**SVKM’s NMIMS**

**Mukesh Patel School of Technology Management & Engineering**

**Department of Electronics and Telecommunication Engineering**

**Subject: Image and Video Processing Program: B.Tech/BTI/MBA**

**Sem: VII/IX/V ACAY: 2020-21**

**EXPERIMENT NO. 6**

**Aim:**

1. To write a program in PYTHON to implement FFT and IFFT on an image
2. To write a program in PYTHON implement LPF and HPF in frequency domain on an image

**Software:**  PYTHON.

**Prerequisite:**

|  |  |
| --- | --- |
| Sr. No | Concepts |
| 1. | Discrete Fourier Transform and Filtering in Frequency domain |

**Outcome:**

After successful completion of this experiment students will be able to:

Concept of filtering in frequency domain in image processing

**Theory:**

**Discrete Fourier Transform**

Discrete Fourier Transform (DFT) is used for performing frequency analysis of discrete time signals. DFT gives a discrete frequency domain representation whereas the other transforms are continuous in frequency domain.

The N point DFT of discrete time signal x[n] is given by the equation



Where N is chosen such that , where L=length of x[n].

Here the twiddle factor matrix is given by

We can find DFT of the image by using following matrix notation.

The inverse DFT allows us to recover the sequence x[n] from the frequency samples.



**Basic steps in frequency domain filtering:**

1. Multiply the input image by as follows:
2. Compute of the image obtained in step (1) using DFT
3. Multiply by a filter function
4. Computer inverse DFT of the result obtained in step (3)
5. Obtain real part of the result obtained in step (4)
6. Multiply the result in (5) by

For ideal LPF,

For ideal HPF,

For Butterworth HPF,

For Butterworth LPF,

is the distance from the point to the origin of the frequency rectangle for an MXN image.

Therefore, for an image when u=M/2, v=N/2, D(u,v)=0This centres our H(u,v). Here H(u,v) is our filter transfer function.

|  |
| --- |
| Name of the Experiment: To implement FFT and IFFT of an image. |
| Roll No. C018 Name: Dhruvit Jain |
| Program: B.Tech ExTC Semester : VII |
| Date of Performance:14/08/2020 Date of Submission: 14/08/2020 |

**CODE:**

from skimage import io

import matplotlib.pyplot as plt

import numpy as np

from skimage.color import rgb2gray

image\_ori = io.imread('C:/Users/dhruv/Desktop/College/NOTES/IVP/Labs/6/home.jpeg')

image=rgb2gray (image\_ori)

plt.figure()

plt.subplot (1,2,1)

io.imshow(image\_ori)

plt.title("Original Image")

plt.subplot (1,2,2)

io.imshow(image)

plt.title("Gray Image")



sh=image.shape

row=sh[0]

col=sh[1]

for r in range (0,row):

for c in range(0,col):

image[r][c]=((-1)\*\*(r+c))\*image[r][c]

im\_fft=np.fft.fft2(image)

sh=im\_fft.shape

#im\_fft[0:4,0:4]

im\_fft\_abs=np.abs(im\_fft)

im\_abs\_log=np.log10(1+im\_fft\_abs)

#im\_fft\_abs[0:4,0:4]

im\_fft\_phase=np.angle(im\_fft) #angle is in radians

#im\_fft\_phase[0:4,0:4]

plt.figure(figsize=(7,7))

plt.subplot(2,2,1)

io.imshow(image,cmap='gray')

plt.title('Gray Image')

plt.subplot(2,2,2)

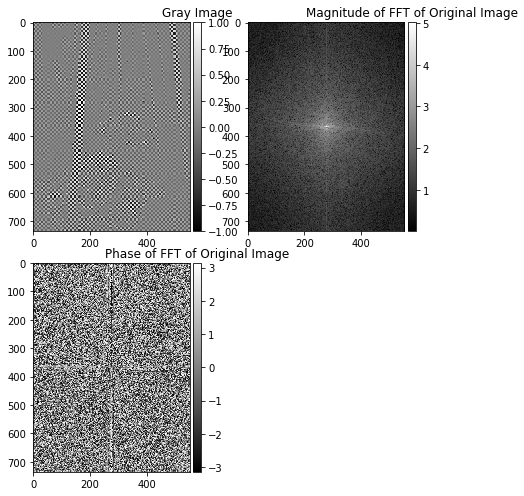
io.imshow(im\_abs\_log,cmap='gray')

plt.title('Magnitude of FFT of Original Image')

plt.subplot(2,2,3)

io.imshow(im\_fft\_phase, cmap='gray')

plt.title('Phase of FFT of Original Image')



mx=np.max(np.max(im\_fft\_abs))

mx

mn=np.min(np.min(im\_fft\_abs))s

mn

mx=np.max(np.max(im\_abs\_log))

mx

mn=np.min(np.min(im\_abs\_log))

mn

# Inverse FFT to get the original image

im\_inv=np.fft.ifft2(im\_fft)

im\_inv=np.real(im\_inv)

im\_post=im\_inv.copy()

for r in range (0,row):

for c in range(0,col):

im\_post[r][c]=((-1)\*\*(r+c))\*im\_inv[r][c]

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

plt.subplot(1,2,1)

io.imshow(im\_inv, cmap='gray')

plt.title('Inverse of FFT of Preprocessed Image')

plt.subplot(1,2,2)

io.imshow(im\_post, cmap='gray')

plt.title('Post Processed Image')



**CONCLUSIONS:**

* If we take inverse of an image by considering only magnitude or only phase, then we do not get the original image. To get the original image, we have to take both phase and magnitude.
* Log of absolute value of FFT is taken to reduce the dynamic range and increase variations.
* For better visual representation of the magnitude of the FFT, we should centre the image.
* After taking inverse of centered FFt, we require post processing to reverse the effect of pre processing.