SYDE 575
Lab 3 - Report

Prepared by:
Timothy Tsang 20556306
Jonathan Wen 20515413

Professor David A. Clausi

Wednesday October 17th, 2018

1. Introduction

This lab investigates how the process of image restoration within the frequency domain. Specifically, the use of fourier transform to perform image restoration, along with the use of various types of filters was completed. First, the use of fourier transform to display the fourier spectra of an image was done, along with looking at the phase and magnitude components. Next, several noise reduction techniques were investigated, using various filters such as an ideal low pass filter with various cutoff radiuses, and a Gaussian low pass filter. Lastly, a filter design implementation was done on an input image that had an unknown noise added to itself. A filter was designed and implemented to remove the noise and restore the image to its original state.

2. Fourier Analysis

An image of a white rectangle (pixel intensity of 0) was created and used as the test image. The image and it's fourier spectra was plotted. The fourier spectra was calculated by applying the fourier transform to the test image, shifting the transform, and then plotting the spectra. The shift must be done in order to center the fourier transform spectra into the center of the plot. Below shows the test image and it's corresponding fourier spectra.

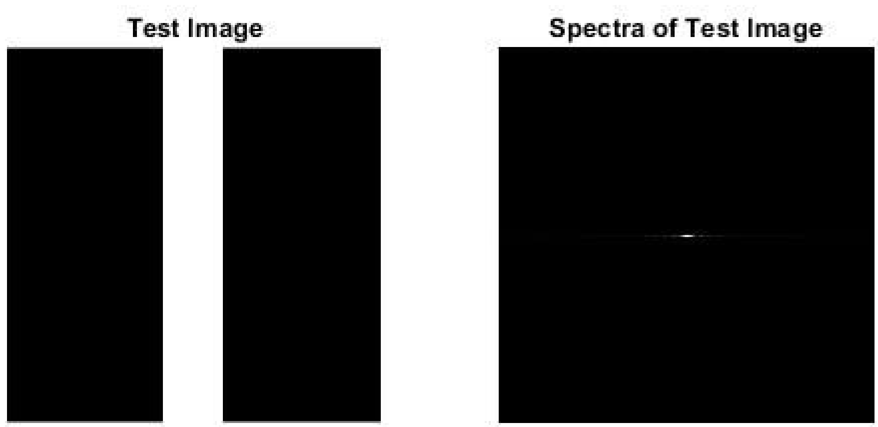
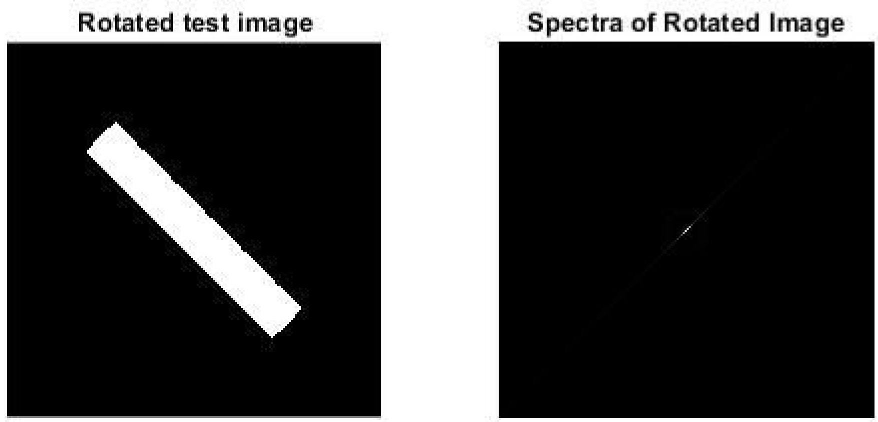


Figure #1 - Fourier spectra of Test Image

1. The fourier spectrum describes the distribution of the power/energy of the original time-series signal - it is the polar form of the fourier transform. Here, the general distribution of the energy in the Fourier spectra is in the horizontal direction and the center contains the highest values, as shown from the spectra figure. Since the image changes intensities from left to right (black, then white, then black), the energy distribution is on the horizontal direction.

2. The energy in the vertical direction is zero in the Fourier spectra. This means that there is no change in pixel intensity vertically. Therefore, looking at the vertical direction of the original image, the pixels are the same intensity.

Next, the test image was rotated by 45 degrees, using the build in MATLAB imrotate function. The fourier spectra was also computed and plotted for the rotated test image, shown below.



1. Prior to the rotation, the spectra of the images should still be the same. When the image was rotated and then the spectra was calculated for the rotated image, it can be seen that the fourier spectra has also been rotated by 45 degrees.
2. Since the Fourier spectra follows the same rotation as the original image rotated, we can infer that the original image pixel intensities change in the direction of the spectra. Here, we can see that the spectrum has also been rotated with the exact angle that was applied to the test image. From the images, we can conclude that the fourier spectrum is <u>insensitive to translation</u>, whether it is vertical, horizontal, or diagonal.

Next, the mandrill image was loaded, converted to grayscale, and the amplitude and phase components were calculated through the fourier transform. The amplitude could be calculated by taking the absolute values of the shifted transform the the image, while the phase could be calculated by finding the quotient of the shifted transform and the amplitude. Lastly, the amplitude and phase were individually fourier transformed inversely, in order to reconstruct the image based on those two individual components. Below displays the original mandrill image, the grayscale image, and the two reconstructed images via the amplitude and the phase.

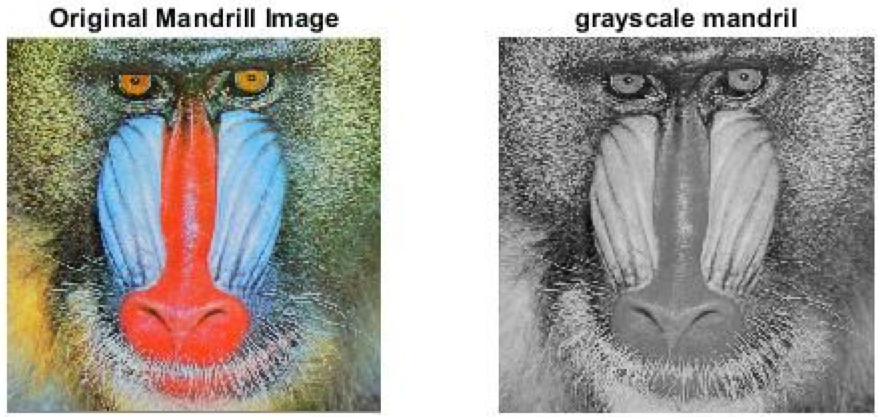


Figure #3 - Mandrill image

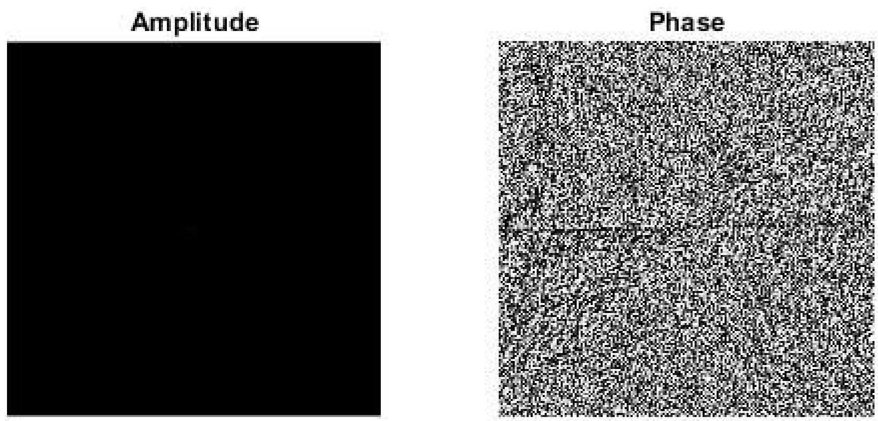


Figure #4 - Amplitude and Phase component of image

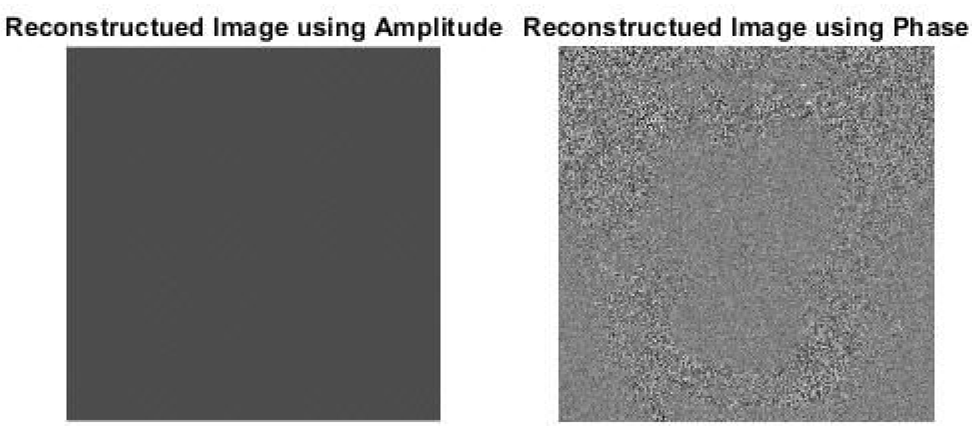


Figure #5 - Reconstructed image using Amplitude and Phase

1. Looking at the reconstructed image using amplitude in Figure #5, it does not have resemblance to the original image. This is because we did not account for the phase, and

did the reconstruction purely on amplitude. From this, we know that amplitude does not strongly represent an image as much as phase.

2. The phase components of the fourier transform define the structural information about the image. As denoted by the phase image in Figure 4, it just looks like static noise. From the reconstructed image using only the phase, we can see a blurry image that contains a somewhat blurry representation of the main structures/edges of the grayscale mandrill image. Comparing the two reconstructed images, it is clear that the phase component is more important when reconstructing the features of the image.

3. Noise Reduction in the Frequency Domain

Next, noise reduction techniques were used via frequency domain filtering to see their effects on the quality of the image. The first noise added to the mandrill image was an additive Gaussian noise, with a mean of 0 and a variance of 0.005. The log spectrum was calculated by logging the calculated shifted fourier transform of the original and noisy image respectively. Below displays the original image, the noisy image, and their corresponding log fourier spectrums.

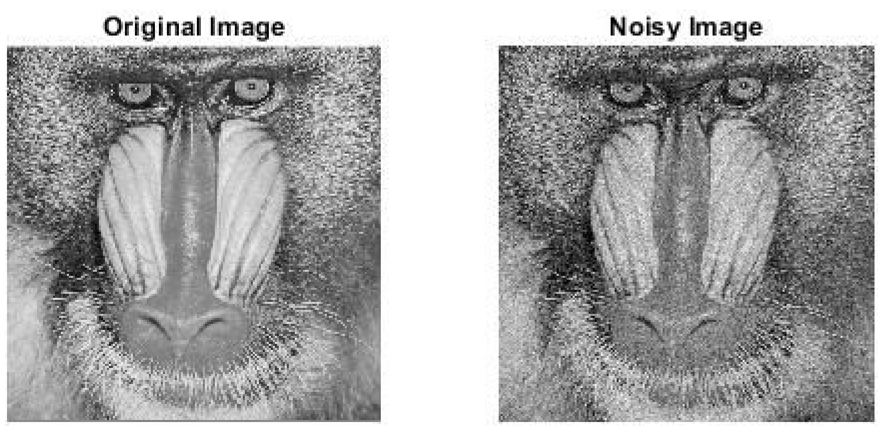


Figure #6 - Original and Noisy image

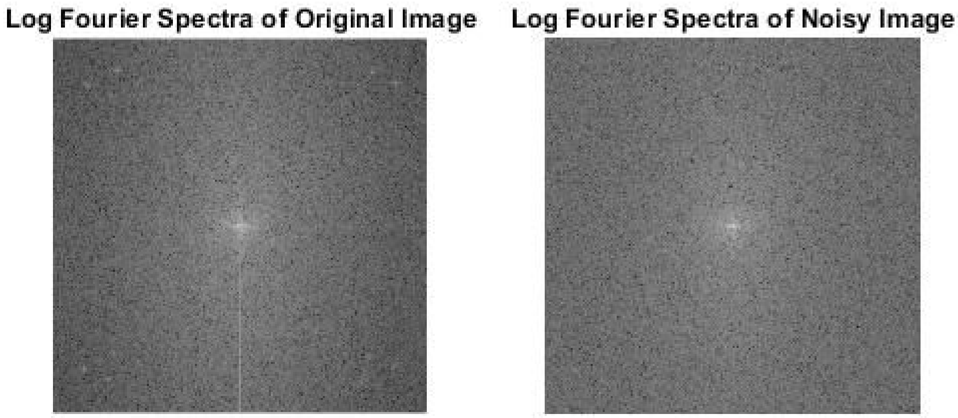


Figure #7 - Log Fourier Spectrums

1. One can see that the noisy image has a lot more noise (looks like static noise) that was added to the original grayscale image. When the fourier spectrum is expanded, the original image had a more noticeable horizontal band while the noisy image spectra did not.

Next, an ideal low pass filter was created, with a cutoff radius of 60 using the fspecial built in function in Matlab. Also, a black image was created, where the height and width of the image was 512 for the mandrill image. The fourier spectrum of that resulting low pass filter was calculated and plotted.

Additionally, that filter was applied to the noisy mandrill image, as shown above. The filter was applied in the frequency domain and then inversely transformed back to reconstruct the original image with reduced noise. It's PSNR was also calculated, which was 17.7112. Below displays the low pass filter spectrum and the inverse fourier transformed image through the use of the low pass filter with a radius of 60.

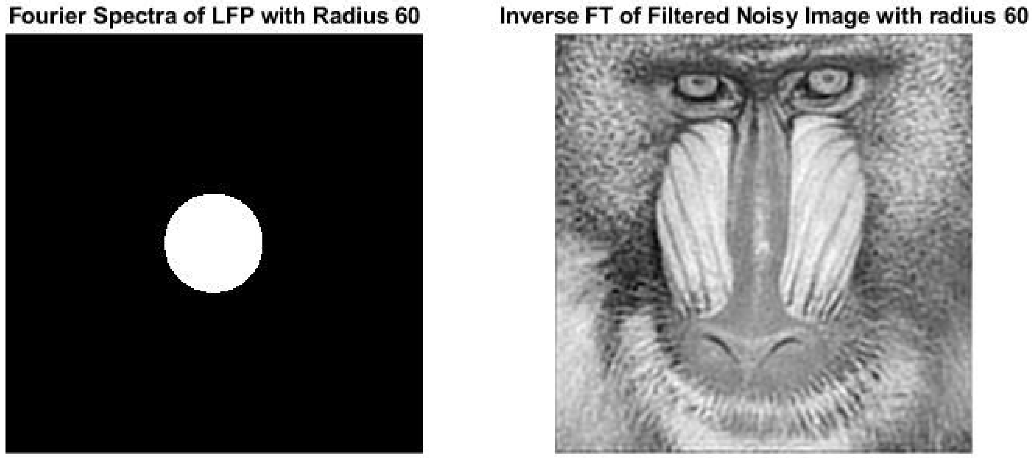


Figure #8 - Reconstructed mandrill image using LFP of radius 60

Here, the mandrill image had a lot of high frequency and thus we tried the exact process stated above with the cameraman image. However, we can still see some ringing effects on the mandrill image, where there are white and black stripes and bands. Below displays the inverse fourier transform of the cameraman noisy image. The PSNR of the cameraman was 23.3571.

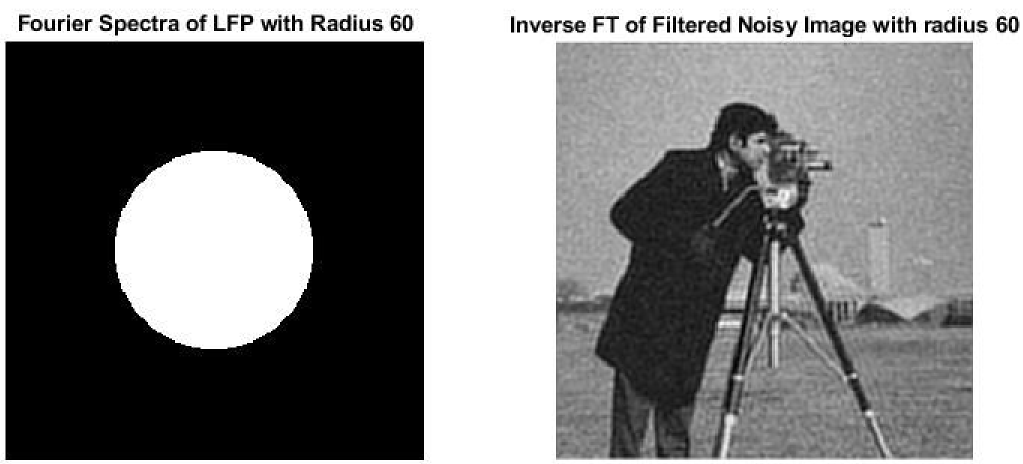


Figure #9 - Reconstructed cameraman image using LFP of radius 60

1. The denoised image has clearly removed the noise introduced by the additive gaussian noise, but through the filter, it also has a lot more smoothing and blurring compared to the original grayscale image.

2. Since the mandrill image had a lot of high frequency, it was harder to visualize what artifact was being created in the frequency domain. Using the cameraman, we can see the ringing effect/artifact, in which there are black and white stripes/bands that are created

from the filter design. Mathematically, the filter's output has oscillations in which produce a ringing band effect.

Now the same process was done using a filter with a smaller radius of 20. Again, the mandrill image was initially used. The fourier spectrum and the inverse fourier transformed image was plotted. The PSNR of the mandrill image with filter radius 20 was calculated to be 18.2917. Below displays the low pass filter with radius of 20 and the inverse fourier transform of the mandrill image.

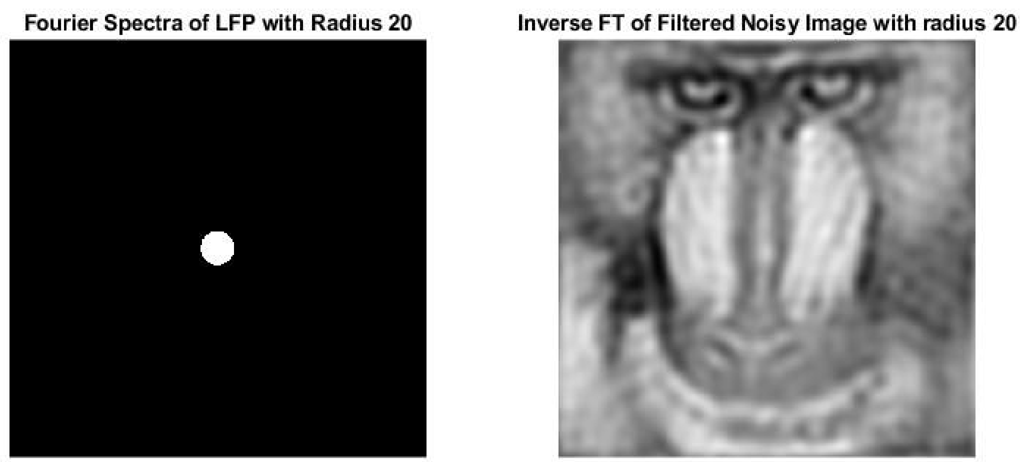


Figure #10 - Reconstructed mandrill image using LFP of radius 20

Afterwards, the cameraman image was additionally used, and the PSNR was calculated to be 21.0378. Below displays the low pass filter with radius of 20 and the inverse fourier transform of the cameraman image.

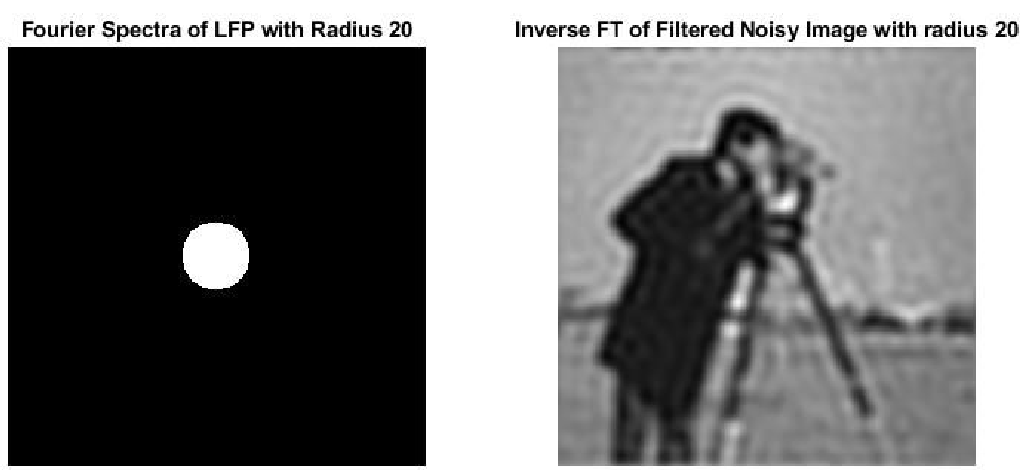


Figure #11 - Reconstructed cameraman image using LFP of radius 20

1. We can see that the reconstructed image using the radius of 20 is far more blurrier and smoother than the one with radius of 60. Additionally, the ringing artefacts can be seen more clearly here on images with radius 20 compared to the reconstructed images with radius of 60. We can conclude that as the radius increases, less blurring and smoothing is applied to the image since less power/distribution of energy is removed. The PSNR for the cameraman image with radius 60 is higher than the PSNR of the cameraman image with radius 60. However, for the mandrill image, the PSNR for the image with radius 60 was a bit lower than that of radius 20.

2. As stated above, when the cutoff radius of the filter is high, less blurring and smoothing occurs to the filtered reconstructed image. In doing so, a high radius can create less blurring of the image but will reduce the noise less compared to smaller radiuses.

Lastly, a Gaussian low pass filter with a standard deviation of 60 was created. It was also normalized by dividing the gaussian filter by the max value of itself, as shown here:

```
gaussian_filter = gaussian_filter./max(max(gaussian_filter));
```

The filter was then fourier transformed to display the gaussian fourier spectrum. It was then applied to the noisy mandrill image, in which the inverse fourier transformed reconstructed image was then plotted and it's PSNR was calculated, which was 13.3060. Below displays the Gaussian filter, and its corresponding spectrum, and the inverse transformed gaussian image.

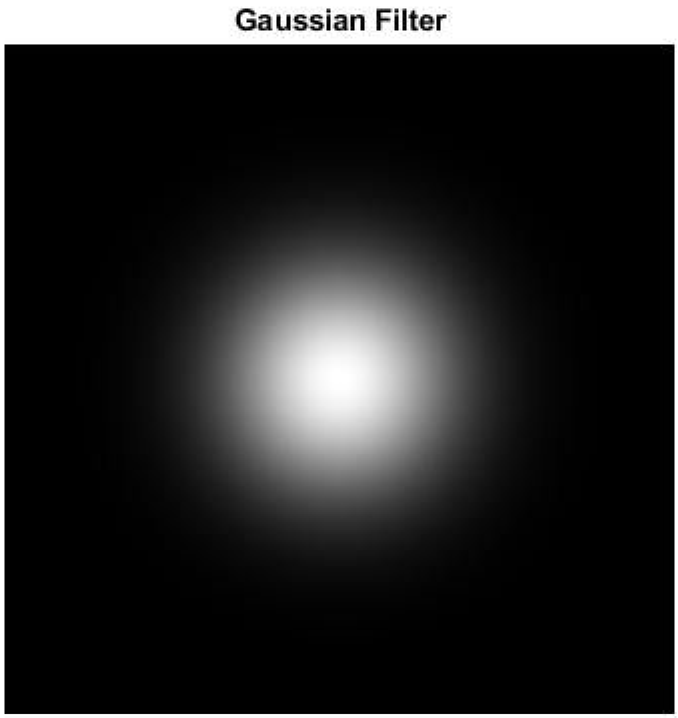


Figure #12 - Gaussian filter

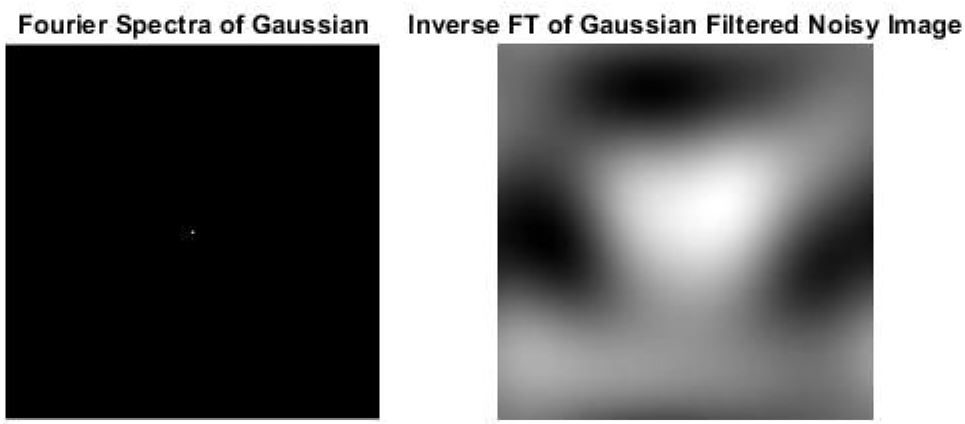


Figure #13 - Spectrum and Inverse FT of Gaussian Filtered image

1. It is clear that the Gaussian filtered image was reconstructed very poorly, as shown above. The PSNR of the image was 13.3 for the mandrill image, which was the lowest of the 3 PSNR's calculated for the reconstructed mandrill image. One main advantage of gaussian low pass filters is that the ringing artifacts will not be present in the final reconstructed image, where the gaussian in the frequency domain will remain as a Gaussian in spatial domain.

4. Filter Design

Next, we plotted the frequnoisy.tif image and the fourier spectrum. It can be seen that there are diagonal noise that was added to the image. Below shows the figure:

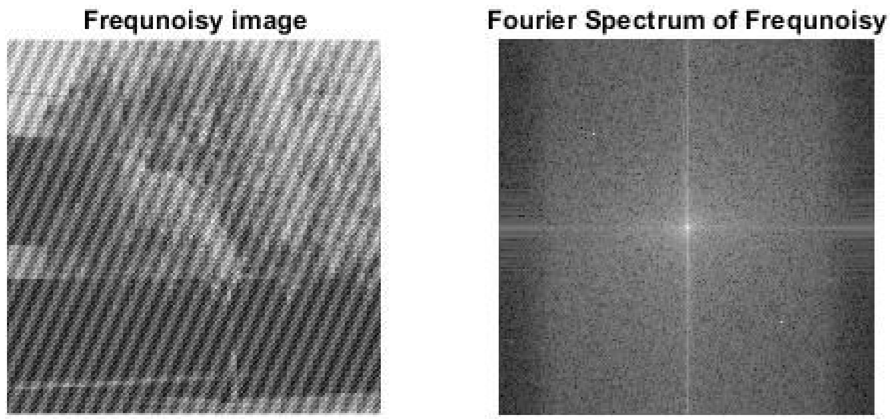


Figure #14 - Freunoisy.tif image and it's Fourier Spectrum

It can be seen in the fourier spectrum that there are 4 white points/dots that illustrate the low and high noise bands that correspond on the original noisy image. By deleting those noise points, we can reconstruct the original image without any noise.

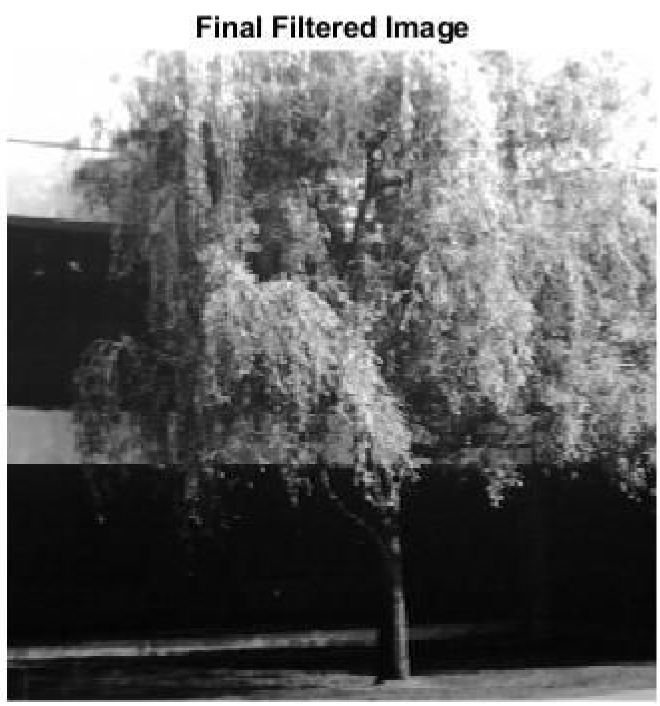


Figure #15 - Final filtered image

We simply took the points that were directly across from them and replaced the values. After doing an inverse fourier transform, we were able to see that this filter worked.

5. Conclusion

In this lab, we investigate how the process of image restoration within the frequency domain. Specifically, the use of fourier transform to perform image restoration, along with the use of various types of filters was completed. First, we learned how a fourier transform of a rectangle looked like in the frequency domain. Next, several noise reduction techniques were investigated, using various filters such as an ideal low pass filter with various cutoff radii, and a Gaussian low pass filter. Lastly, we implemented a new filter that would help filter out an image with periodic noise by looking at its fourier transform. We saw a few points that were causing the artifacts and managed to restore the image to its original state.