IMAGE PROCESSING AND INTERPRETATION

CS474.674.1001

PROGRAMMING ASSIGNMENT NO: 3

A Report

By

PRATIK WALUNJ

JALPA KAILA

Division of Work Statement

Pratik Walunj and Jalpa Kaila divided the work for this assignment as follows:

Coding:

- Pratik handled coding for Questions 2 to 3.

- Jalpa handled coding for Questions 1.

Report Write-up:

- Pratik contributed to Section 2.1, 2.2, 3.1 and 3.2.

- Jalpa contributed to Section 2.2, 2.3, 3.2 and 3.3.

We collaborated on coding and provided mutual input for report sections to ensure balanced contribution in both aspects.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1. MOTIVATION

This assignment is like a puzzle where we get to explore something called the Discrete Fourier Transform (DFT) and a quick way to do it using computer code. We got the chance to implement what we learned in class and this assignment lets us play with images in frequency domain. It's like peeking into the hidden details of images. We will be using a special technique called Fast Fourier Transform (FFT) in python. This is like a hands-on journey to understand how computers process images in frequency domain. It's not just theory; we will be doing real experiments, and that's what makes it so interesting for us!

CHAPTER 2. THEORY

2.1 Fourier Transform

The Image processing tasks can be best performed in a domain other than the spatial domain. Here we perform tasks in the frequency domain. It is easy to manipulate the frequencies in frequency domain and it's faster to perform some operations such as convolution in frequency domain rather than spatial domain. Fourier Transform either be a continuous or Discrete in frequency domain.

2.1.1 Discrete Fourier Transform

Discrete Fourier Transform is defined as converting continuous functions into a sequence of discrete values using sampling. Here we use Discrete Fourier Transform for both Forward and Inverse transforms.

1. Forward DFT

$$F(u) = 1/N \sum_{x=0}^{N-1} f(x) e^{-\frac{j2\pi ux}{N}}, \qquad u = 0,1,2 \dots, N-1$$

2. Inverse DFT

$$f(x) = \sum_{u=0}^{N-1} F(u) e^{\frac{j2\pi ux}{N}}, \qquad x = 0,1,2 \dots, N-1$$

In general F(u) is complex function with Real and Imaginary part. We can write the Complex function with Real and Imaginary parts and find the Magnitude and Phase of this Functions. Here R(u) and I(u) are Real and Imaginary parts of the complex function.

1. Complex Function:

$$F(u) = R(u) + jI(u)$$

2. Magnitude or Spectrum:

$$|F(u)| = \sqrt{R^2(u) + I^2(u)}$$

3. Phase:

$$\phi(F(u)) = tan^{-1}\left(\frac{I(u)}{R(u)}\right)$$

2.1.2 Cosine Function and Fourier Transform

Fourier Transform of Cosine Function is implemented in frequency domain. Figure 1 shows the cosine wave and result after applying the Fourier Transform on cosine wave.

$$F\,(\cos(2\pi u_0 x)) = \frac{1}{2}[\delta(u-u_0)+\delta(u+u_0)])$$

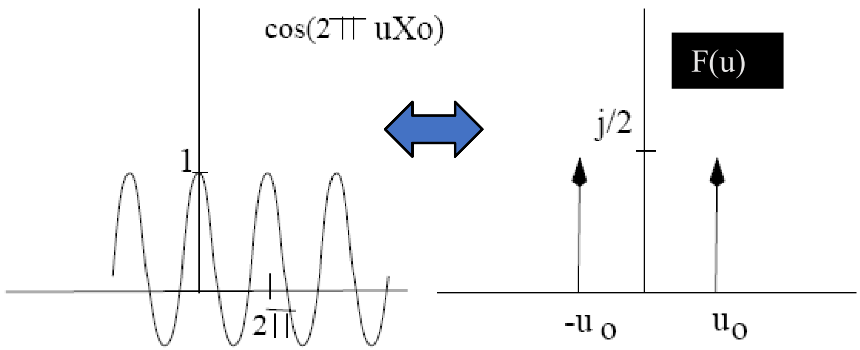

Figure 1: Cosine Wave and Fourier Transform of Cosine Wave

2.1.3 Rectangle Function and Fourier Transform

Fourier Transform of Rectangular function is sin*c* Function. Figure 2 shows that rectangular function, its Fourier Transform and Magnitude of the Fourier Transform. As you can see in the figure Negative Frequencies have no physical meaning and only make sense from a theoretical point of view.

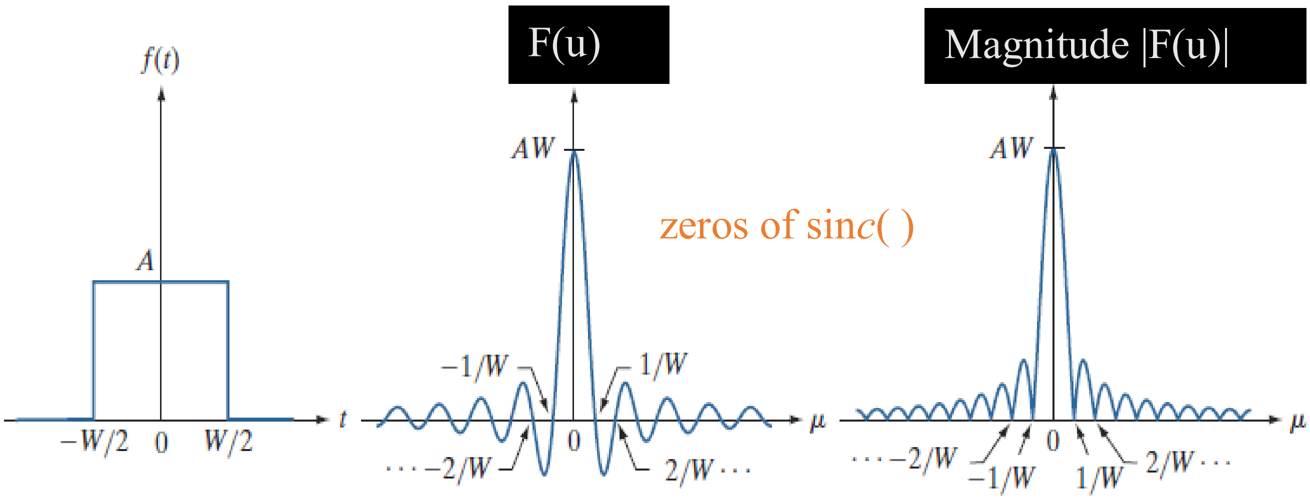

Figure 2: Rectangle Function and Fourier Transform of Rectangle function.

$rect = \frac{x}{W}$, $width\ W$ $\qquad$ $AW\frac{\sin(\pi uW)}{(\pi uW)} = AW\ \mathrm{sinc}(uW)$

2.2 DFT of an Image with Square image

In this Experiment we have a 512 x 512 Image with square 32 x 32 placed at center of the image. For calculating the Fourier Transform of this image and for display the result such as magnitude of the image we need to shift the magnitude using the following Translation property of Fourier Transform. For calculating the 2D DFT, the separability property of Fourier Transform is used which is calculated using the two 1D DFT of rows and columns of given image.

$$\boldsymbol{f(x,y)(-1)^{x+y} = F\,(u - \frac{N}{2} + v - \frac{N}{2})}$$

For visualizing the magnitude |F(u,v)| , we need to apply some stretching transformations since the dynamic range of the magnitude is very large.

$$D(u,v) = c * \log(1 + |F(u,v)|)$$

2.3 DFT of Lenna image:

For Computing the DFT of Lenna image, The Equation for Forward and Inverse transform mentioned in section 2.1 is used. For forward and inverse Fourier transform we are using one single function only for calculating the forward transform we send one extra parameter to function as character “F” and for inverse transform we send character as “I ”.

With the Real, Imaginary part, Magnitude and Phase part of the image we can calculate Forward and Inverse Fourier Transform of the Lenna image.

For Calculating the Fourier Transform of 2D DFT, the separability property is used in which we calculate 1D DFT on rows of given image and then apply Fourier Transform on Columns of image. The following equation presents the separability property of Fourier Transform.

1. Forward 2D DFT

$$F(u,v) = 1/N \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux+vy)/N}$$

2. Inverse 2D DFT

$$f(x,y) = 1/N \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u,v) e^{j2\pi(ux+vy)/N}$$

CHAPTER 3. RESULTS AND DISCUSSION

3.1 Forward and Inverse DFT

3.1.1 DFT of a signal

This section 3.1.1 and Figure 3 shows the result of DFT of a given input signal f = [2, 3, 4, 4]. We calculated the Real, Imaginary parts of the given signal. For the Fourier Transform, we used the equation mentioned in section 2.1 for calculating the Forward and Inverse Transform of a given signal. We also calculated the Magnitude and Phase part of the image using the equation mentioned in section 2.1 for calculating the Magnitude of a Real and imaginary part.

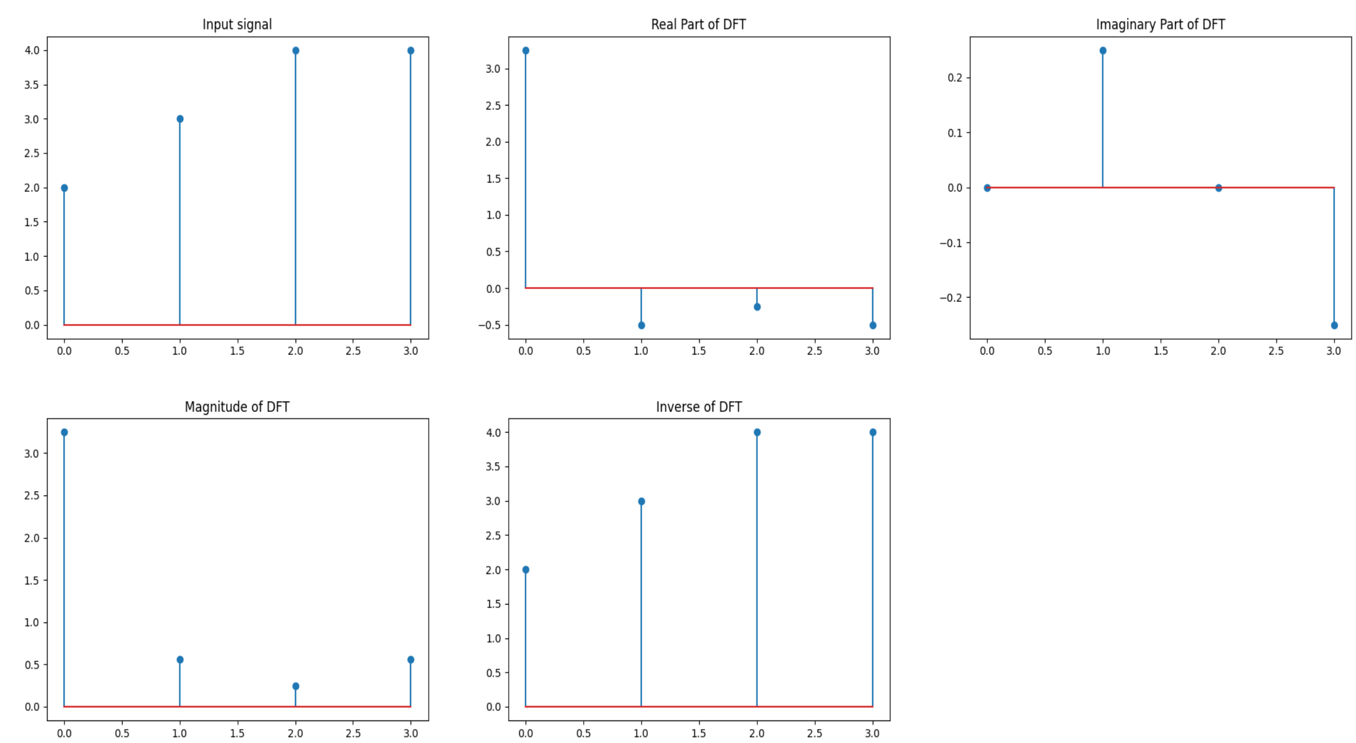

Figure 3: Represents the Input signal, Real part, Imaginary part, Magnitude of DFT and Inverse DFT.

The result as shown in figure 3 To verify the Results of DFT, we calculated the Inverse DFT and we get our input signal f = [2, 3, 4, 4] back as a result.

3.1.2 DFT of Cosine Wave

Section 3.1.2 and figure 4 shows the result of Fourier Transform of one Dimension Cosine wave $f(x) = \cos(2\pi ux)$ with 128 samples and 8 cycles over period. As you can see in figure 4, the Input cosine wave is plot first, then we calculate the real, imaginary part of the cosine wave. We computed the magnitude of the cosine wave using the square root of the real and imaginary part's square and Phase part of the cosine wave using the tan inverse of the Real and imaginary part. We shift the Magnitude of the Cosine wave to center using the translation property mentioned in section 2.2.

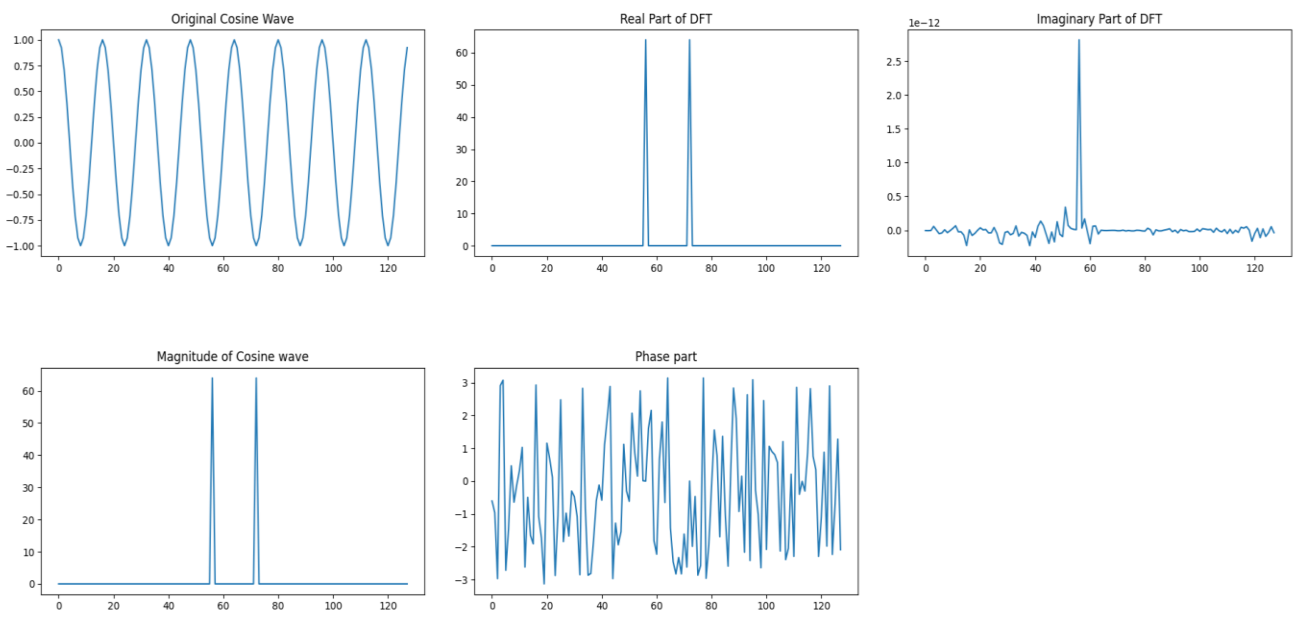

Figure 4: Represents the Original Cosine wave, Real and imaginary part of DFT and Magnitude and Phase part of the DFT.

3.1.3 DFT of Rectangular function

Section 3.1.3 and figure 5 show the result of Fourier Transform of one Dimension Rectangle function with the data given in file Rect_128.dat. As you can see in figure 5, the Input Rectangle function is plotted first, then we calculate the real, imaginary part of the Rectangle function. We computed the magnitude of the Rectangle function using the square root of the real and imaginary part's square and Phase part of the Rectangle function using the tan inverse of the Real and imaginary part. We shift the Magnitude of the Rectangle function to center using the translation property mentioned in section 2.2.

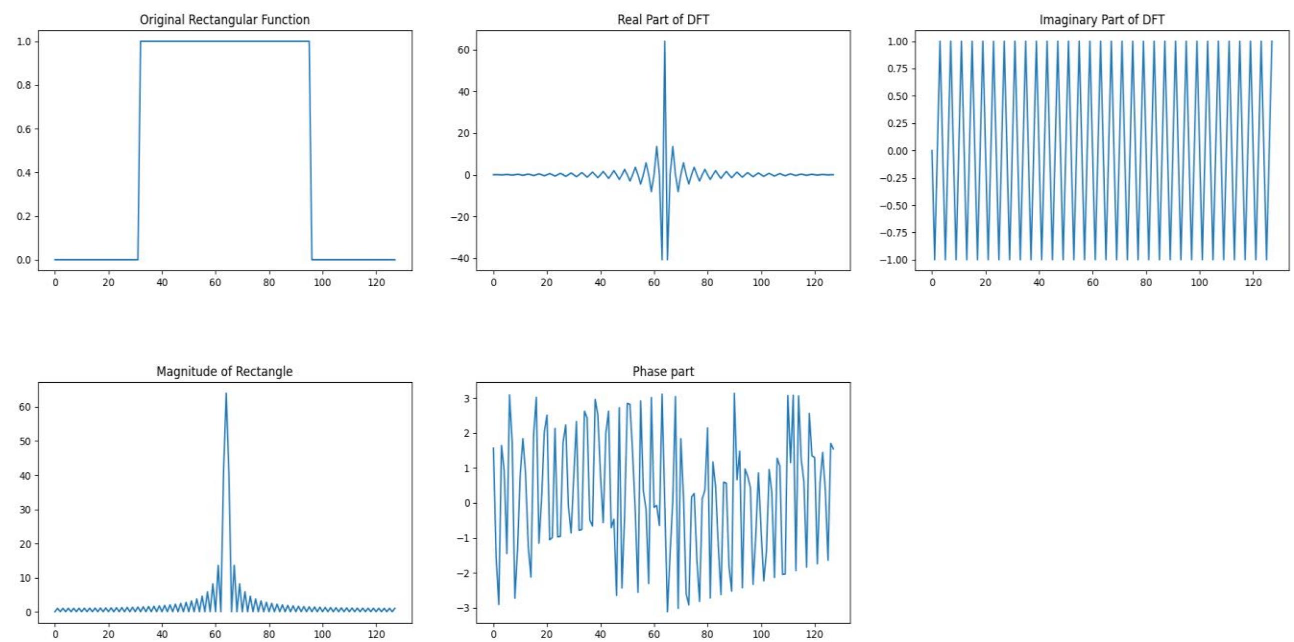

Figure 5: Represents the Original Rectangle Function, Real and imaginary part of DFT and Magnitude and Phase part of the DFT

3.2 Discrete Fourier Transform of Image with Square

3.2.1 DFT of 512 x 512 Image with 32 x 32 Square

This section 3.2.1 shows the result of DFT of 512 x 512 image with 32 x 32 square placed at center of the image. As you can see in Figure 6, first we give input image 512 x 512 with black background (i.e., 0) and square inside it with 32 x 32 size with white (i.e., 255). We applied DFT on this image and figure 6 shows the magnitude of this image without shifting the center of frequency domain which represents the small blurry square.

As we understand in section 2.2 of chapter 2, the magnitude is shifted to the center of frequency domain using Translation property and for proper visualization we used stretching transformation using logarithm function. We can see that there is an increase in signal strength on both horizontal and vertical both axis as it is square.

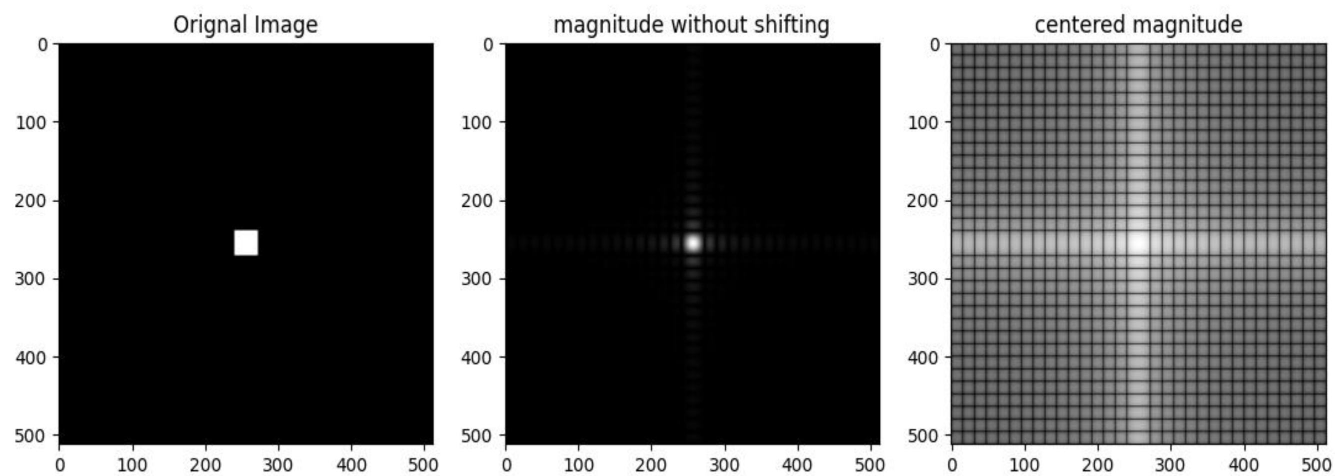

Figure 6: (a) Input image with 32 x 32 square. (b) Magnitude without Shifting (c) Centered Magnitude.

3.2.2 DFT of 512 x 512 Image with 64 x 64 Square

This section 3.2.2 shows the result of DFT of 512 x 512 image with 64 x 64 square placed at center of the image. As you can see in Figure 7, first we give input image 512 x 512 with black background (i.e., 0) and square inside it with 64 x 64 size with white (i.e., 255). We applied DFT on this image and figure 7 shows the magnitude of this image without shifting the center of frequency domain which represents the small blurry square.

As we understand in section 2.2 of chapter 2, the magnitude is shifted to the center of frequency domain using Translation property and for proper visualization we used stretching transformation using logarithm function. We can see that there is an increase in signal strength on both horizontal and vertical both axis as it is 64 x 64 square.

As we increase the square size from 32 x 32 to 64 x 64, the square size in magnitude is smaller than we saw in section 3.2.1 and after shifting the magnitude to center the square visualization in this case gets smaller compared to previous section and the strength of signal is decrease as it gets away from center.

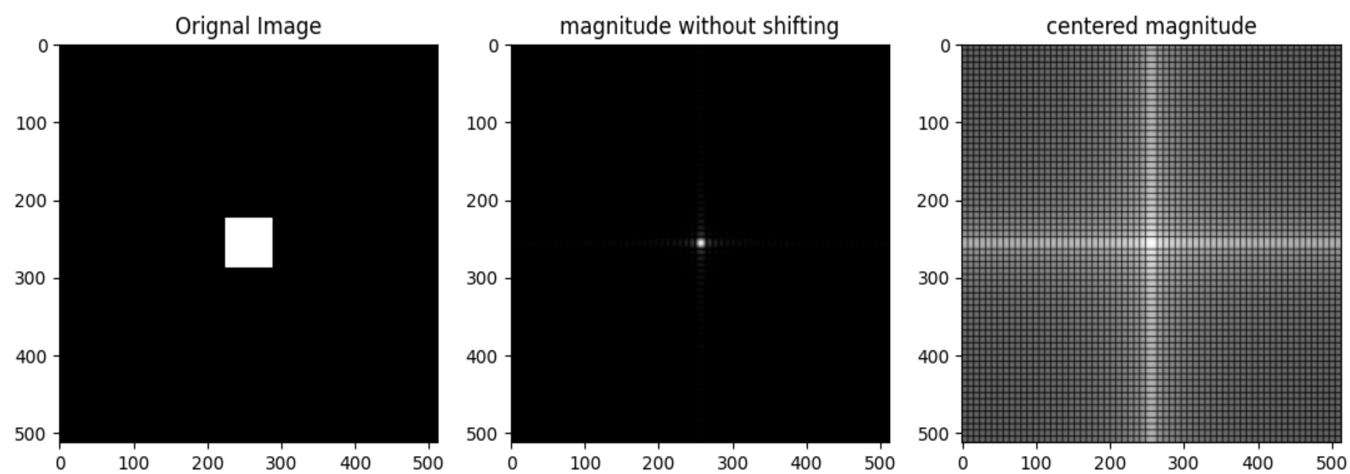

Figure 7: (a) Input image with 64 x 64 square. (b) Magnitude without Shifting (c) Centered Magnitude.

3.2.3 DFT of 512 x 512 Image with 128 x 128 Square

This section 3.2.3 shows the result of DFT of 512 x 512 image with 128 x 128 square placed at center of the image. As you can see in Figure 8, first we give input image 512 x 512 with black background (i.e., 0) and square inside it with 128 x 128 size with white (i.e., 255). We applied DFT on this image and figure 8 shows the magnitude of this image without shifting the center of frequency domain which represents the small blurry square.

As we understand in section 2.2 of chapter 2, the magnitude is shifted to the center of frequency domain using Translation property and for proper visualization we used stretching transformation using logarithm function. We can see that there is an increase in signal strength on both horizontal and vertical both axis as it is 128 x 128 square.

As we increase the square size to 128 x 128, the square size in magnitude is smaller than we saw in section 3.2.1 and 3.2.2. After shifting the magnitude to center the square visualization in this case gets smaller compared to the previous section and the strength of signal decreases as it gets away from center.

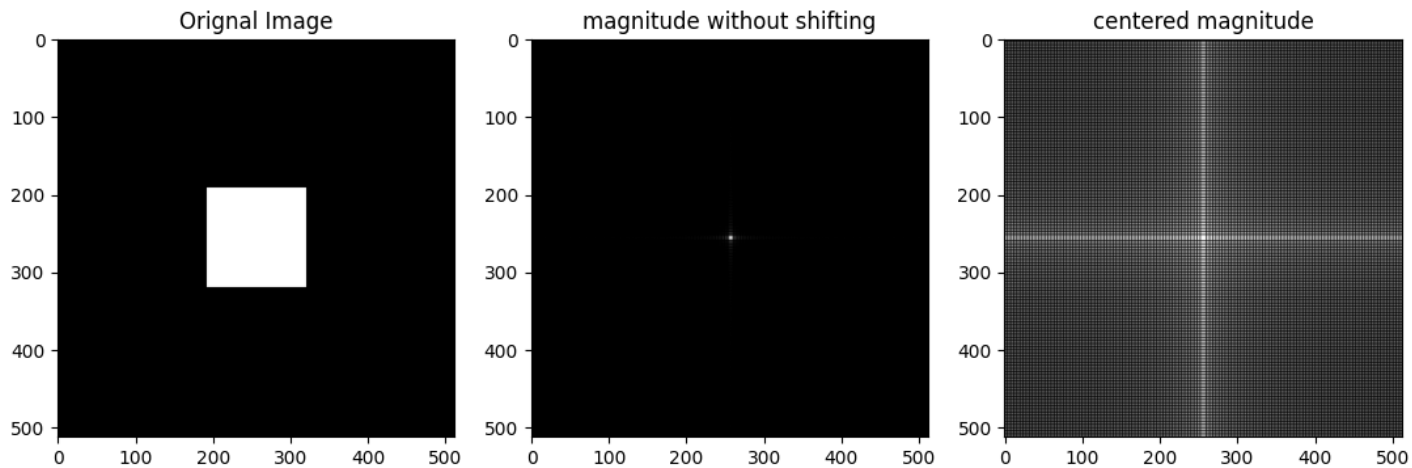

Figure 8: (a) Input image with 128 x 128 square. (b) Magnitude without Shifting (c) Centered Magnitude.

3.3 DFT of Lenna Image

3.3.1 Reconstruction of Lenna image using Inverse DFT with phase zero:

Implementing the DFT on Lenna image involves the Real and Imaginary parts with different values. For Reconstruction, we used the Inverse DFT using magnitude and phase value of Lenna image. As you can see in Figure 9, the Magnitude and Phase part of the Lenna image. The phase part contains more details than magnitude. The reconstruction using phase part contains more details than reconstruction using magnitude.

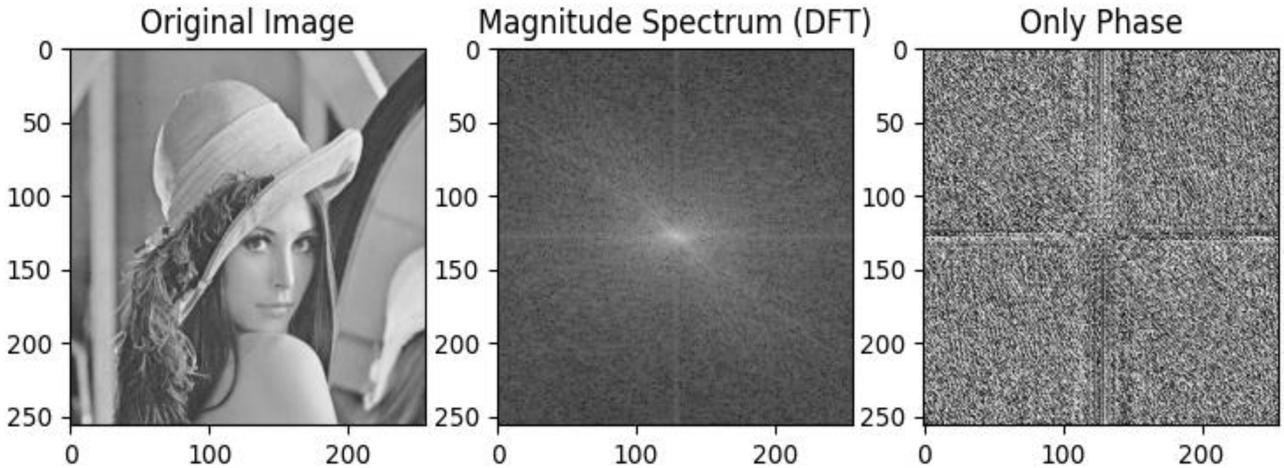

Figure 9: Original Image of Lenna and Magnitude and Phase of Lenna Image.

In 3.3.1 section we implemented using a zero-phase value and for that we set the real part to magnitude and imaginary part to zero. As you can see in figure 10, the reconstructed Lenna image looks nothing like original Lenna image as it is reconstructed only with magnitude, and we set the phase part to zero.

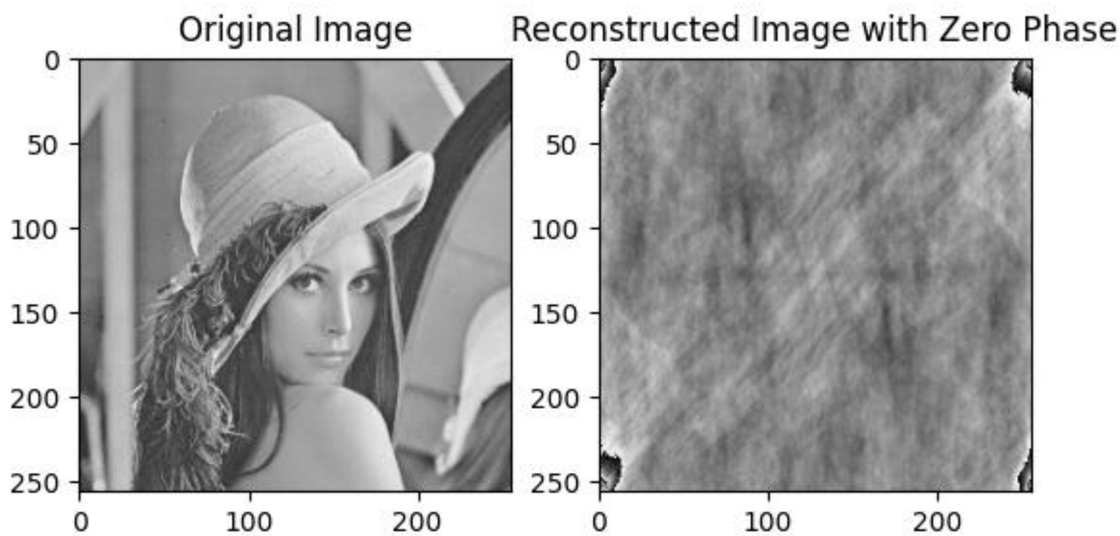

Figure 10: Original Image of Lenna and Reconstrued Lenna image with Phase Zero.

3.3.2 Reconstruction of Lenna image using Inverse DFT with Original phase:

Implementing the DFT on Lenna image involves the Real and Imaginary parts with different values. For Reconstruction, we used the Inverse DFT using magnitude and phase value of Lenna image. In 3.3.2 section we implemented using original phase value and with one Magnitude. For that we set the real part to cos (theta) and imaginary part to sin (theta) where the theta value is phase value. As you can see in figure 11, the reconstructed Lenna image looks like the original Lenna image as it is reconstructed with original phase value and with magnitude one.

The result of this section is better than 3.3.1 section because the phase determines the shift of each sinusoidal component and magnitude determines the strength of each sinusoidal component. The Lenna image constructed using the Phase gives a better result than it constructed using the magnitude.

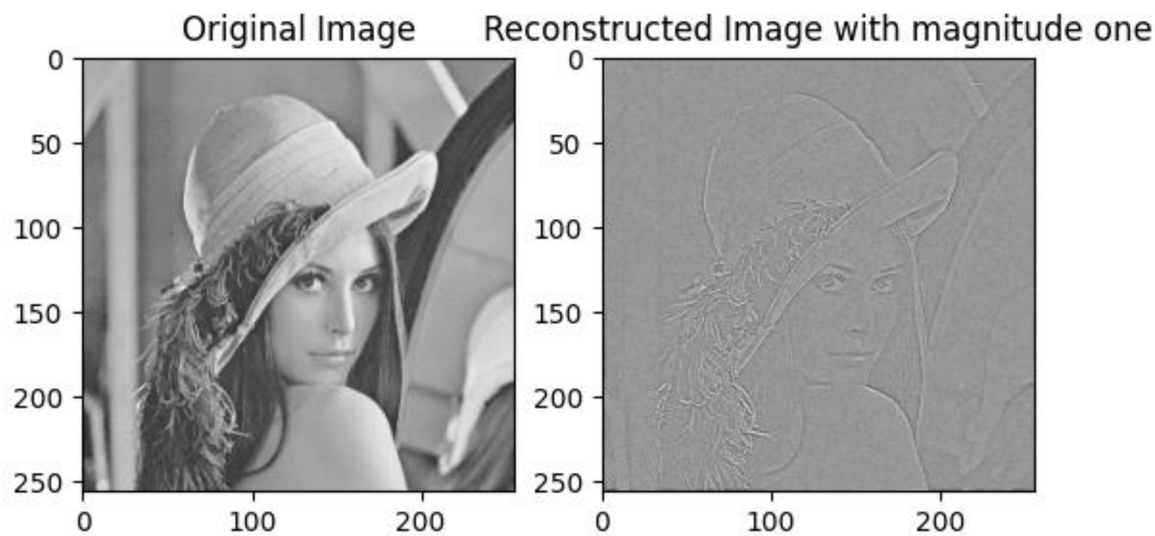

Figure 11: Original Image of Lenna and Reconstrued Lenna image with Original Phase and Magnitude one.

References:

1) https://www.cse.unr.edu/~bebis/CS474/

2) R. Gonzalez and R. Woods Digital Image Processing, 4th edition, Pearson, 2018.