CMPEN/EE455: Digital Image Processing I

Computer Project # 3:

Frequency Domain Analysis of Filters

Pavan Gurudath

Date: 10/20/2017

OBJECTIVES

The project is aimed at understanding the basic DFT- based frequency analysis and perform filtering using frequency-domain analysis on the images. The main objective of this project involves the following -

1. **Basic DFT-based frequency analysis**

- To compute and understand the DFT of an image. Also, to infer the usage of logarithmic and modulation operations on the images.
- To observe the output image after applying a Gaussian lowpass filter on an image:
    - without zero-padding
    - with zero-padding

2. **Filtering a corrupted image**

- To create a corrupted image by introducing a noise component to the image.
- To design a suitable notch filter to eliminate the noise and obtain the output image that resembles the original image.

METHODS

This project consists of two main objectives, i.e. DFT-based frequency analysis and filtering a corrupted image using frequency analysis. In order to run the MATLAB program, please select the main.m file and part_two.m file and run it on the MATLAB environment to obtain the results for the first and second objective. The 'main.m' file calls the functions 'part_two.m', 'dftuv.m' and 'lpfilter.m'. The 'main.m' performs the methods to complete the tasks of the first objective and 'part_two.m' in order to complete the tasks of the second objective. 'lpfilter.m' and 'dftuv.m' performs the task of forming a low pass filter.
The following subsection details the method in which the project statement has been solved.

1. **DFT- BASED FREQUENCY ANALYSIS:**
   The objectives for understanding the DFT-Based frequency analysis involve operations to be done on an input image named ***'checker.gif'.*** The main task of the objective is to understand the DFT and IDFT of the given image under different circumstances.
   1.1 To compute the DFT of the given image:
   - The image 'checker.gif' is denoted by $f(x, y)$. The DFT of the given image can be computed using the inbuilt MATLAB command whose syntax is given as follows:

$$F(u, v) = fft2(f, M, N)$$

$$where\ f - given\ image$$

$$M \times N\ is\ the\ dimension\ of\ the\ image$$

   - The above mentioned command gives the Fourier transform of the given image and the resultant image is depicted in the frequency domain.
   - Now we modulate the image to observe the changes in the frequency domain. For accomplishing this, the following procedure is executed.

$$f_m(x, y) = f(x, y) \times (-1)^{x+y}$$

   - The DFT of the new image $f_m(x, y)$ is computed in a similar manner using the inbuilt MATLAB command *fft2* and is denoted by $F_m(u, v)$.

   1.2 To compute the IDFT of the given image:
   - Initial step of this objective is to make the $F(1,1)$ (as per MATLAB indexing) to zero.
   - The inverse DFT of the given image can be found out by the inbuilt MATLAB command whose syntax is given as follows:

$$g(x, y) = ifft2(F, M, N)$$

$$where\ F - DFT\ of\ f(x, y)$$

$$M \times N\ is\ the\ dimension\ of\ the\ image$$

   - The $g(x, y)$ is the resultant image of the inverse DFT computation.

The observable difference between g and the original image f is that the image in g looks darker in comparison to the original image f. This is due to the fact that since the DC component is set to 0, the intensity of the image is reduced significantly.

1.3 To understand the convolution of images without zero- padding:

- The Gaussian low- pass filter is generated using the inbuilt MATLAB command whose syntax is given as follows-

$$H(u,v) = lpfilter('gaussian', M, N, sig)$$
$$where\ M \times N\ is\ the\ dimension\ of\ the\ image$$
$$sig\ is\ the\ cutoff\ frequency = 15$$

- To visualize the 3D plot of the resultant filter $H(u,v)$, the inbuilt MATLAB command *mesh* is used.
- The DFT of the input image obtained is allowed to pass through the filter *H(u, v).* This can be mathematically represented as –

$$G(u,v) = H(u,v) \times F(u,v)$$

- The *G(u, v)* is the resultant DFT of the image after passing through the low-pass Gaussian filter. The filtered image can be obtained by performing Inverse DFT using the *ifft2* command in the MATLAB.
- Appropriate modulation and logarithmic expressions are carried out to display the image.

1.4 To understand the convolution of images with zero-padding:

- In the zero padding , the dimensions of the images are changed to P and Q. The image occupies MxN dimension of the PxQ matrix. The remaining rows and coloumns are padded with zeroes.
- Similar set of tasks are done as explained in section 1.3, and the images are observed.

While displaying images in the frequency domain, the absolute value is plotted using the function $\log(1 + |F(u,v)|)$ where $|F(u,v)|$ is the absolute value of the DFT of $f(x,y)$**.** The log operation is used in order to compress each pixel value whenever the dynamic range of the image is too large. This has the effect that low intensity pixel values are enhanced while the high intensity values are able to be displayed upon the screen. The function is added with 1 because logarithm of 0 is undefined and since pixel intensity usually consists of 0, therefore it is added.

For the results of part b and c, the filter has the impact of eliminating the high frequency components. This causes the image to blur. Also, it can be observed at the corner edges, a perfect 0 at the top left corner cannot be observed in the case of part b. However, in part c it can be observed since due to zero padding the circular convolution in spatial domain allows the top left and bottom right corner to have a perfect zero.

Yes, the wrap around error can be seen to be eliminated after zero padding. This can be seen by observing the pixel values at the edges.

2. **FILTERING A CORRUPTED IMAGE:**

In order to perform the objectives under this category, an input image named ***'lake.tiff'*** that is of the dimensions [512*512*2] has been utilized. The .tiff image file consists of two layers, where the first layer consists of the necessary information and the second layer consists of the white pixels throughout. To satisfy the objectives, the image in the first layer is required. Therefore, the first layer is extracted and the subsequent set of processing steps are applied. The new image is of the dimension [512*512].

2.1. To create the corrupted image:

- The image 'lake. If' is denoted by $f(x.y)$. The image $(f(x,y)$ is scanned through every pixel and subsequently added with the noise component. The noise is given by

$$\mathbf{32 * cos(\frac{2\pi 32x}{N})}$$

- The frequency analysis of the above mentioned noise is depicted below:

$$n(x,y) = 32\cos\left(\frac{2\pi 32x}{N}\right)$$

Here N=512

$$n(x,y) = \left(\frac{1}{2}e^{-j\frac{2\pi 32x}{N}}\right) + \left(\frac{1}{2}e^{j\frac{2\pi 32x}{N}}\right) \quad \text{……..…… (1)}$$

Now,

$$e^{-j\frac{2\pi ux}{N}} \quad \text{is periodic in U with period N = 512.}$$

$$e^{-j\frac{2\pi ux}{N}} = e^{-j\frac{2\pi Nx}{N}} . e^{-j\frac{2\pi ux}{N}}$$

$$= e^{-j\frac{2\pi(N-u)x}{N}}$$

This can be applied to (1) above with u = 32:

$$n(x,y) = \left(\frac{1}{2}e^{j\frac{2\pi 32}{N}}\right) + \left(\frac{1}{2}e^{\frac{2\pi 480x}{N}}\right) \quad \text{…………………..(2)}$$

Now, if we know the DFT of $n(x,y)$ which is $N(u,v)$:

$$n(x,y) = \left(\frac{1}{N^2}\right)\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} N(u,v)\, e^{-j\,\frac{2\pi(ux+vy)}{N}} \quad \text{…………………….. (3)}$$

Comparing (2) and (3), we can read of the DFT of $n(x,y)$:

$$N(32,0) = 32\frac{N^2}{2}$$

$$N(480,0) = 32\frac{N^2}{2}$$

$$N(u,v) = 0 \qquad\qquad for\ all\ other\ (u,v)$$

Now representing as a function,

$$N(u,v) = 32\frac{N^2}{2}\{\delta(u-32,v) + \delta(u-480,v)\}$$

- The resultant corrupted image is stored as $\mathrm{c}(x,y)$ and is given by the following equation:

$$\boldsymbol{c(x,y) = f(x,y) + 32\cos(\frac{2\pi 32x}{N})}$$

- The resultant image is stored as 'lake_corrupted.jpeg'.

2.2. To design a suitable notch filter:

- A notch filter $H(u,v)$ that when applied to the $c(x,y)$ gives an image $g(x,y)$ that nearly resembles the original image $f(x,y)$.
- The notch filter rejects specific frequencies while passing all other frequencies.
- In the given objective, the notch filter has to be designed in such a way that the noise component frequencies gets rejected.
- From the analysis shown in **2.1,** it is deducible that the noise component given by,

$$N(u,v) = 32\frac{N^2}{2}\{\delta(u-32,v) + \delta(u-480,v)\}$$

are concentrated on the points (32,0) and (480,0).

- Hence, the filter can designed to reject the frequencies at (32, 0) and (480, 0).
- The filter can be realized and represented by the following,

$$H(u,v) = \begin{cases} 0, & u = 32\ ,480 \\ 1, & Otherwise \end{cases}$$

Also, it can be observed that the original image cannot be completely recovered from a noisy image after passing it through a filter. This is because while using the notch filter, the particular frequency is set to zero and that particular frequency is completely lost. Therefore if the image consists of that frequency component that was set to zero, then while passing through the filter, the image loses that frequency and thereby we cannot recover the original image completely.

RESULTS

This section details all the images that are obtained after completing each of the objective.

The image considered for the filtering using frequency analysis is the 'checker.gif' is shown below.

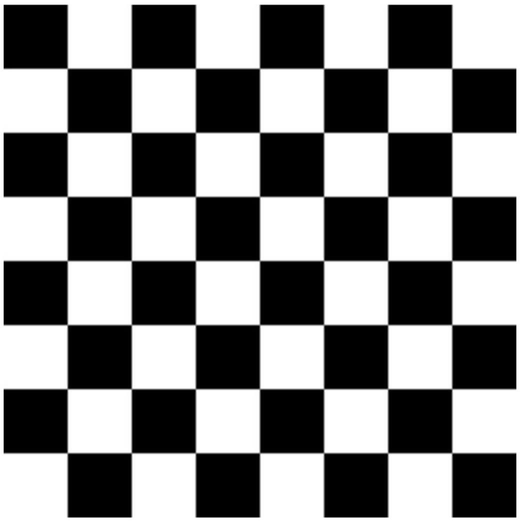

Figure 1: Checker.gif

We plot the figures for |F(u,v)| and log(1+|F(u,v)|) as shown in figure 2 and 3 respectively.

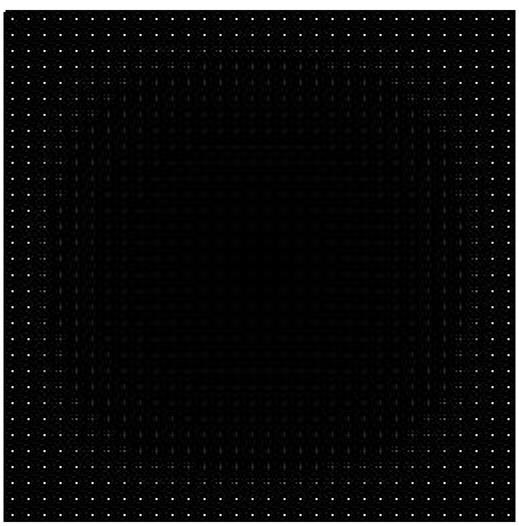

Figure 2: |F(u,v)|

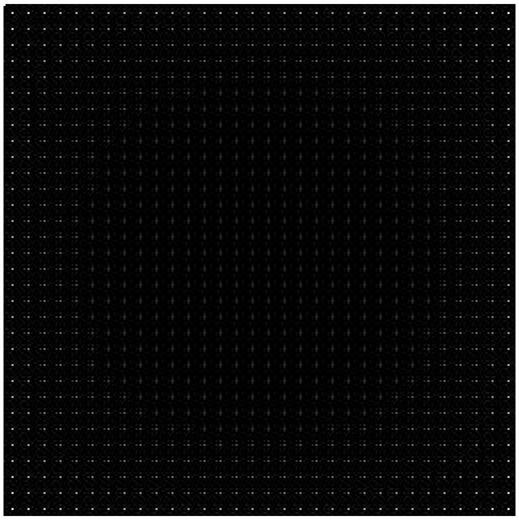

Figure 3: log(1+|F(u,v)|)

To observe the functionality of the modulation, we apply modulation to f(x,y) to obtain fm(x,y) and hence obtain |Fm(u,v)| and log(1+|Fm(u,v)|) as shown in figure 4 and 5 respectively.

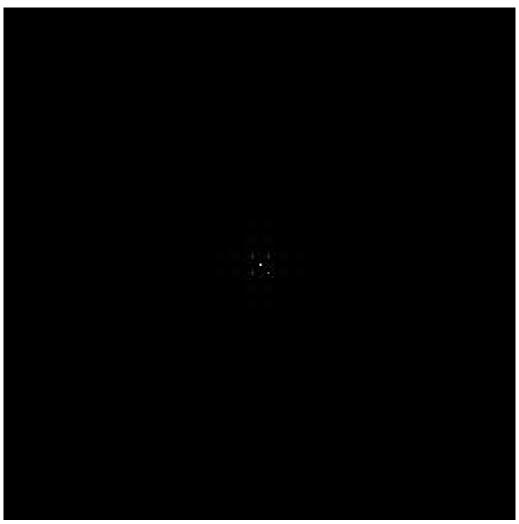

Figure 4: |Fm(u,v)|

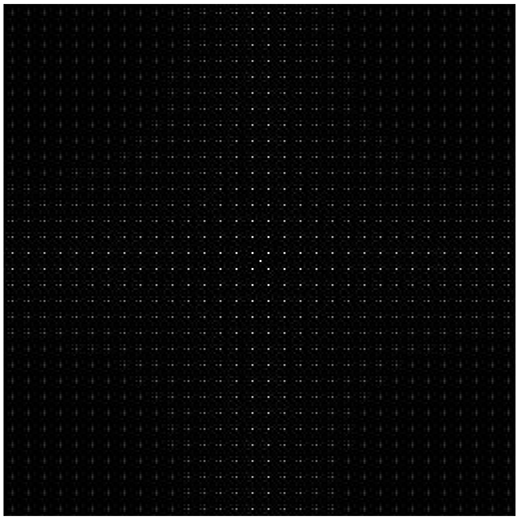

Figure 5: log(1+|Fm(u,v)|)

It can be observed that the centre consists of the higher frequencies and the intensity diminishes as you move away from the center.

Further, the DC component is set to zero and figure 6 g(x,y) is plotted.

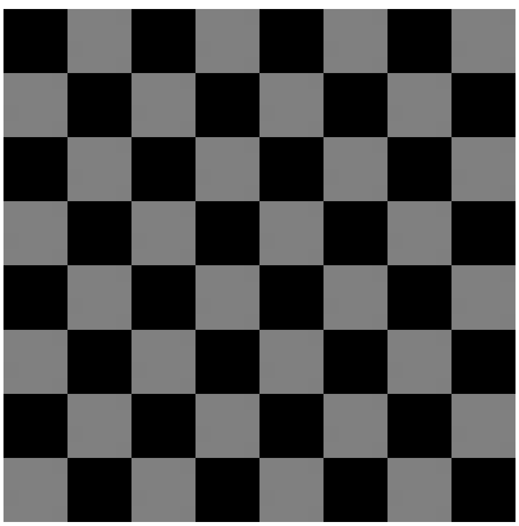

Figure 6: g(x,y)

Therefore, the image is observed to be darker than what it originally was.

Next, the gaussian low pass filter is chosen as follows:

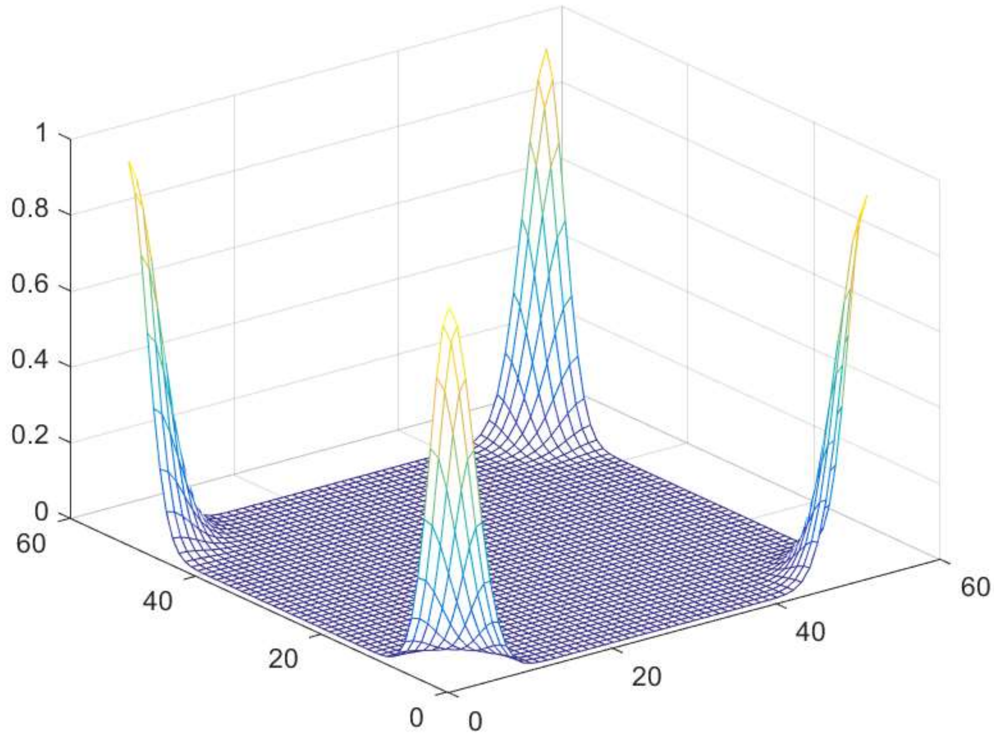

Figure 7: Mesh for gaussian filter with sigma=15 for 256x256 image

This filter is passed through the image and the resultant image g(x,y) is obtained in the spatial and frequency domain as follows. The figure 8 and 9 are displayed appropriately after modulating and then taking its DFT.

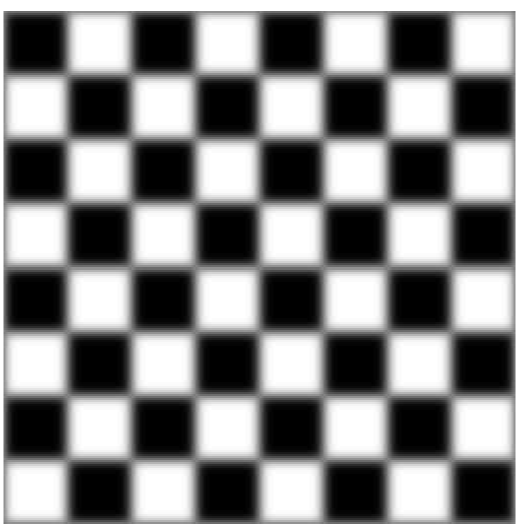

Figure 8: Modulated output image without zero padding

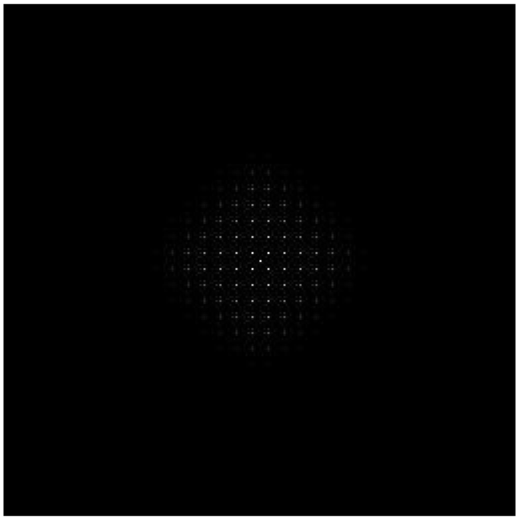

Figure 9:Modulated output image in frequency domain without zero padding

Figure 9 represents log(1+|G(u,v)|). In order to ensure the wrap around error doesn't occur, the image is padded with zeros. And the filter that is considered for the new PxQ image is as shown

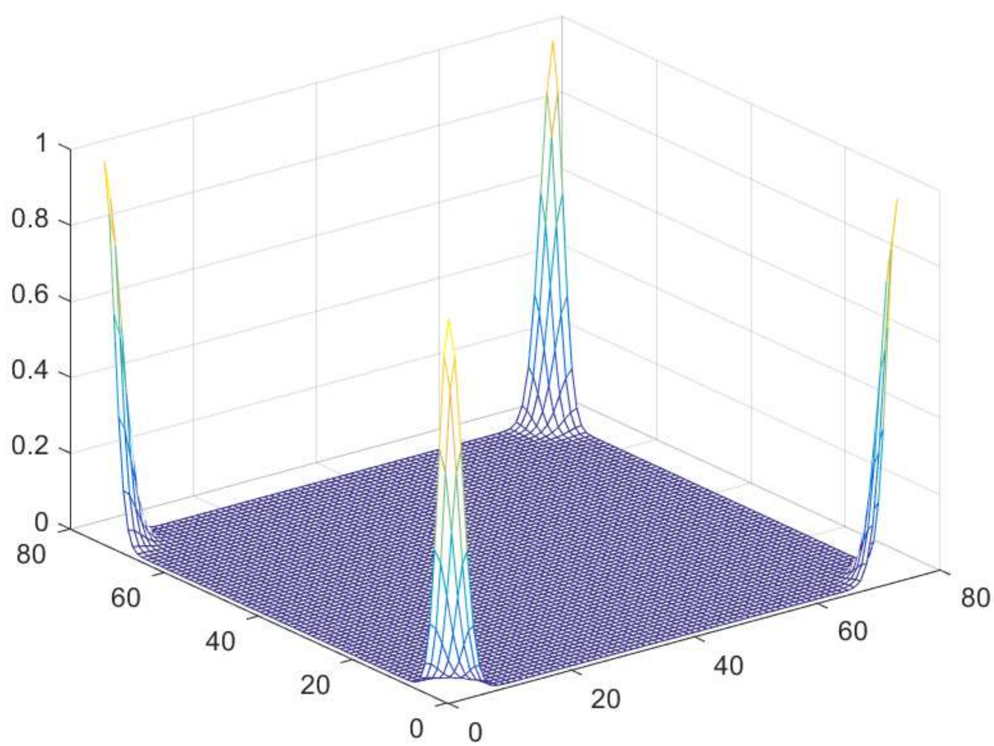

Figure 10: Mesh for gaussian filter with sigma=15 for 512x512 image

After passing this filter through the zero-padded image, the resultant image g(x,y) is obtained in the spatial and frequency domain as follows. The figure 11 and 12 are displayed appropriately after modulating and then taking its DFT and displaying the logarithmic value of it.

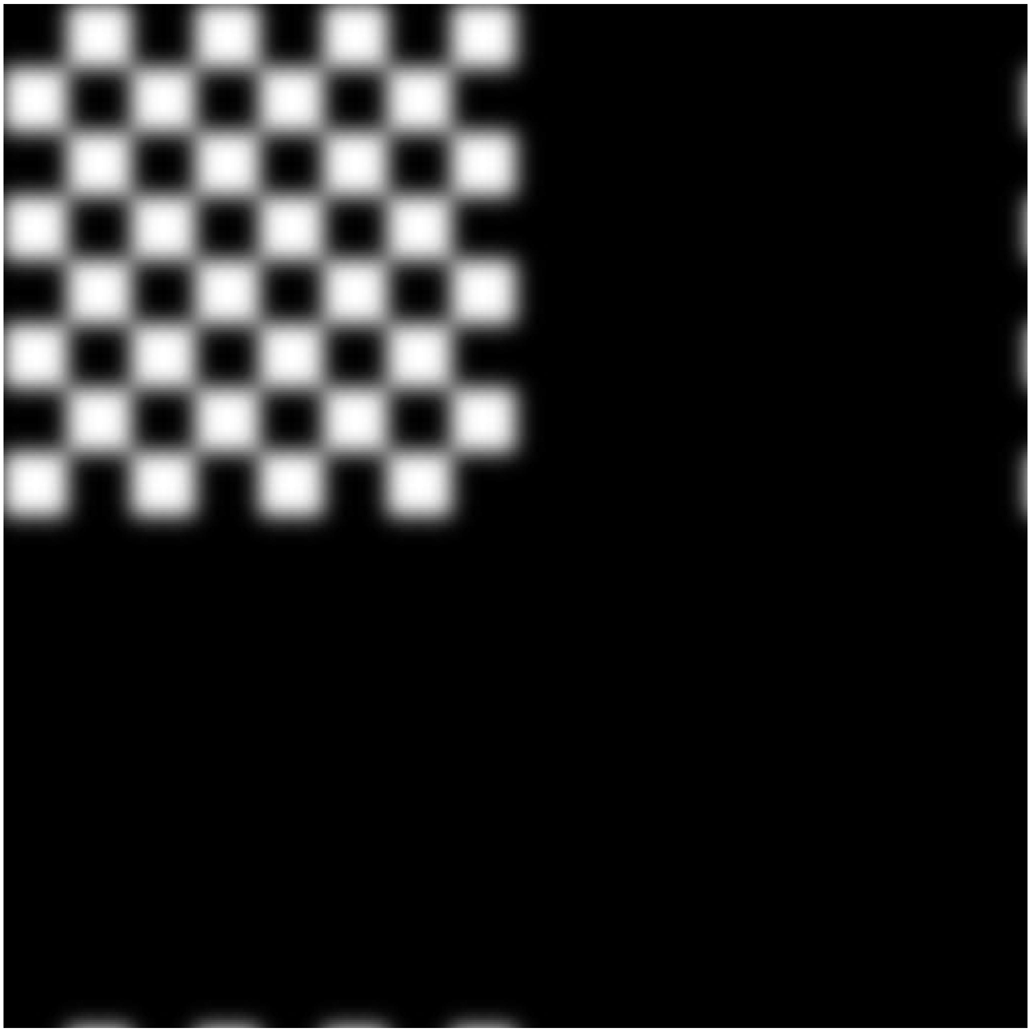

Figure 11: Modulated output image with zero padding of size 512x512

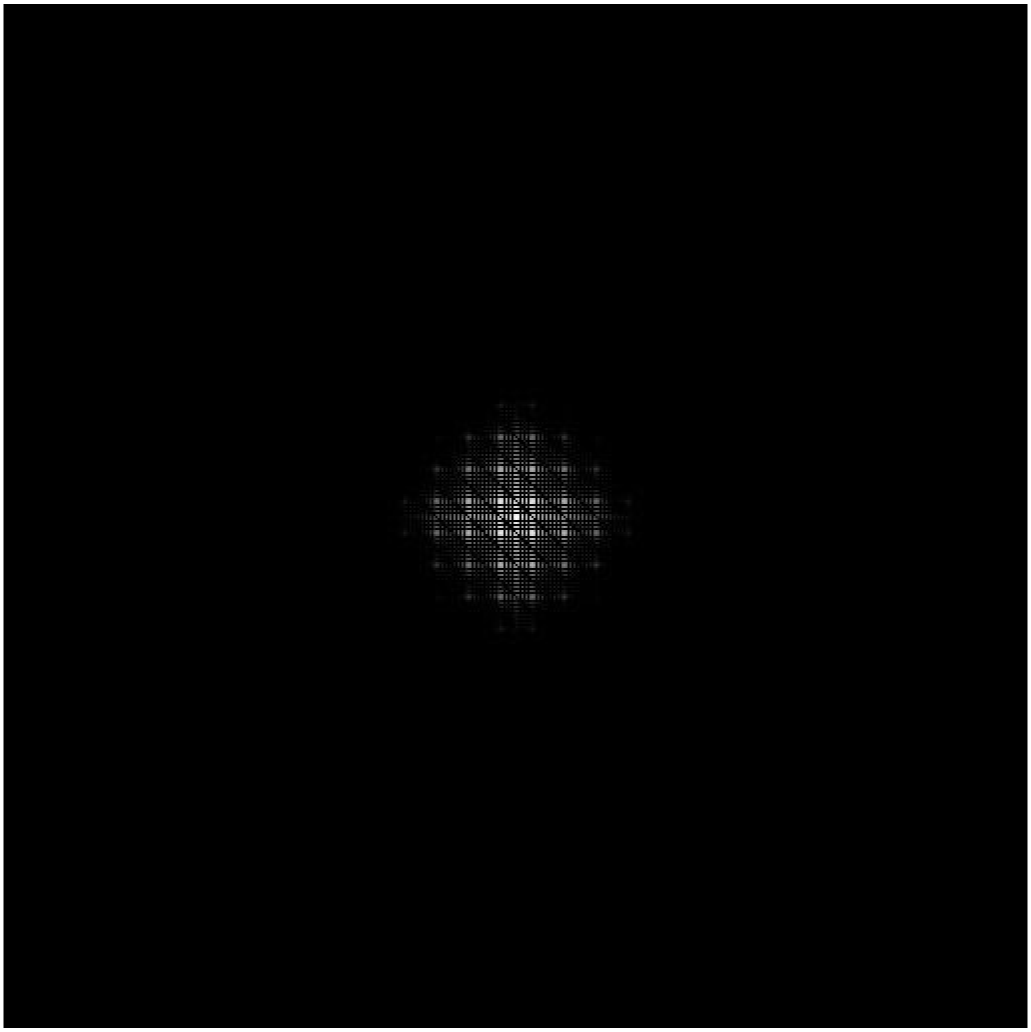

Figure 12: log(1+|G(u,v)|) - Modulated output image in frequency domain with zero padding

It can be observed from figure 11 that the desired area of the image is in the upper left portion. Its size corresponds to MxN. The remaining area is the regions formed due to zero padding. The effects of zero padding can also be seen at the right and bottom edges of the images where the alternating white and black bands are formed.

PART-2:

The image considered for the filtering using frequency analysis is the ‘lake.tiff’ as shown below in figure 13.

Figure 13: Lake.tiff

This is the original image to which noise is added. The noisy image is as shown in figure 14.

Figure 14: Noisy image of the lake c(x,y)

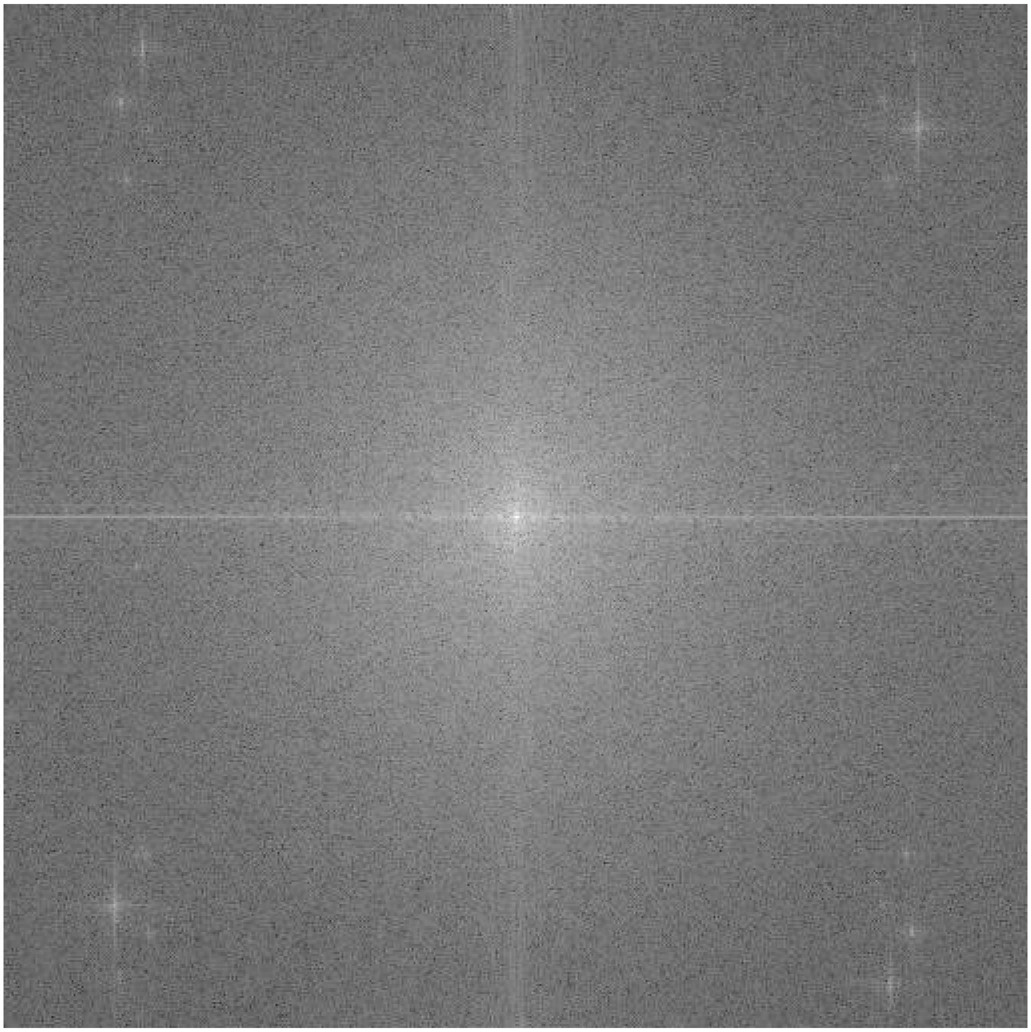

Figure 15: Frequency domain of the original image f(x,y)

The figure 15 has been plotted after modulating the original lake image and then taking the logarithmic scale.

Figure 16 showcases the DFT plot of the noisy image c(x,y) and is represented as |C(u,v)| and is plotted using the function log(1+|C(u,v)|).

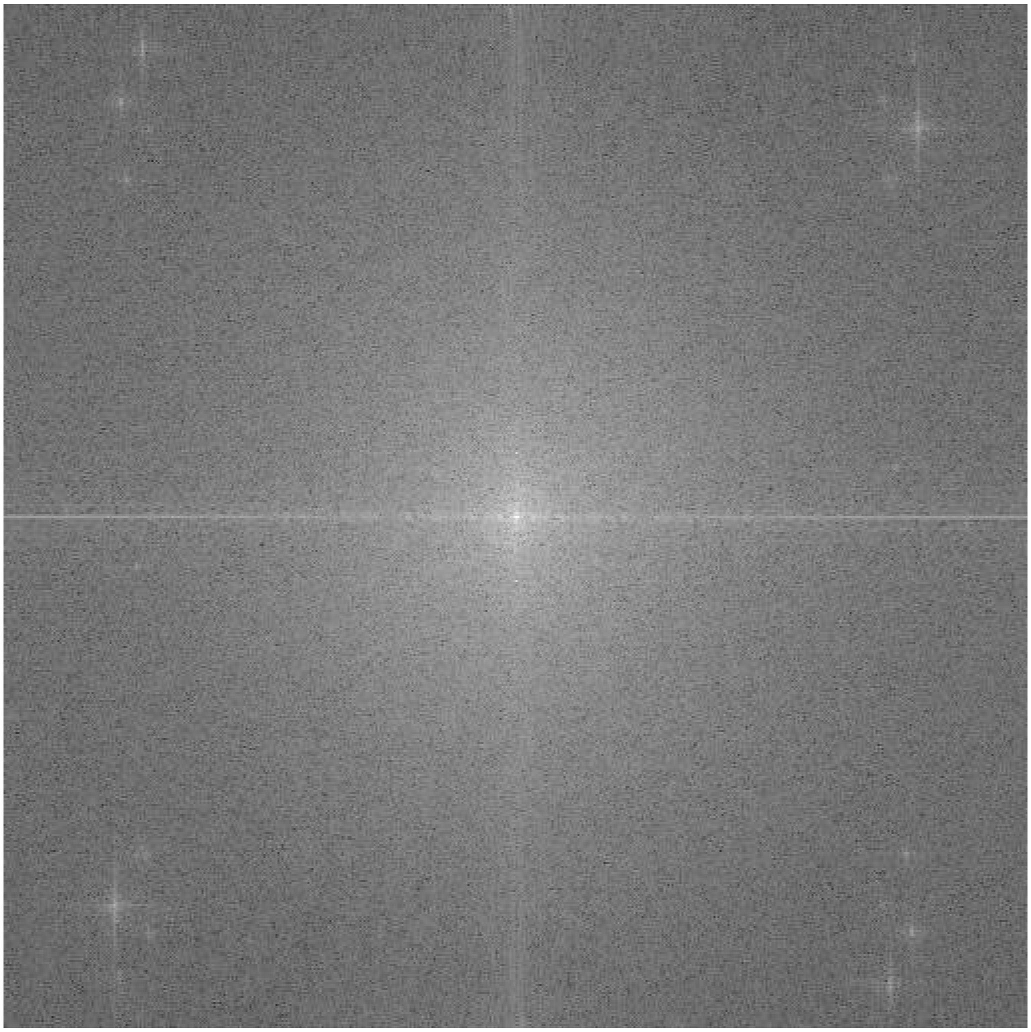

Figure 16: Frequency domain of the noisy image c(x,y)

The notch filter so designed in its frequency domain is as shown.

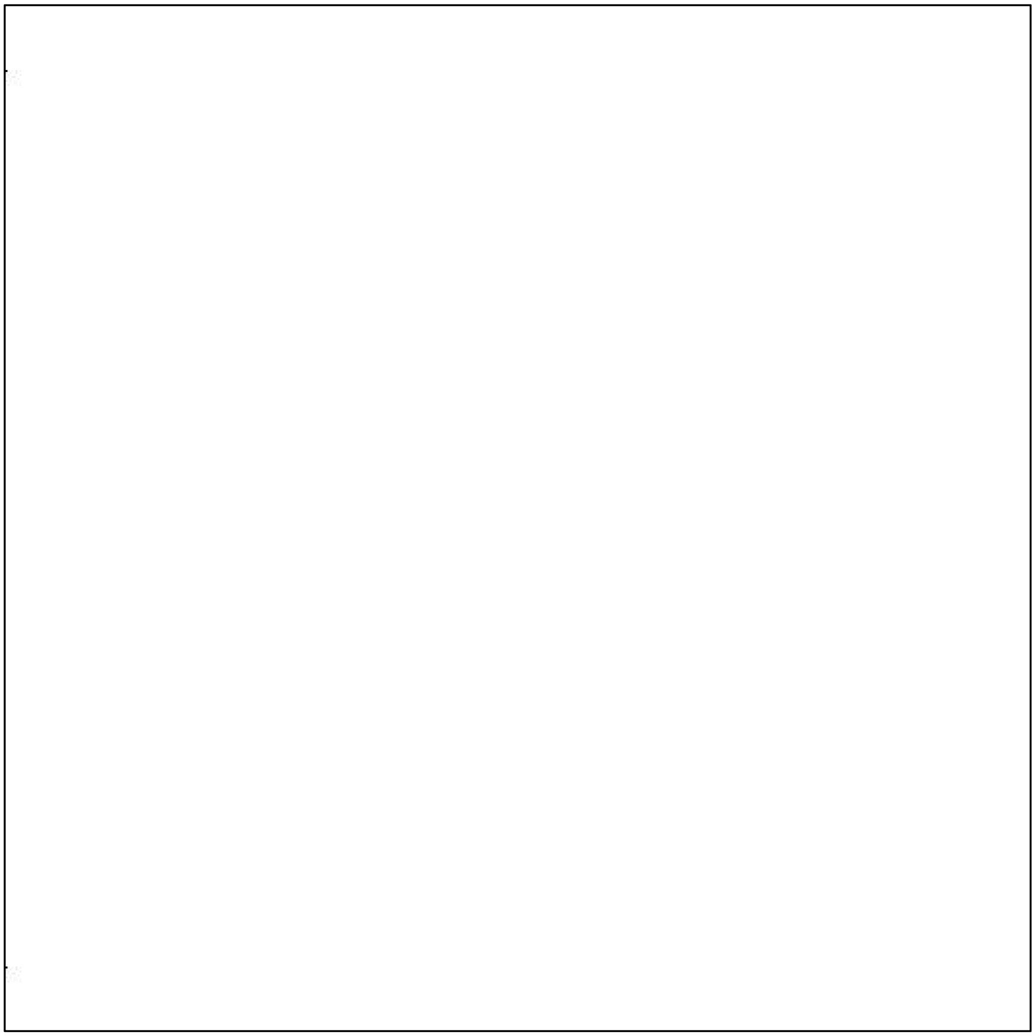

Figure 17: |H(u,v)|Notch filter

This notch filter has two pixels whose values are zero and can be observed in the figure shown below.

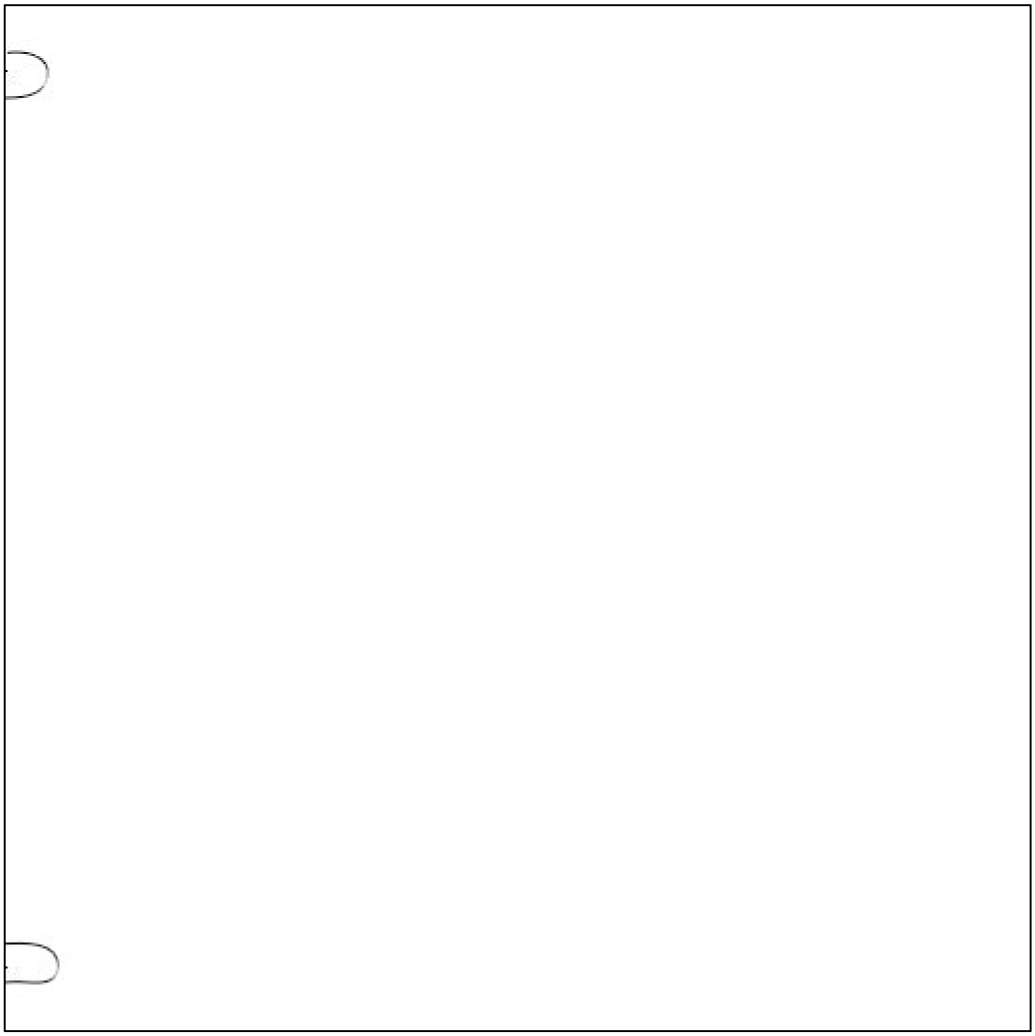

Figure 18: Two pixels of H(u,v) shown where its intensity is zero.

This notch filter is passed through the noisy image in order to obtain the output image as shown below in figure 19.

Figure 19: Output image after the noisy image is passed through the notch filter

The frequency domain image is shown in Figure 20 once the image above is modulated and its DFT computed.

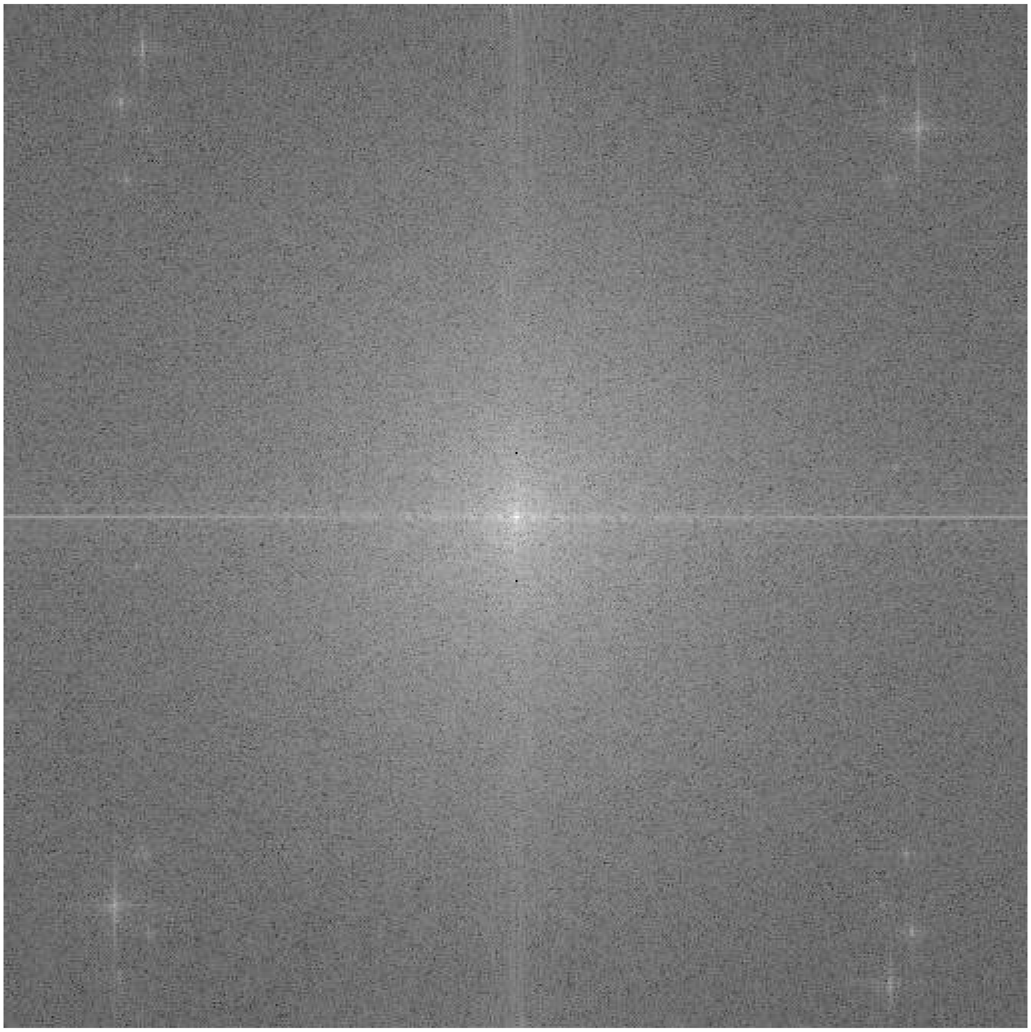

Figure 20: Frequency domain image of g(x,y)

The difference in the image of g(x,y) and f(x,y) which is the difference in the image recovered after adding noise, and the original image is as shown in figure 21.

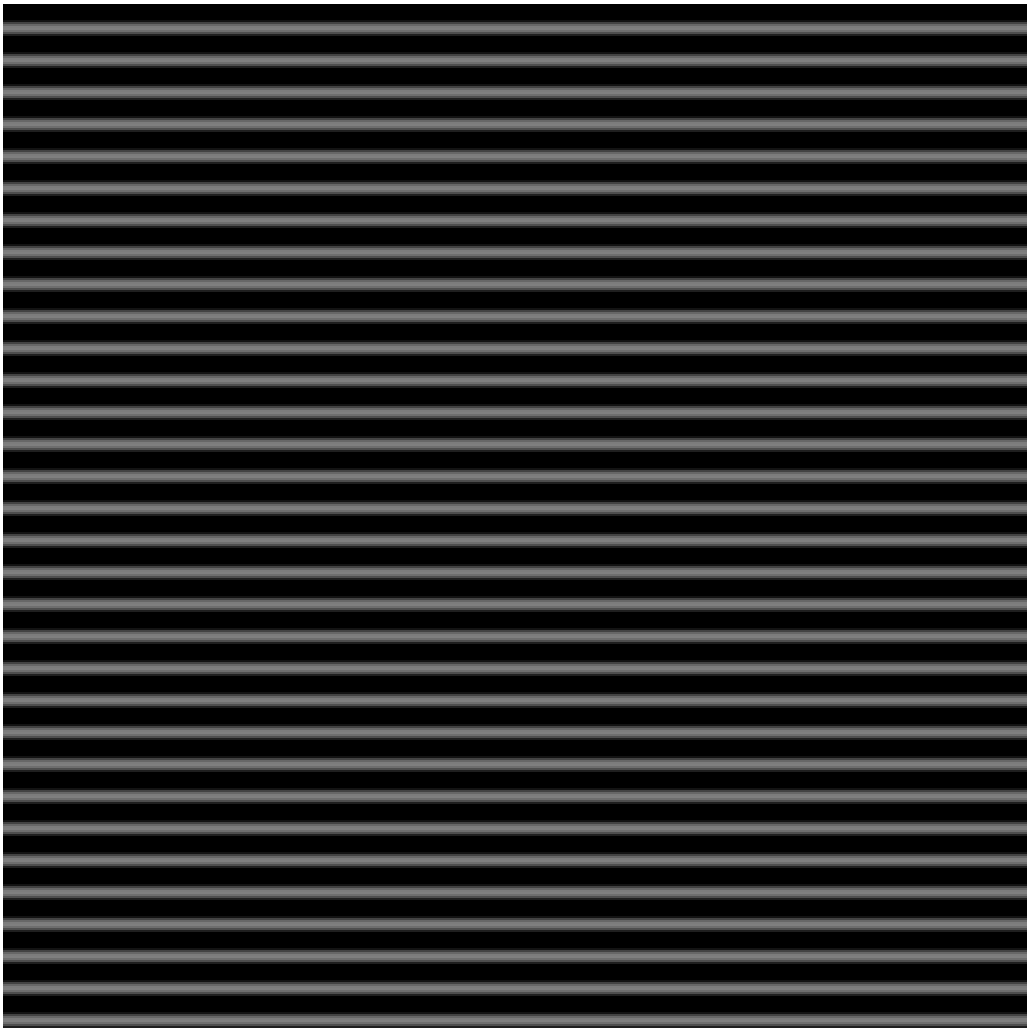

Figure 21: Difference image: f(x,y) - g(x,y)

The DFT of this difference image is as shown below in figure 22.

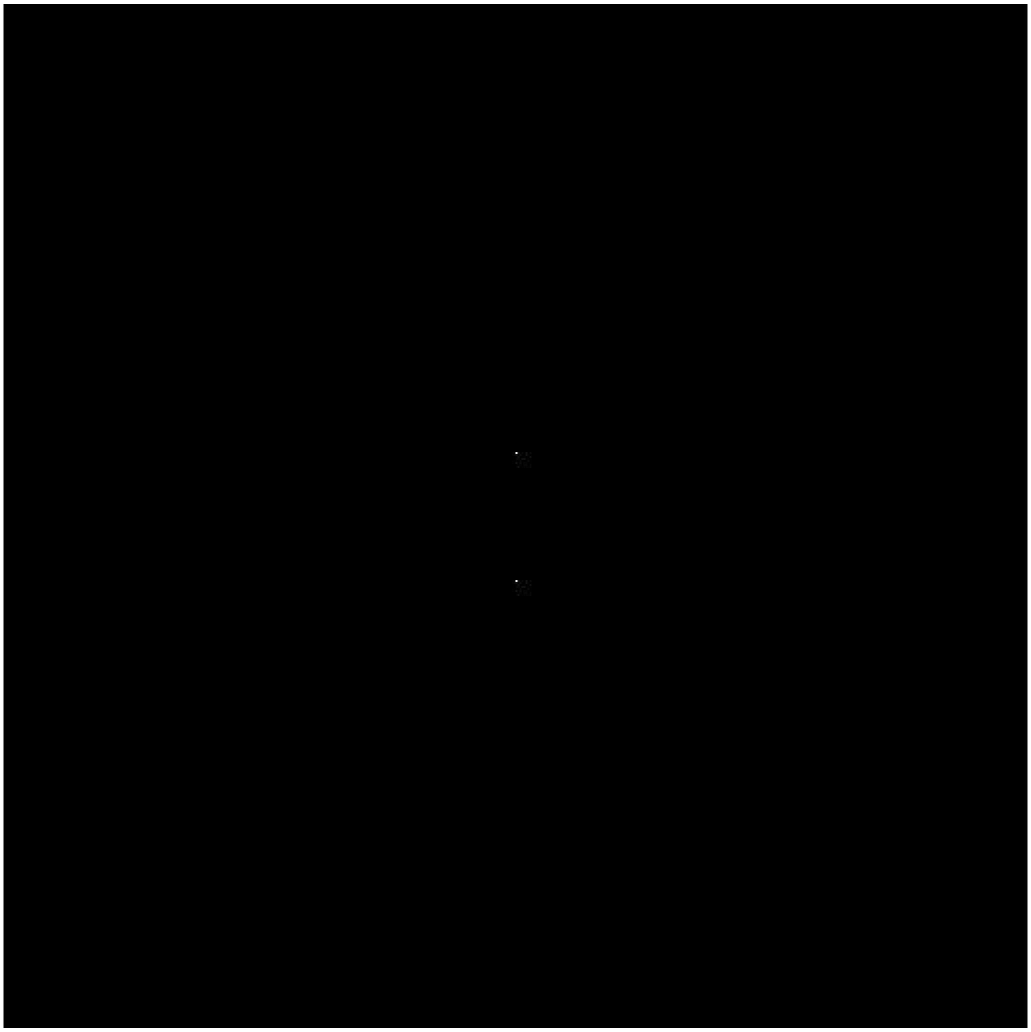

Figure 22: log(1+|DifferenceImage(u,v)|)

As seen from Figure 22, the difference image consists of only two frequencies similar to the noise image. The reason that the two frequencies appear is the fact that although these frequencies were set to zero in the filtered image, these components existed as a part of the original image. Thus difference image consists of only the two frequency components of the original image which were set to 0 in the filtered image which results in the alternating band.

CONCLUSION

After completing the necessary tasks to meet the objectives of this project, we can come to a set of following conclusions:

1) The DC component of any image contains maximum information pertaining to that image.
2) The low frequency components of any image is closer to the outer edges, and as we move towards the center, the frequency increases. Modulation helps in bringing the components from the edges to the center by proving a shift of N/2.
3) Low pass filters are used to remove the high frequency components, thereby causing the image to be blurred. However, unlike the low pass filters that do not offer a strict cutoff frequency, the use of digital notch filters using frequency domain analysis allows the elimination of very specific frequencies from an image.
4) It may not be possible to retrieve the original image from a noisy image completely, but the use of a notch filter for particular frequencies will allow us to retain the overall image quality to a great extent.
5) It can be observed that for a digital image, the DFT performs circular convolution and therefore zero padding an image becomes necessary.