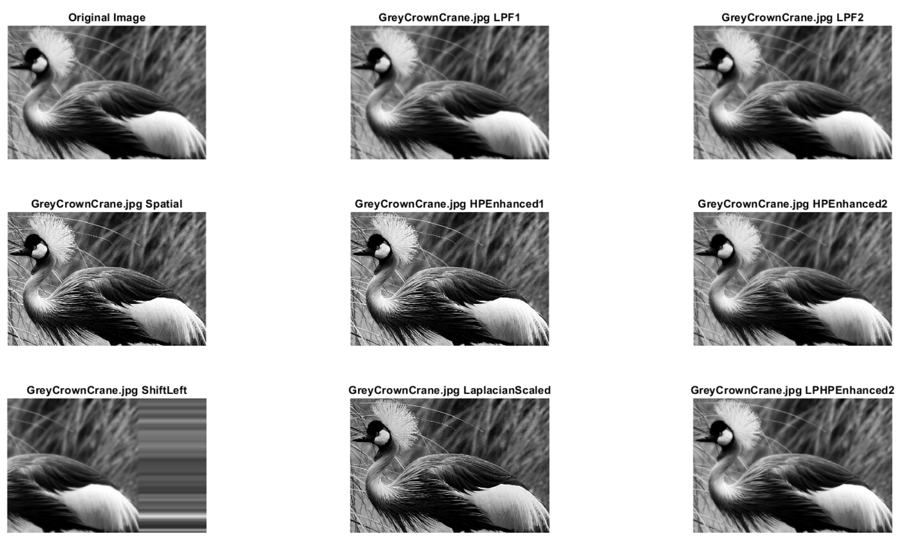
# Part 2: Filtering an Image

## Different Filtering Methods Applied to *GreyCrownCraneGrainy.jpg*




The results show that the lowpass filter indeed makes an image more blurry and sort of smooths out contrast while high pass filters intensify the contrast.  The high pass enhanced image as expected looks like a combination of the high pass filtered image and the original. The shift left kernel also was successful in shifting the images

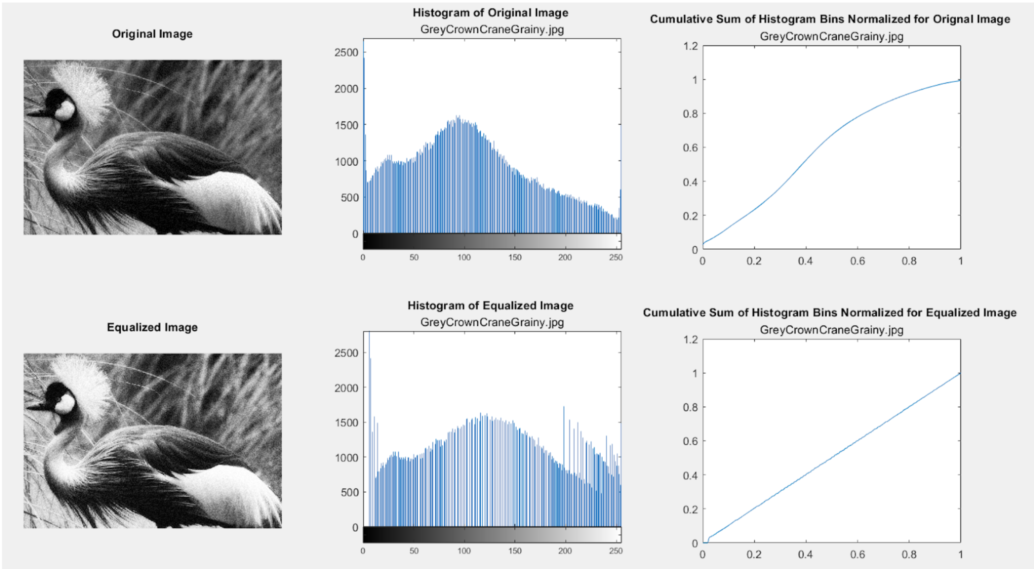
## Different Filtering Methods Applied to *GreyCrownCrane.jpg*




With this original image of the Grey Crowned Crane, the high pass filtered image does not seem as distorted as they did with the grainy photo. Although the grains were not as visible, in the original GreyCrownCraneGrainy.jpg, the high pass filters brought them out with high levels of contrast. Now when the original image is high pass filtered, only the edges of the image become sharper and the ‘static’ does not take over the picture.

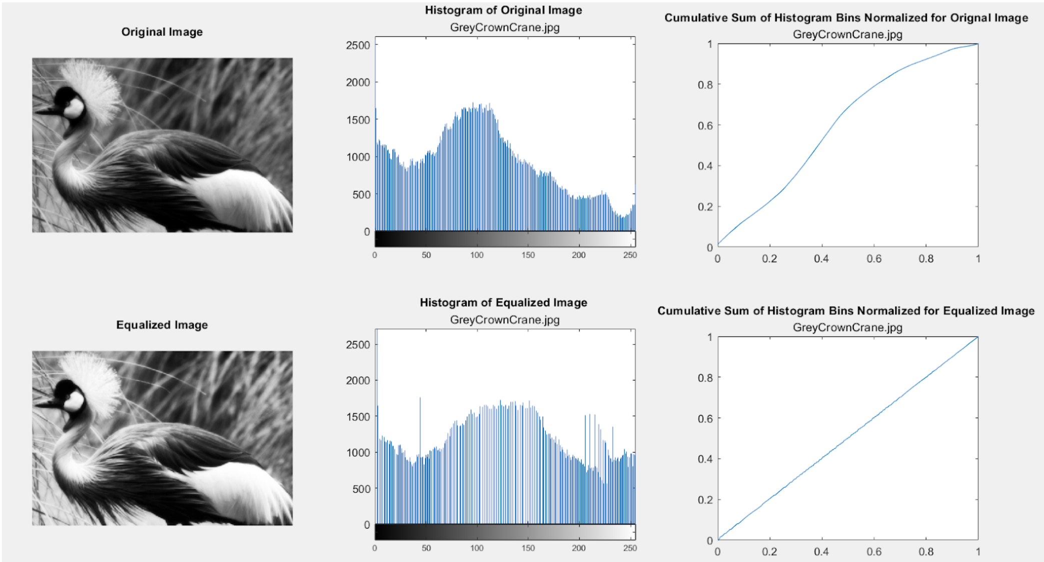
## Part 3: Enhance Contrast using Histogram Equalization

### Histogram and Cumulative Sum Graph of *GreyCrownCraneGrainy.jpg*



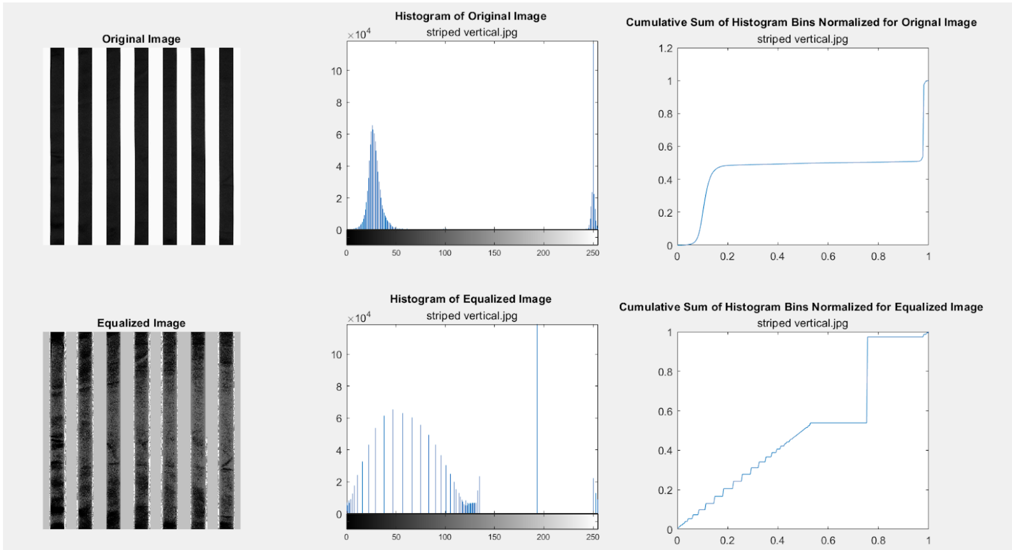

Looking at the histogram of the original image, it is clear that the GreyCrownCraneGrainy.jpg has a lesser amount of high intensity pixels. When the image is equalized, the histogram shows the added high intensity pixels to make the image more balanced. The cumulative sum graph for the equalized image shows that the average amount of pixels around a certain intensity is constant throughout the intensity spectrum.

### Histogram and Cumulative Sum Graph of *GreyCrownCrane.jpg*




Again, the histogram of the original image is lacking in high intensity pixels but there is also a dip in the amount of pixels at around an intensity of 30-40. In the equalized image more pixels of these intensities are added back to make the cumulative sum a constantly increasing line.

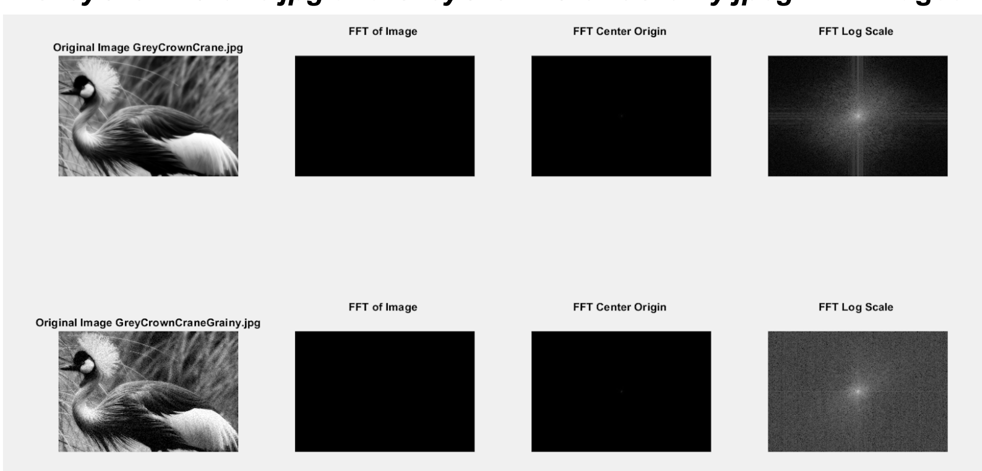
**Histogram and Cumulative Sum Graph of *stripe_vertical.jpg***




The stripe vertical image had a concentrated amount of pixels of two specific intensities. This is evident in the histogram of the original. Both of these concentrated intensities follow a normal distribution. The equalizing of the image was not as successful as it was with the previous images. Since there are only a few pixel intensities with high frequencies, the equalization could not distribute the pixels far enough to fill the gap of under represented pixels in the mid-range of intensity.

## Part 4: Two-Dimensional Discrete Fourier Transform

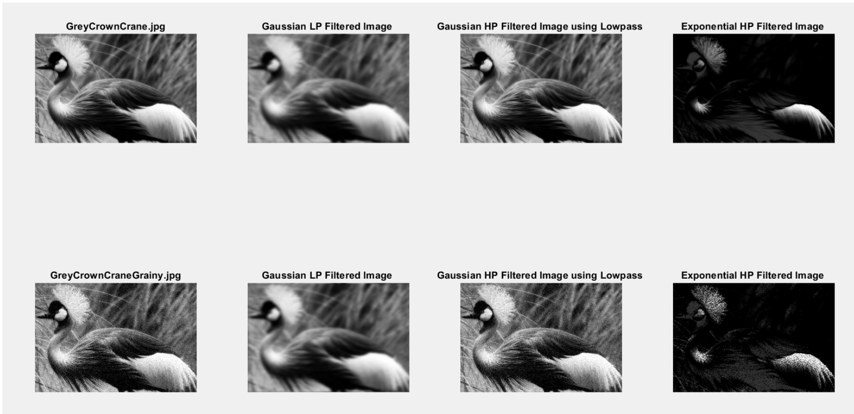
***GreyCrownCrane.jpg* and *GreyCrownCraneGrainy.jpeg* FFT Images**




The FFT transformed image of the grainy image seems to have more pixels of relatively high intensity (lighter pixels). This is indicative of higher frequency content in the grainy image. This makes sense as the grains in the original image cause quick changes in intensity which represent high frequencies. With the grains, parts of the image which would otherwise not change in intensity (have low frequency) now have multiple quick changes from one intensity to another (high frequency).

## Part 5: Filtering in Frequency Domain (Gaussian Lowpass and Highpass)

**Gaussian Filtered *GreyCrownCrane.jpg* and *GreyCrownCraneGrainy.jpg* images**




Part 5 produces similar results to that of part 2, but the low pass filtered image now seems to be even more blurred and the contrast of the high pass filter does not seem as extreme. These are the benefits of having a single variable on which your filter is based on. It is easier to change how strong the filter's effect is. The exponential of the high pass image is simply the high pass filtered image raised to a power of 4 when the pixel intensities are represented as doubles less than 1. The contrast of the image is extremely high and lower intensity pixels blend in while the higher intensity pixels pop.