# LAB Manual

(PART A : TO BE REFFERED BY STUDENTS)

Experiment No.06

PART B

(PART B : TO BE COMPLETED BY STUDENTS)

***(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Black board access available)***

|  |  |
| --- | --- |
| Roll No.: C026 | Name: Anirbaan Ghatak |
| Class: B | Batch: B1 |
| Date of Experiment: | Date of Submission |
| Grade : |  |

## Software Code written by student:

#Name: Anirbaan Ghatak

#Roll no.: C026

#Aim: Write a program to apply Discrete Fourier Transform (DFT) to an image and compare the results

import numpy as np

import cv2

from matplotlib import pyplot as plt

# read the input image

# you can specify the path to image

image = cv2.imread("IMG\_2458.jpg", cv2.IMREAD\_GRAYSCALE)

magnitude\_spectrum = 20 \* np.log(cv2.magnitude(dft\_shift[:, :, 0], dft\_shift[:, :, 1]))

# calculating the discrete Fourier transform

DFT = cv2.dft(np.float32(image), flags=cv2.DFT\_COMPLEX\_OUTPUT)

# reposition the zero-frequency component to the spectrum's middle

shift = np.fft.fftshift(DFT)

row, col = image.shape

center\_row, center\_col = row // 2, col // 2

# create a mask with a centered square of 1s

mask = np.zeros((row, col, 2), np.uint8)

mask[center\_row - 30:center\_row + 30, center\_col - 30:center\_col + 30] = 1

# put the mask and inverse DFT in place.

fft\_shift = shift \* mask

fft\_ifft\_shift = np.fft.ifftshift(fft\_shift)

imageThen = cv2.idft(fft\_ifft\_shift)

# calculate the magnitude of the inverse DFT

imageThen = cv2.magnitude(imageThen[:,:,0], imageThen[:,:,1])

# visualize the original image and the magnitude spectrum

plt.figure(figsize=(10,10))

plt.subplot(131), plt.imshow(image, cmap='gray')

plt.title('Input Image'), plt.xticks([]), plt.yticks([])

plt.subplot(132), plt.imshow(magnitude\_spectrum, cmap='gray')

plt.title('Magnitude Specturm'), plt.xticks([]), plt.yticks([])

plt.subplot(133), plt.imshow(imageThen, cmap='gray')

plt.title('Output Image'), plt.xticks([]), plt.yticks([])

plt.show()

## Input and Output:

## 

## Observations and learning:

## *The Discrete Fourier Transform (DFT) is a mathematical technique used to analyze the frequency components.* *In the context of digital images, DFT is applied to understand the frequency content of the image.* *Understood DFT and its properties is essential in image processing and computer vision. It enables us to analyze and manipulate images in the frequency domain, which often leads to improved techniques for image compression, enhancement, and feature extraction.*

## Conclusion:

## *. By transforming images into the frequency domain, we gain insights into their underlying structure and can apply various filters and processing techniques to achieve specific objectives.*

## Question of Curiosity

***(To be answered by student based on the practical performed and learning/observations)***

Q1: Discuss Different Properties of DFT Transform.

Properties of DFT:

Several properties characterize DFT:

**Linearity:** DFT is a linear transformation, which means that it can be applied to individual image components or combined images (e.g., addition or subtraction of two images) in a straightforward manner.

**Shift Invariance:** DFT is shift-invariant, meaning that shifting an image in the spatial domain results in a phase shift in the frequency domain.

**Periodicity:** The DFT of an image exhibits periodicity, with the period determined by the size of the image.

**Complex Conjugate Symmetry:** For real-valued images, the DFT exhibits symmetry properties, with complex conjugate symmetry for the magnitude and antisymmetry for the phase.

Q2: Discuss any one application of DFT/FFT in detail.

Image filtering in the frequency domain involves transforming an image into the frequency domain using the DFT or FFT, applying filtering operations to modify its frequency components, and then transforming it back to the spatial domain. This process allows for various image enhancement and manipulation tasks. Here's how it works:

Transforming to the Frequency Domain: The input image is initially transformed from the spatial domain (pixel values) to the frequency domain using the DFT or FFT. This results in a complex-valued frequency representation of the image.

Filtering Operation: In the frequency domain, specific frequency components can be modified or filtered to achieve various image processing goals.

Inverse Transformation: After filtering, the modified image is transformed back to the spatial domain using the inverse DFT or FFT. This yields the processed image with the desired enhancements or modifications.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*