DSAA ASSIGNMENT2 - REPORT

PARYUL JAIN
20171083
IIIT-H

```
t = timer('TimerFcn', 'stat=false; disp(''Timer!'')',...
                 'StartDelay',10);
start(t)

stat=true;
while(stat==true)
%   disp('.')
  pause(1)
end
clear all; close all; clc
```

The FFT function will return a complex double array. If you read in a .JPG or a .TIF file, you will notice that they are UINT8 arrays. So, you will have to take the real part of the IFFT and then convert it back into UINT8.

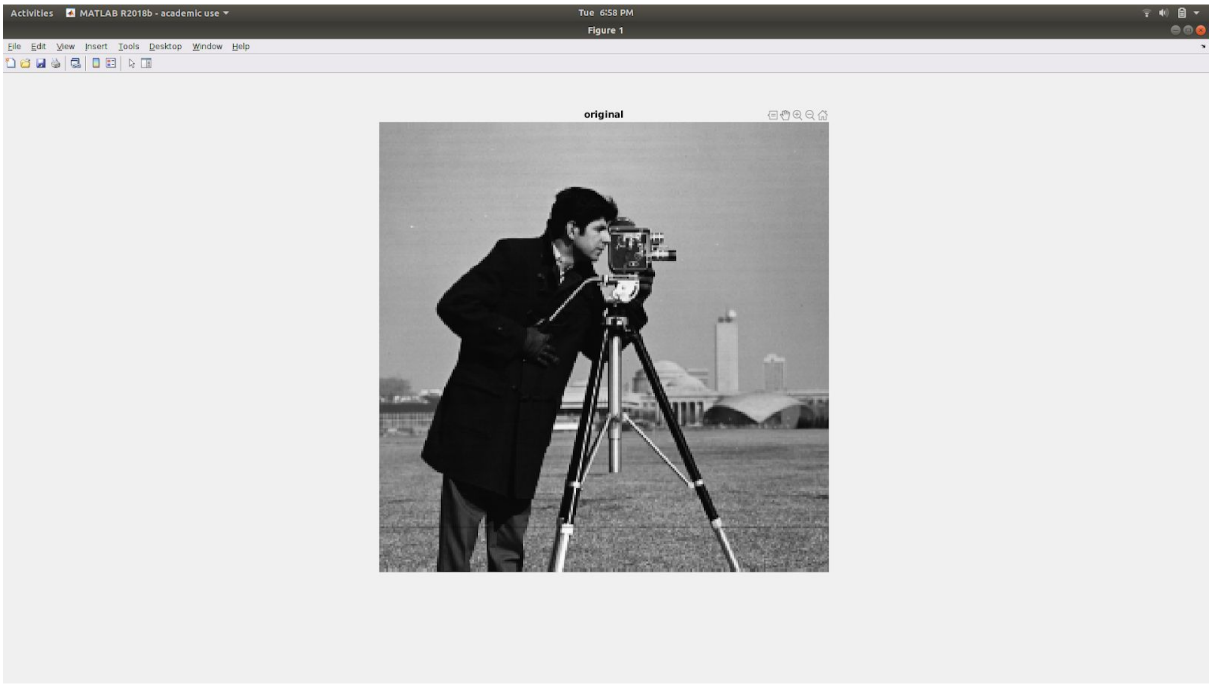
```
new_image = flipdim(original_image,1) // vertical flip
new_image = flipdim(original_image,2) // horizontal flip
```

----

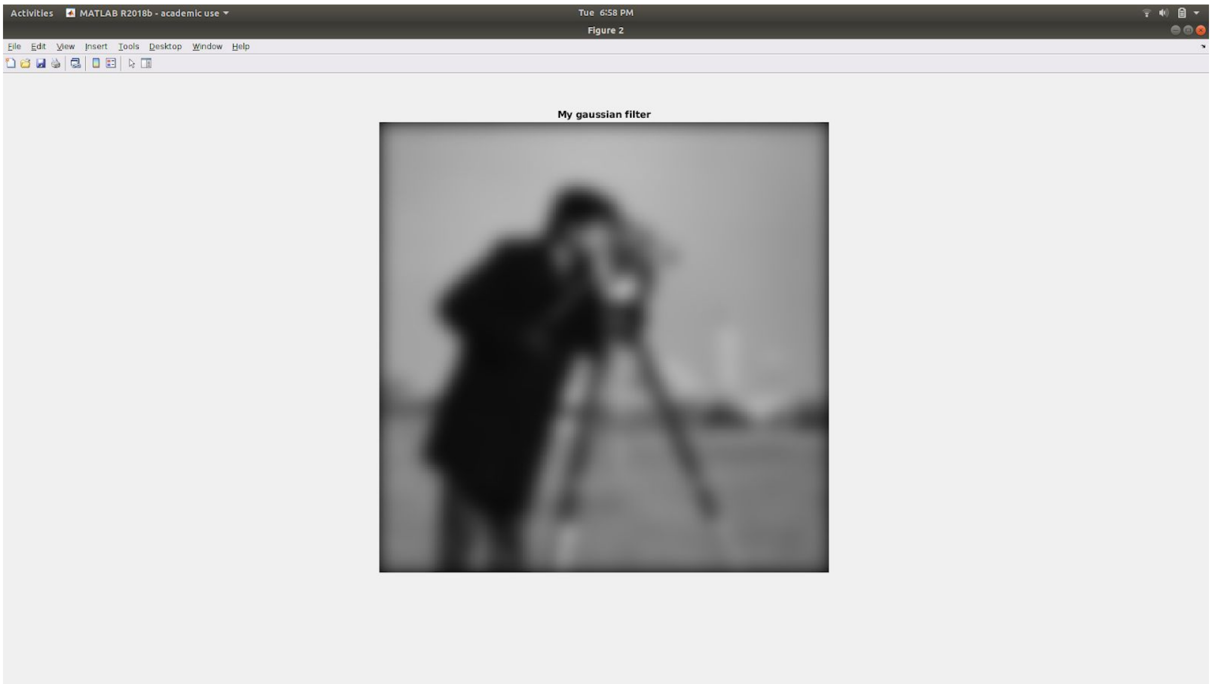
dftmtx(N) -> gives the W matrix

QUESTION-1

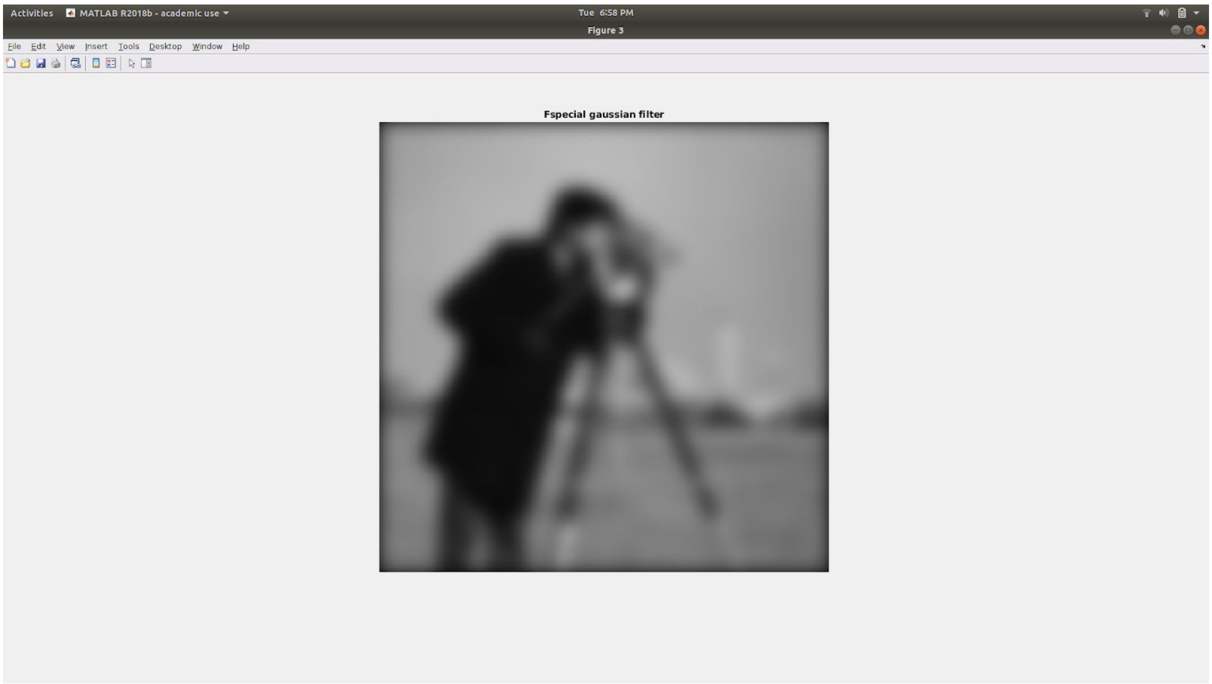
PART 1,3


Activities
MATLAB R2018b - academic use
Tue 6:58 PM
Figure 1
File Edit View Insert Tools Desktop Window Help
original

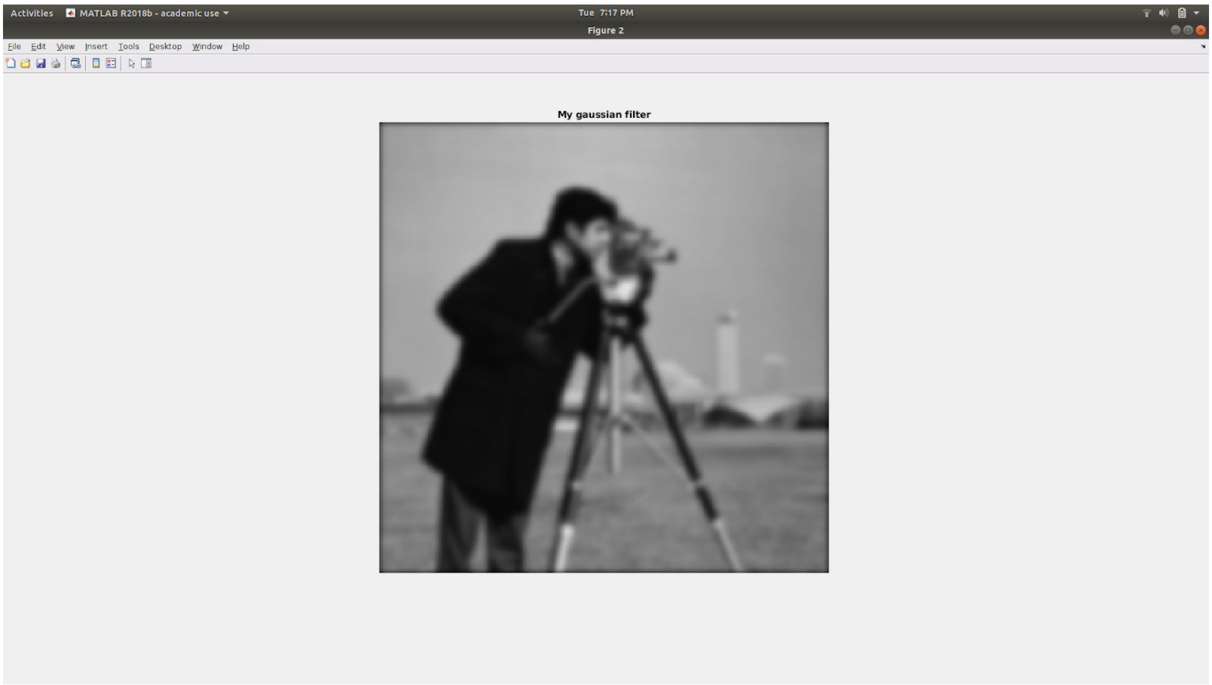
cameraman.tif -> ORIGINAL IMAGE


Activities
MATLAB R2018b - academic use
Tue 6:58 PM
Figure 2
File Edit View Insert Tools Desktop Window Help
My gaussian filter

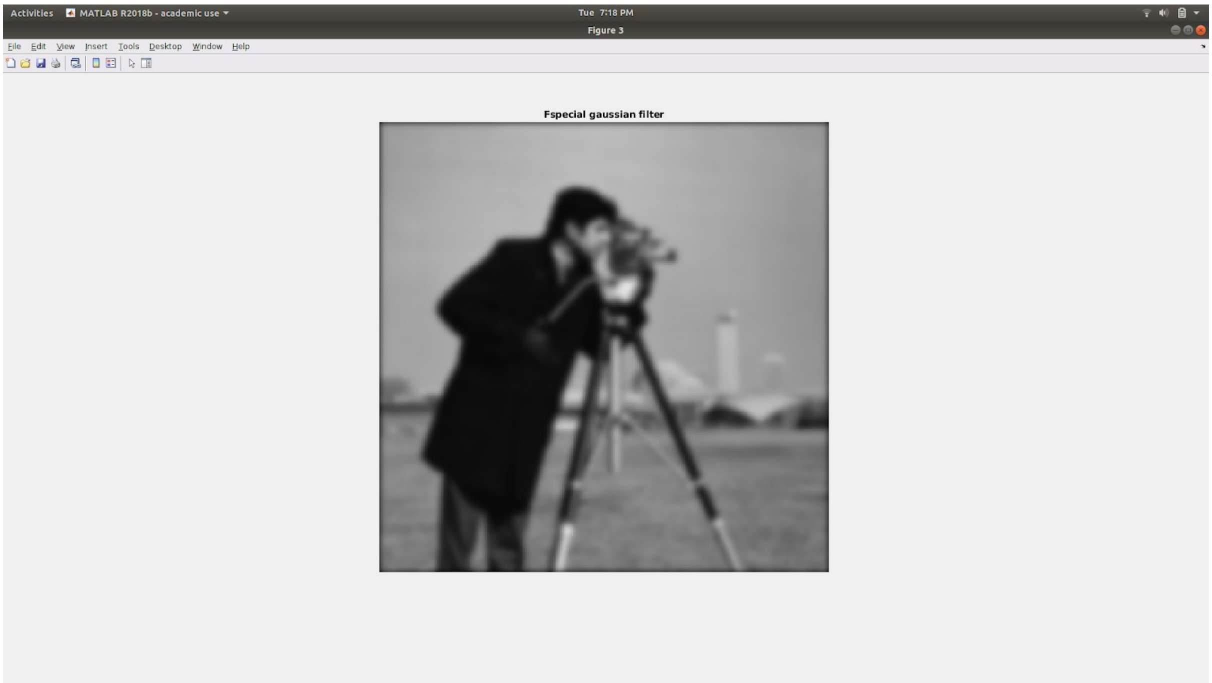
USING IMPLEMENTED GAUSSIAN FILTER with N=100, SIGMA=5


Activities
MATLAB R2018b - academic use
Tue 6:58 PM
Figure 3
File Edit View Insert Tools Desktop Window Help
Fspecial gaussian filter

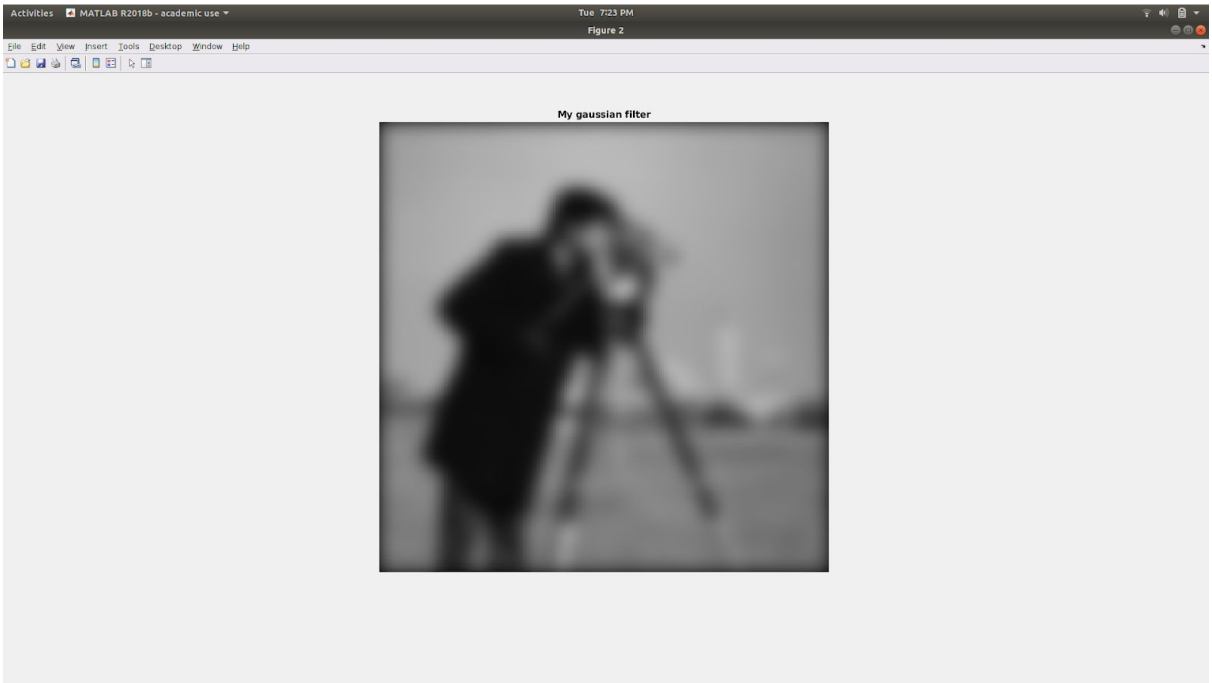
USING FSPECIAL(creates predefined 2d filter). GAUSSIAN FILTER with N=100, SIGMA=5


Activities
MATLAB R2018b - academic use
Tue 7:17 PM
Figure 2
File Edit View Insert Tools Desktop Window Help
My gaussian filter

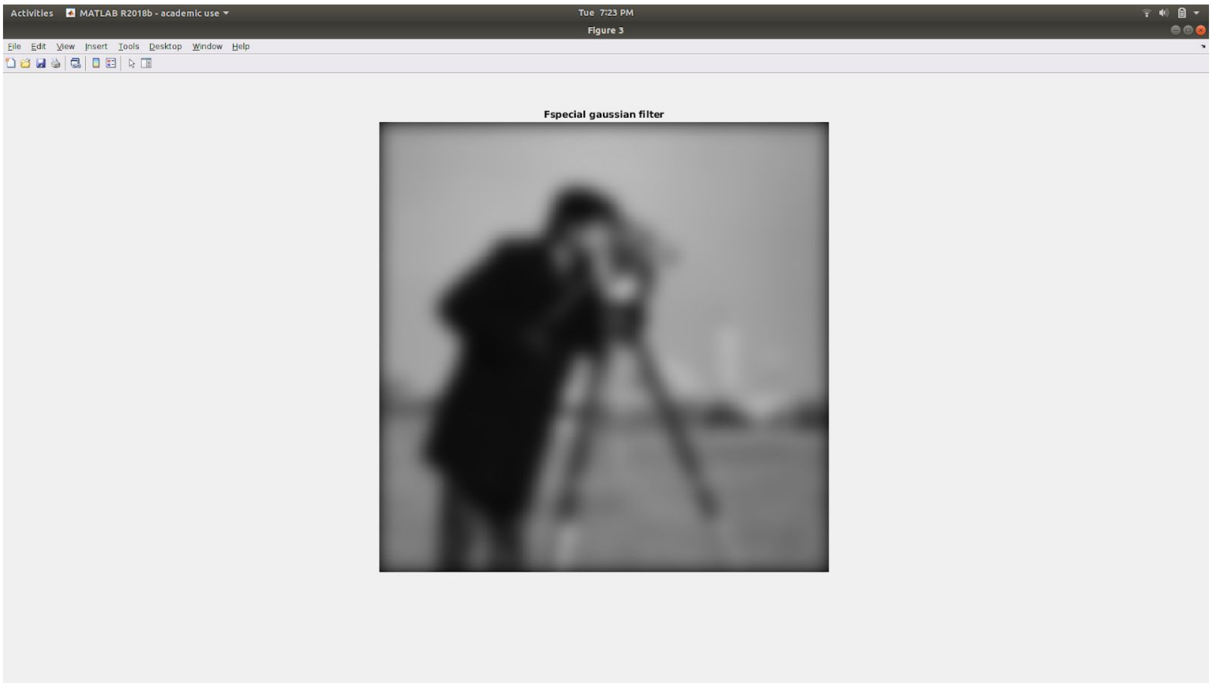
USING IMPLEMENTED GAUSSIAN FILTER with N=100, SIGMA=2


Activities
MATLAB R2018b - academic use
Tue 7:18 PM
Figure 3
File Edit View Insert Tools Desktop Window Help
Fspecial gaussian filter

USING FSPECIAL(creates predefined 2d filter). GAUSSIAN FILTER with N=100, SIGMA=2


Activities
MATLAB R2018b - academic use
Tue 7:23 PM
Figure 2
File Edit View Insert Tools Desktop Window Help
My gaussian filter

USING IMPLEMENTED GAUSSIAN FILTER with N=1000, SIGMA=5


Activities
MATLAB R2018b - academic use
Tue 7:23 PM
Figure 3
File Edit View Insert Tools Desktop Window Help
Fspecial gaussian filter

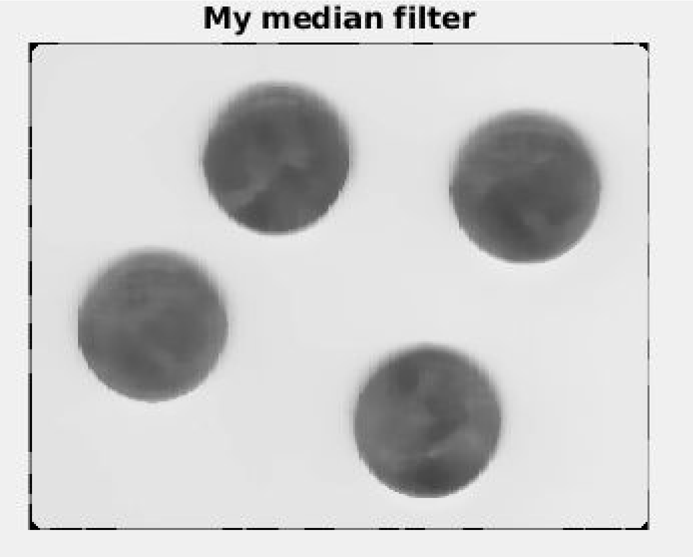
USING FSPECIAL(creates predefined 2d filter). GAUSSIAN FILTER with N=1000, SIGMA=5

As we see that when we increase sigma , the distortion increases in both the implemented as well as the fspecial generated filter.
Also when we increase N , the value at any point is generated using values from very large areas, so we can see that distortion comes into play.

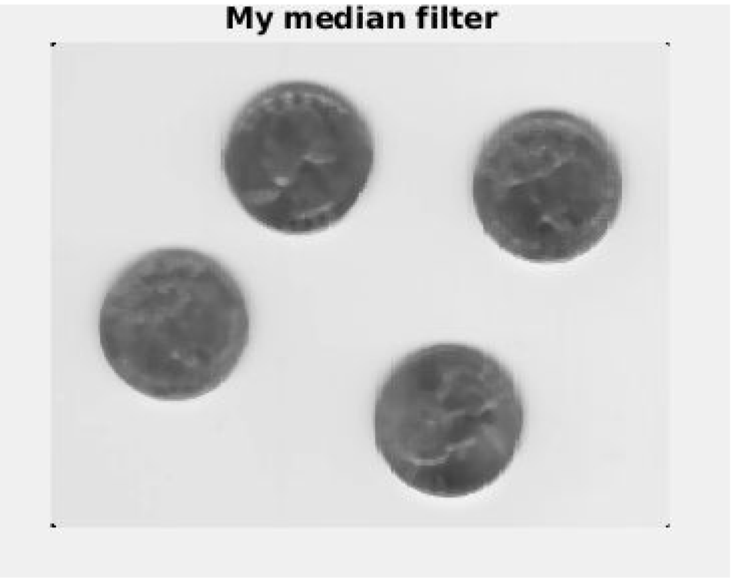
The only difference i could infer was that the values by the implemented gaussian filter tend to be lesser than the values obtained after passing thru the fspecial filter.

PART 2,3

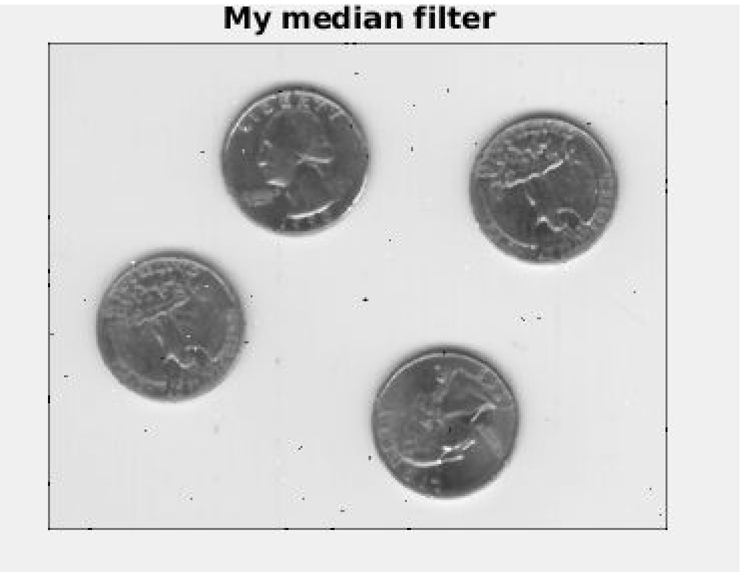
Designed median filter with N=10 (WINDOW SIZE);



Designed median filter with N=5 (WINDOW SIZE);

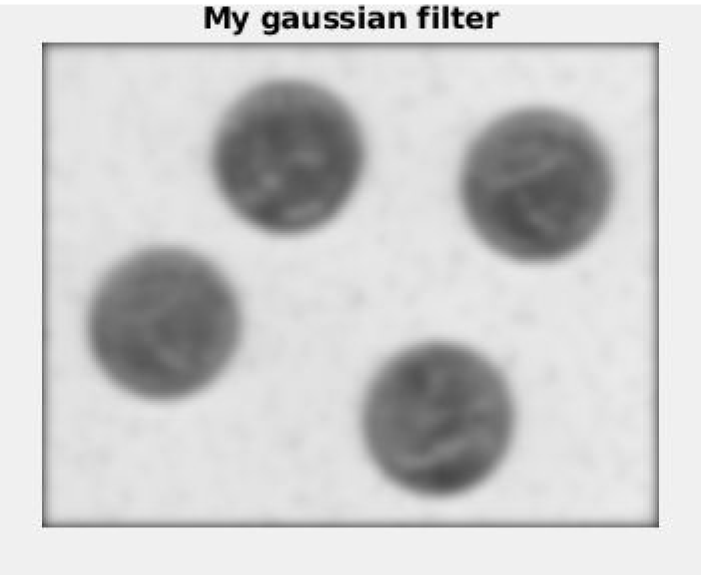


Designed median filter with N=2 (WINDOW SIZE);

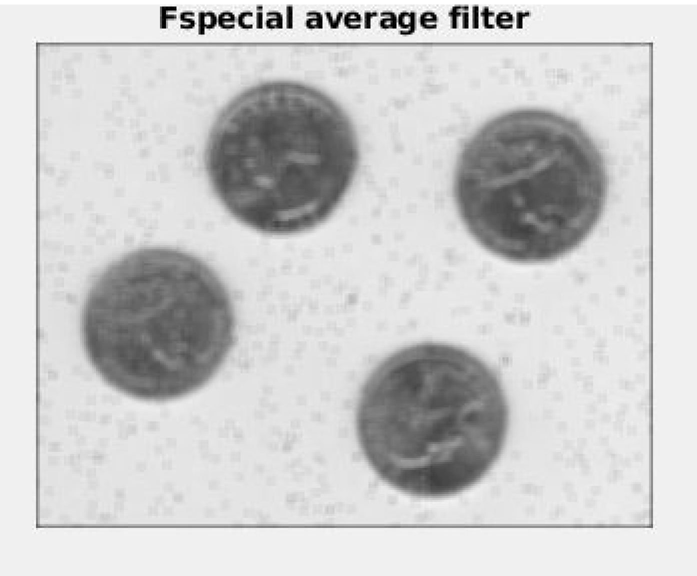


Median filter helps remove outliers. Since the median of a window is unaffected by an outlier , it helps remove ‘salt and pepper’ noise.

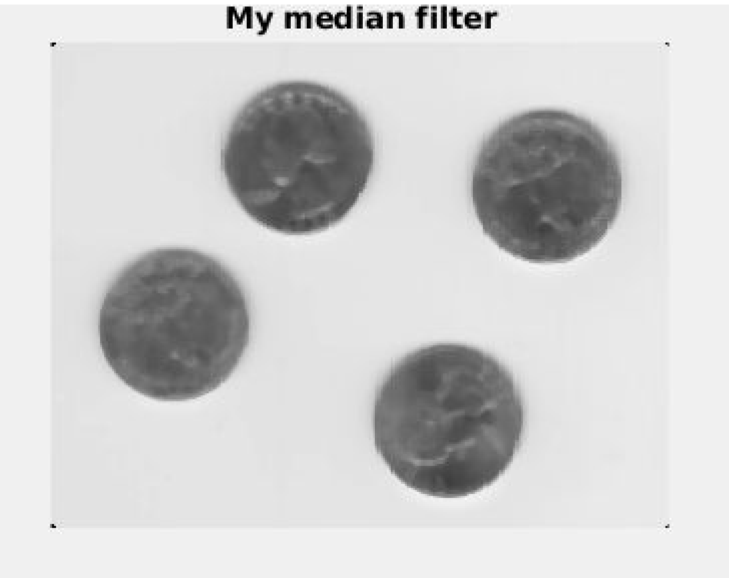
PART 4



My gaussian filter



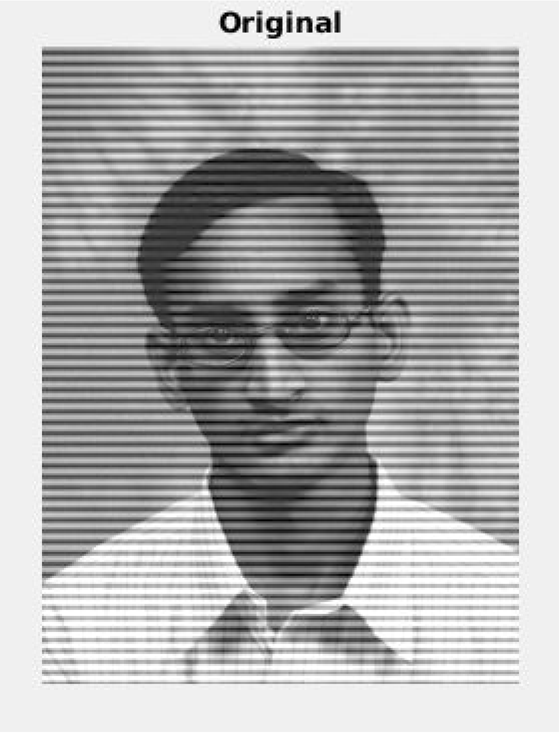
Fspecial average filter

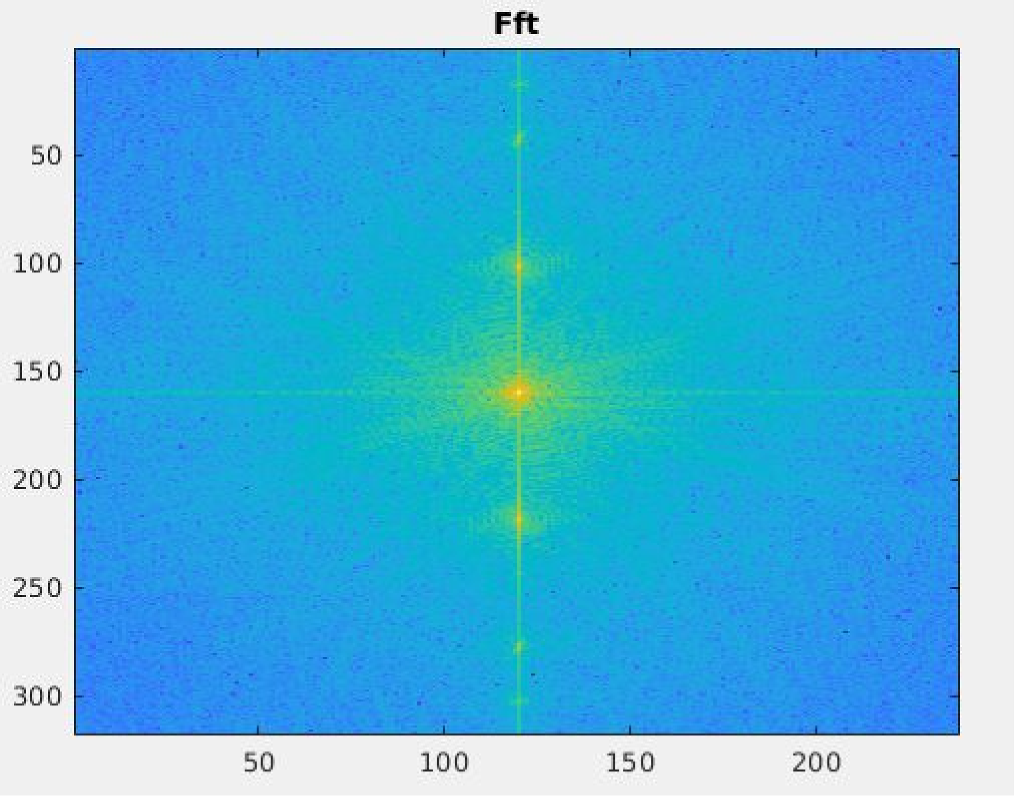


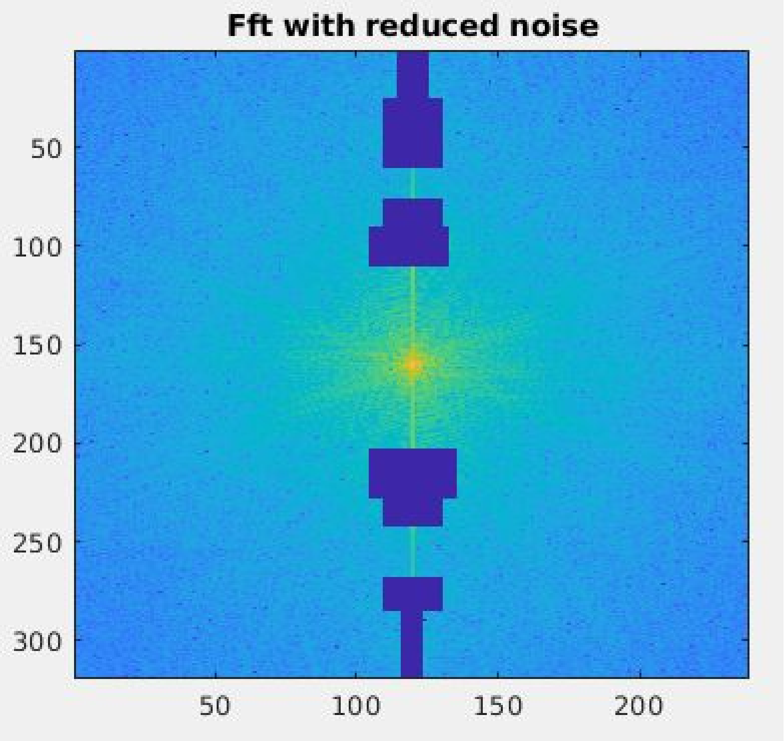
My median filter

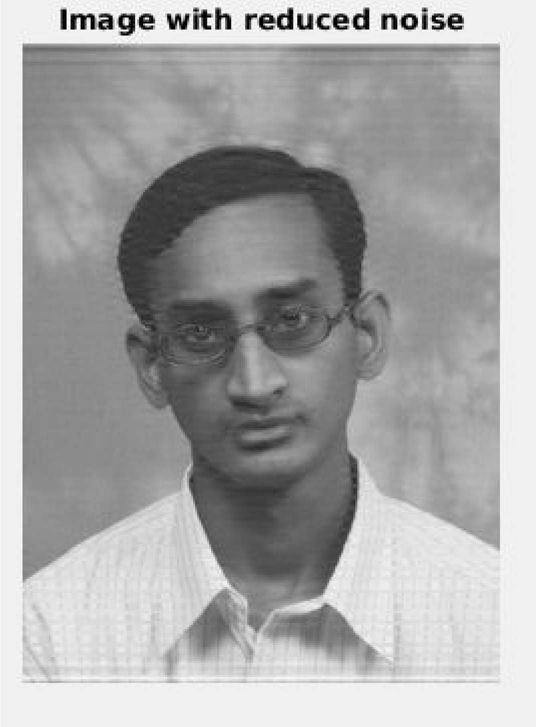
Median filter is the best as it removes salt and pepper noise from inp1.png

PART 5









Notch filter will be best suitable .Since we can see the noise in figure 2, we need to remove the specific noise . So we filter out specific frequencies.

--xxxxxxxxxxxxxx---

QUESTION 2

A.

After convolution the height ,width and channels of the output are as follows.

H = Height of the original image
W = Width of the original image
C = Channels
F = Filter Height and Width
Z = Zero Padding.
N = Number of filters.

New_height = floor((H+ 2*Z -F)/S)+1;

New_width = floor((W+ 2*Z -F)/S)+1;

New_channels = N ;

B.

So for each element there will be F*F*C multiplications and (F*F*C-1) additions.

And now since the total number of elements in the resultant matrix are

T = New_height * New_width * N;

So total number of Operations of multiplications will be = T * F * F * c; and

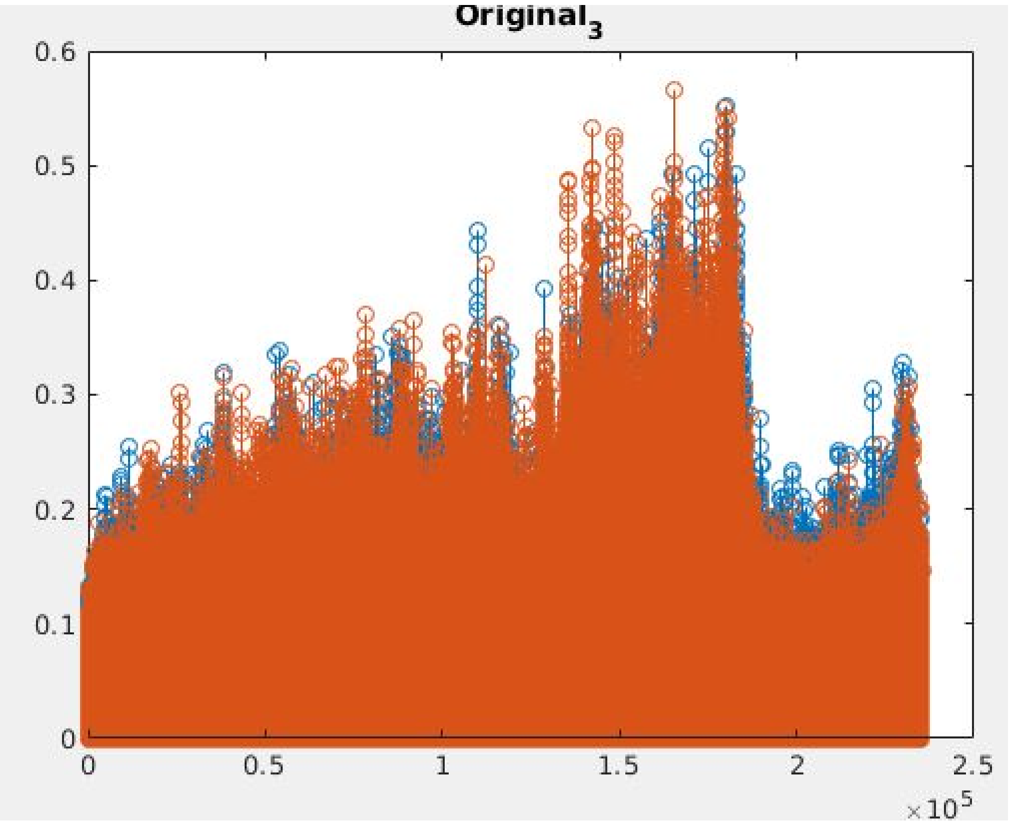
The total number of Operations of additions will be = T * ((F * F * C) - 1);

---xxxxxxxxxxxxxxx---

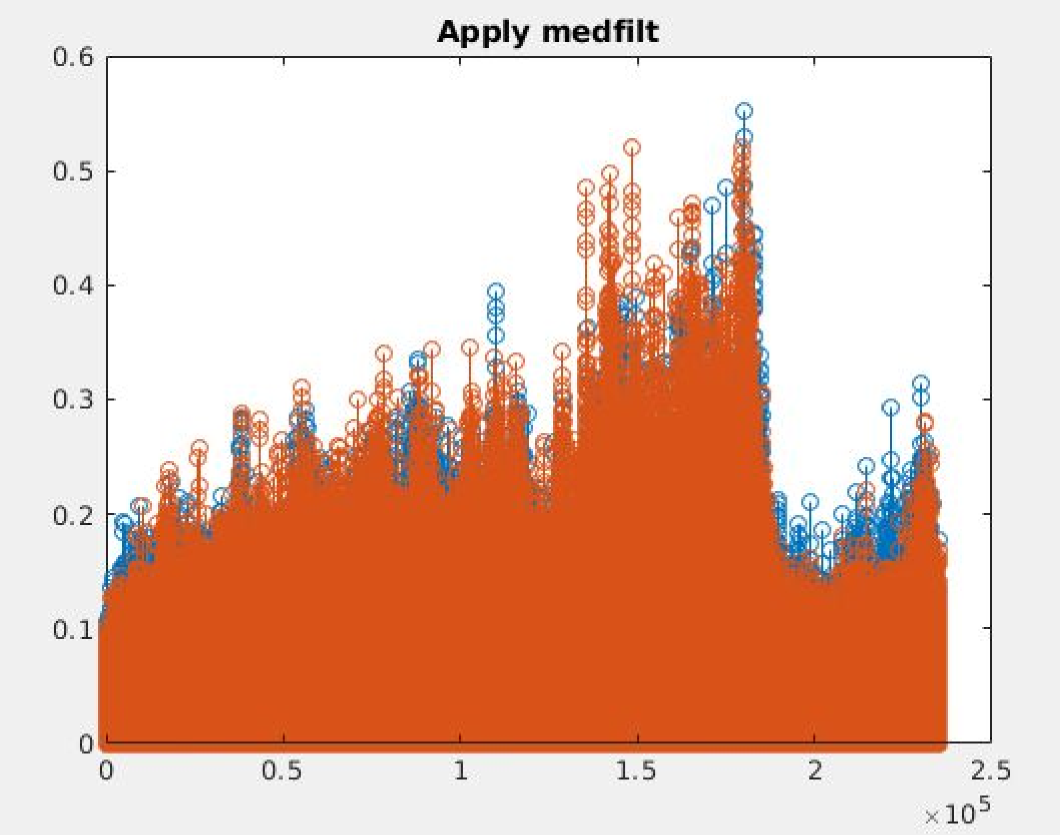
QUESTION 3

---xxxxxxxxxxxxxxx---

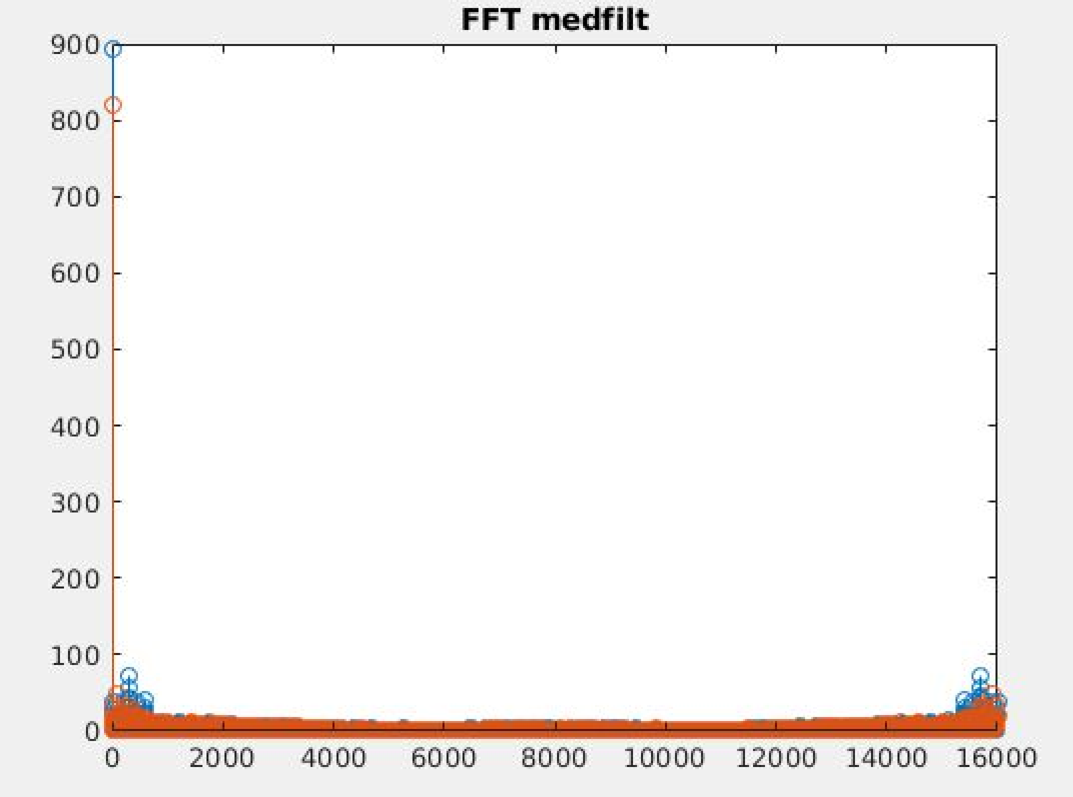
QUESTION 4



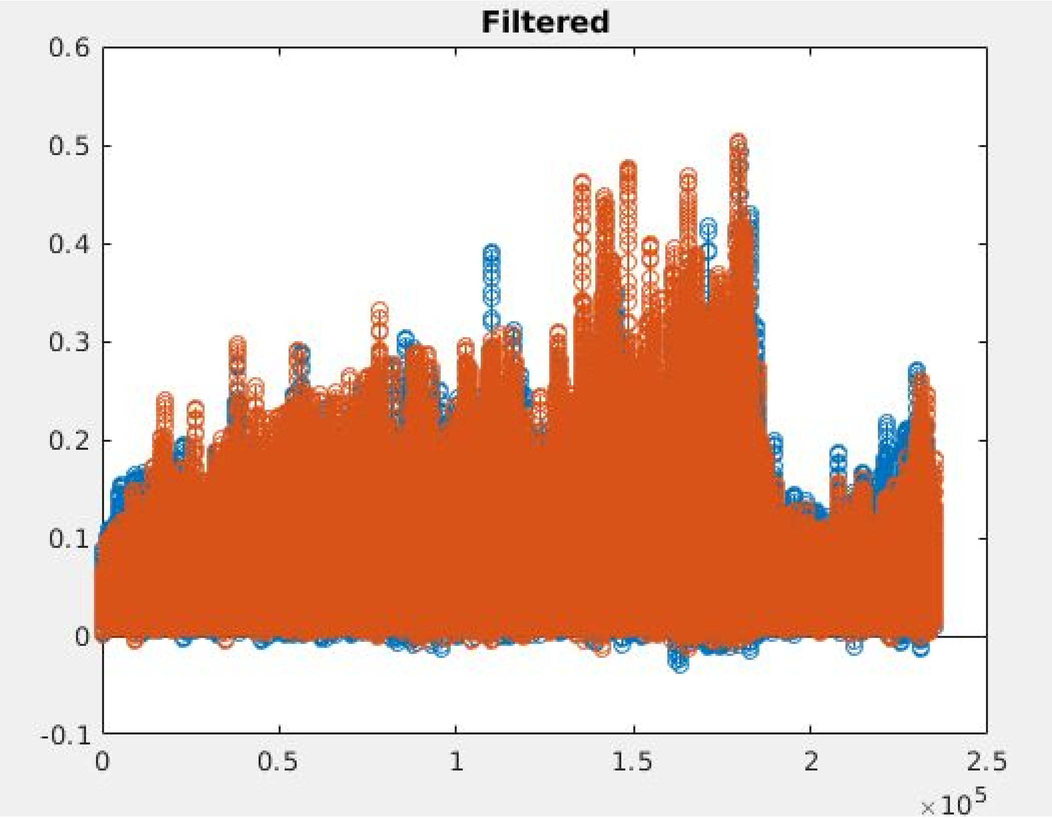
I first apply median filter on this to remove some of the outliers.



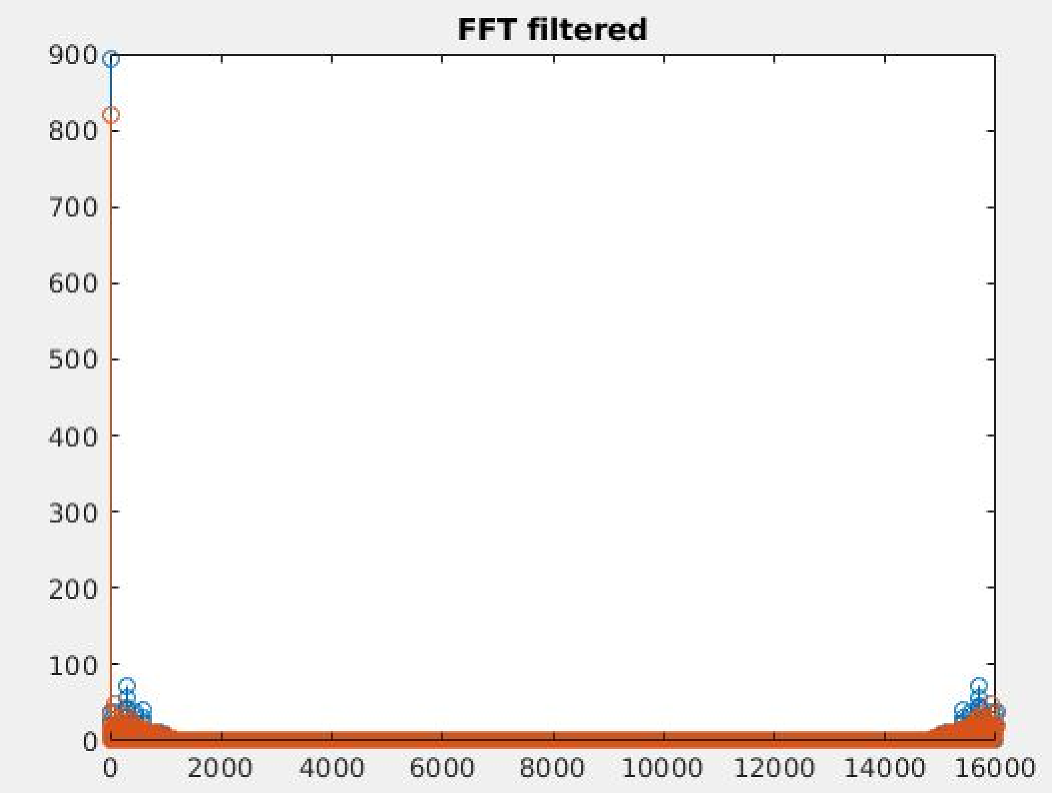
We can see it removes some of the noise.



FFT AFTER TAKING THE MEDIAN FILTER



Filtered signal after passing it through the customised low pass filter



FFT OF THE FILTERED SIGNAL

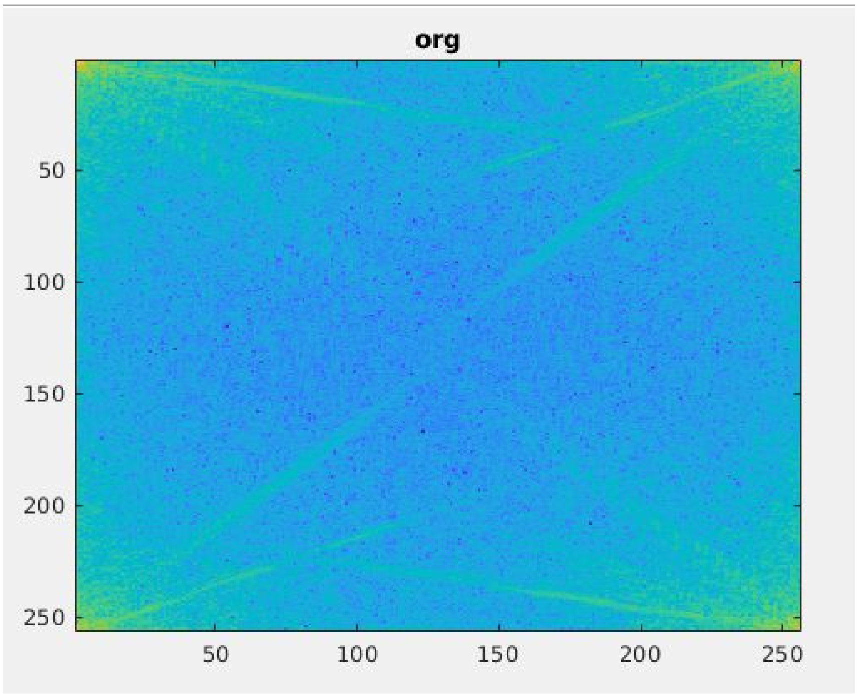
As we can see that the frequencies above 1000 Hz are chopped off by the customised low pass filter designed by buffer. We can the smoothness in the mid area from 1000 - 15000 . Since all the frequencies(1000-8000) are not allowed.

--xxxxxxxxxxxxxx---

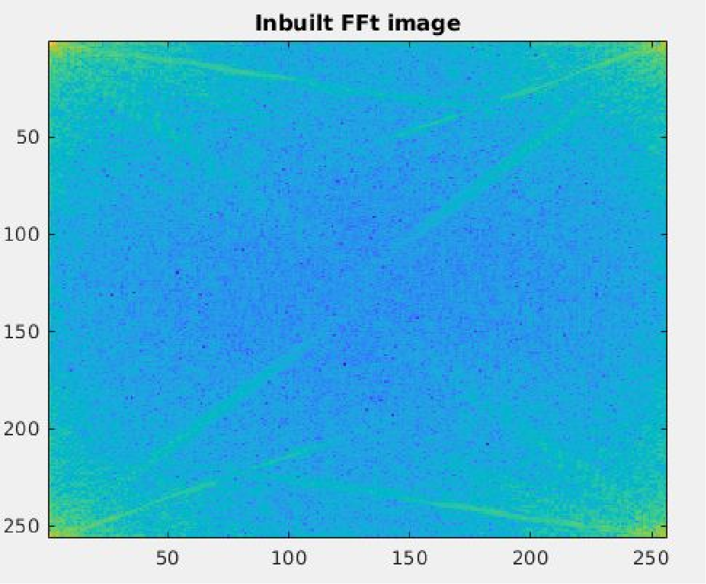
QUESTION 5

Original Image

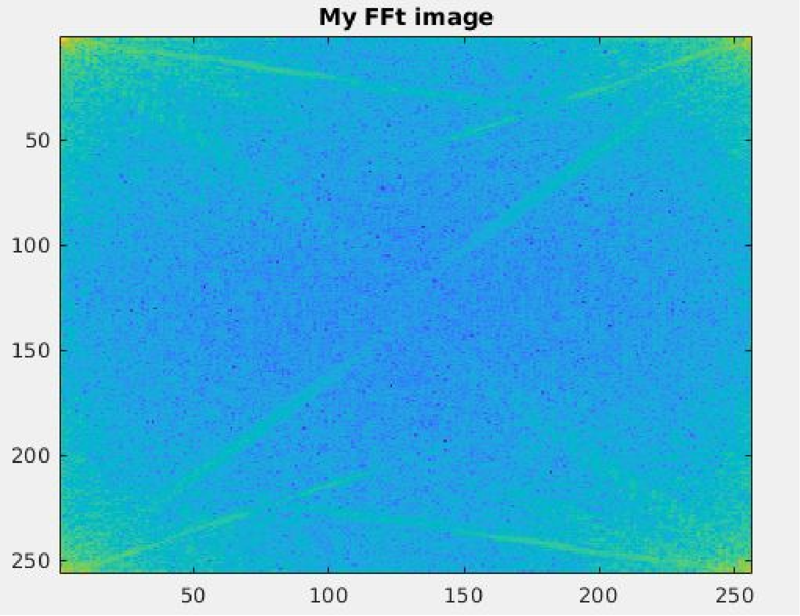
Output of Implemented DFT on 256*256



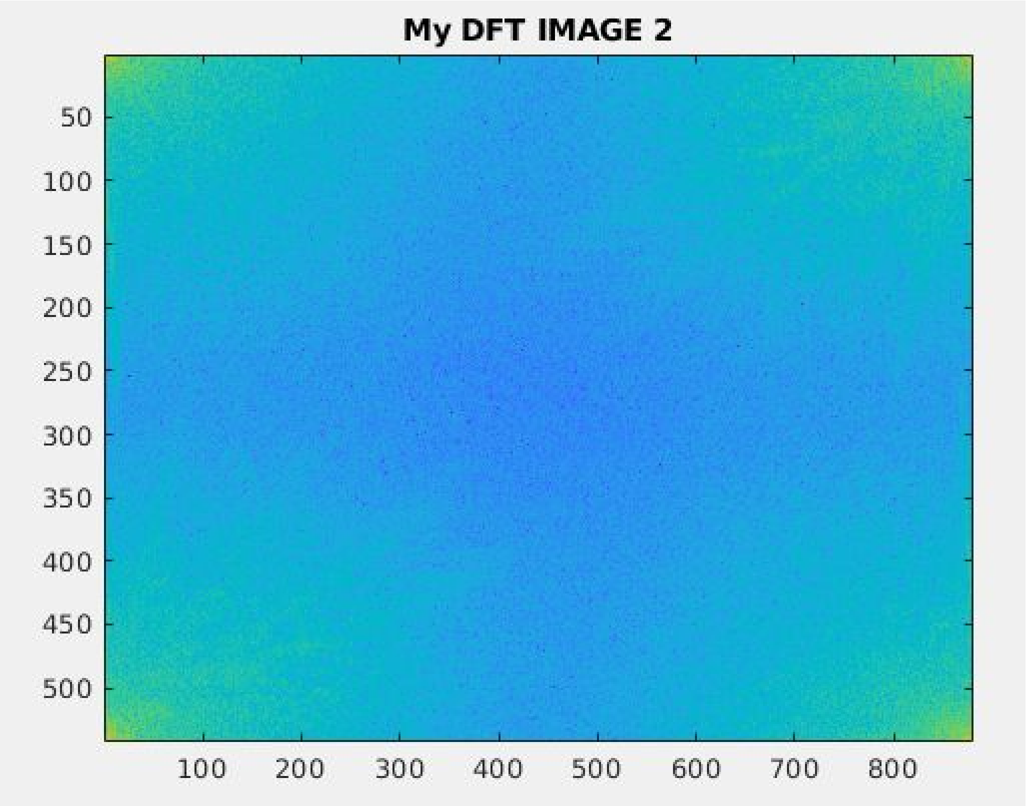
Output of inbuilt Fft

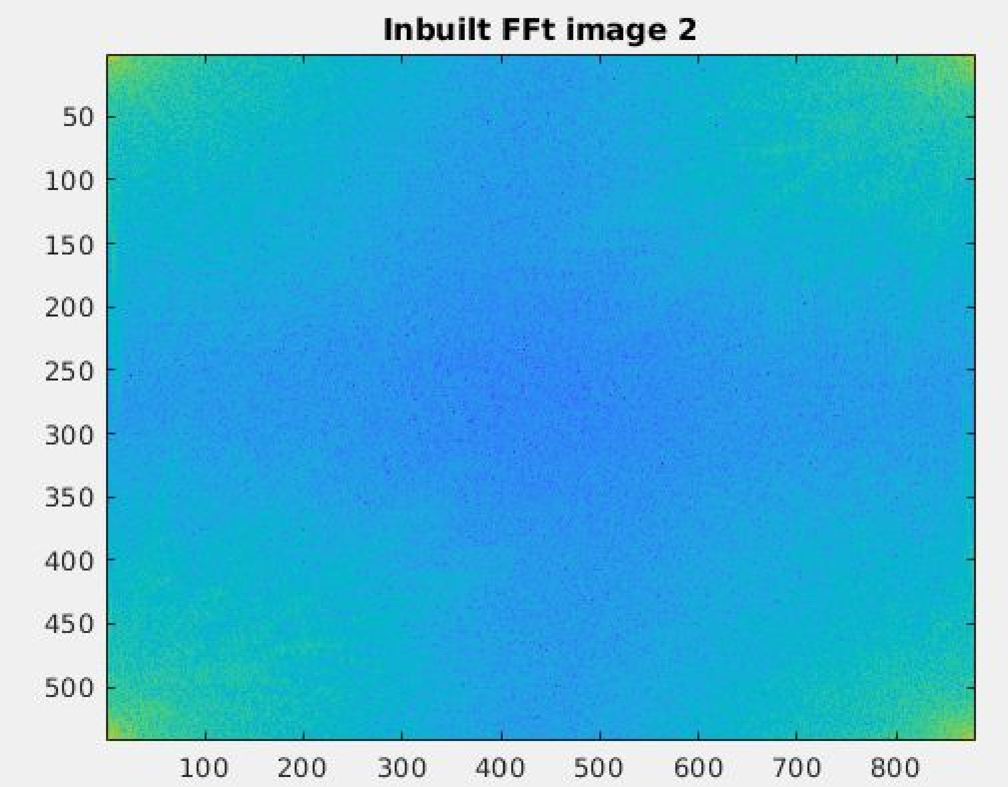


Output of Recursive FFT (Implemented).

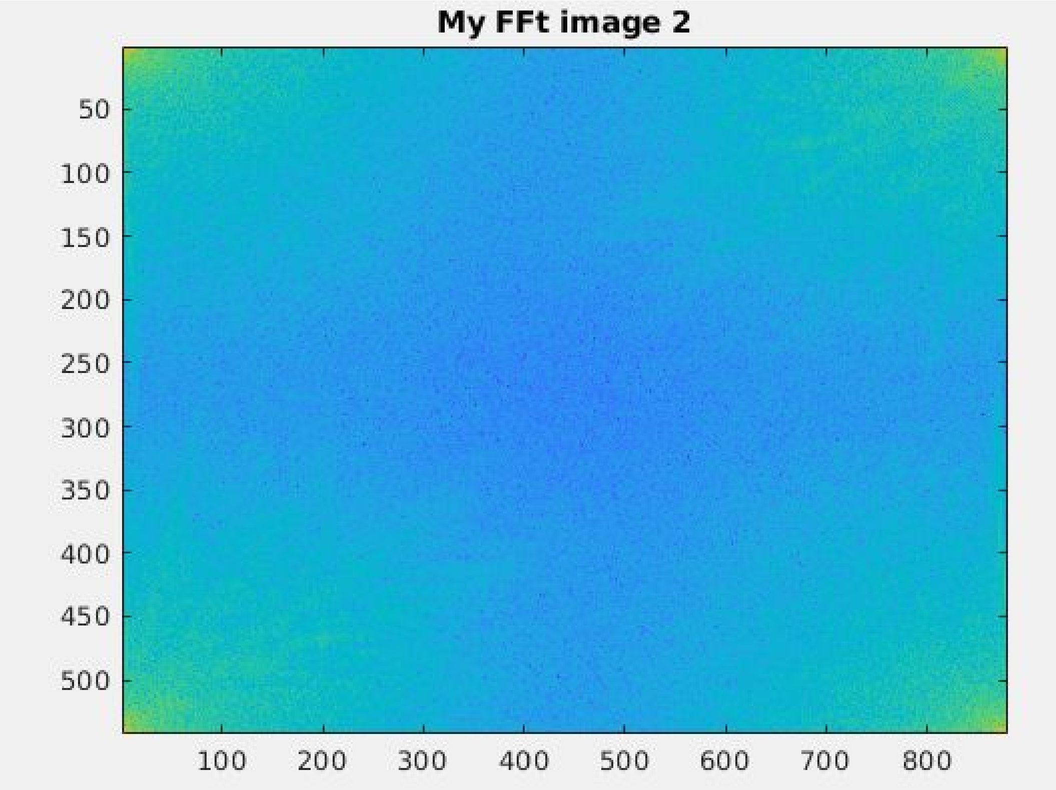


1024 *1024 image





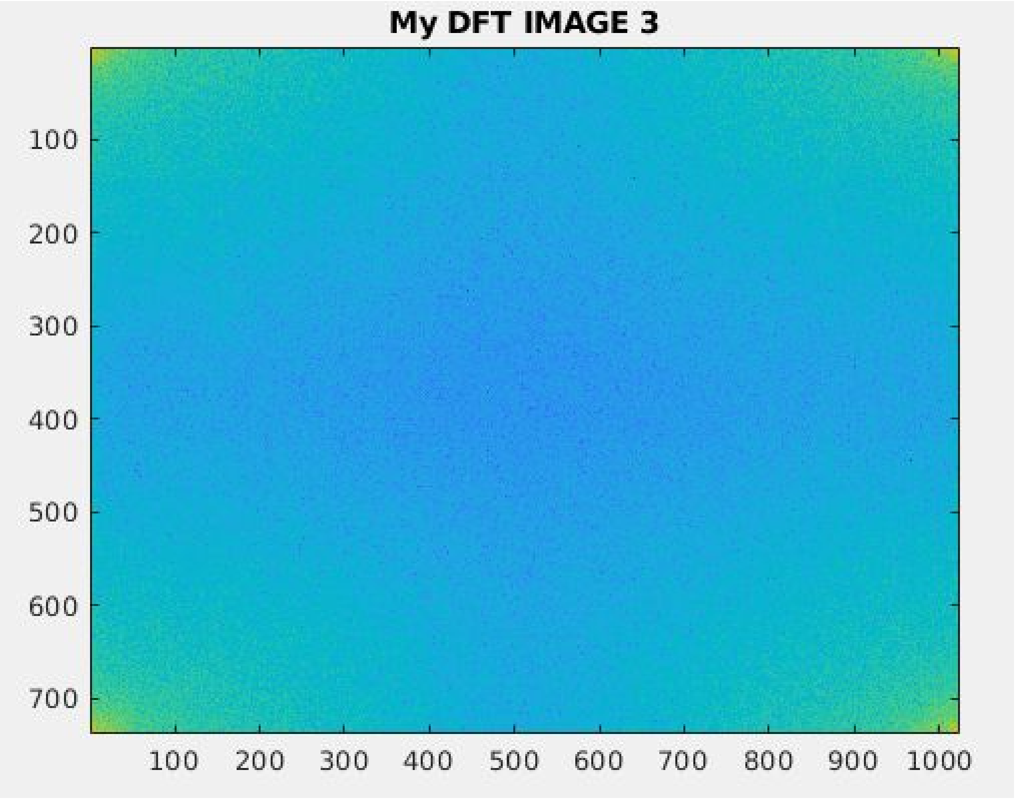
Inbuilt FFt image 2

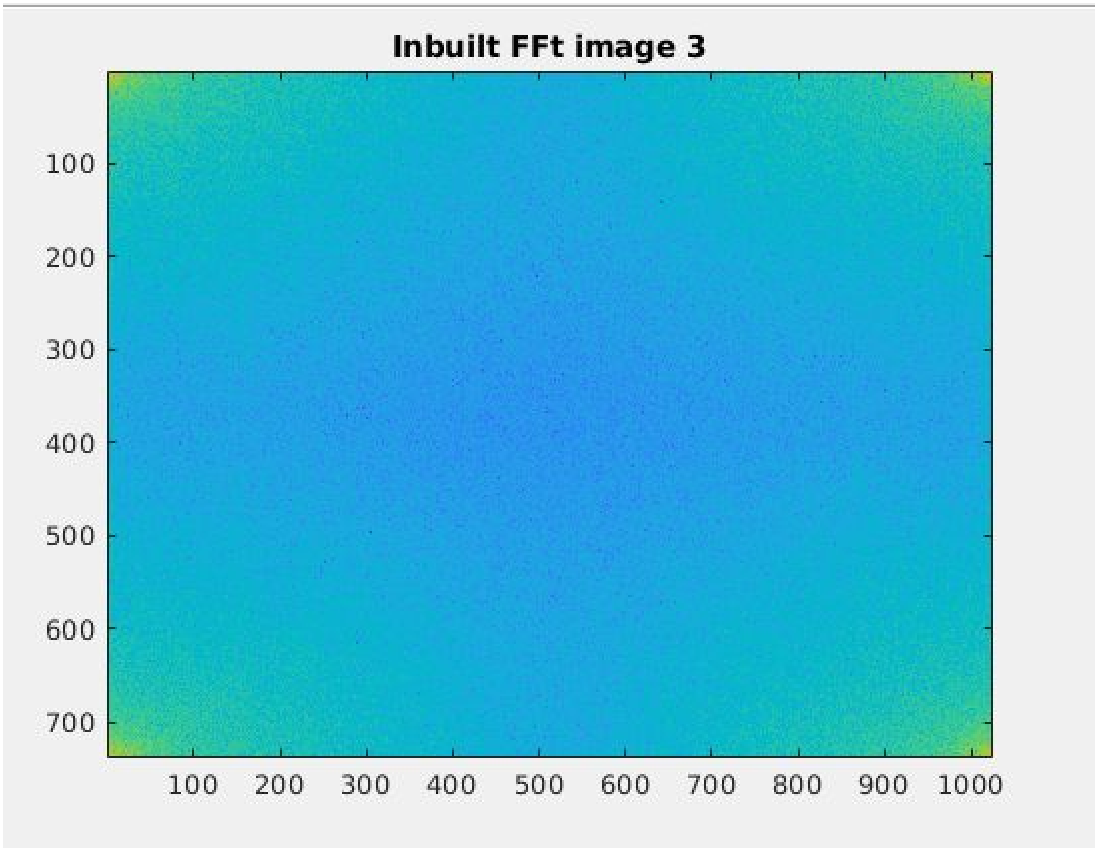


My FFt image 2

org
gettyimages
Mark Mawson
468971391

738 * 1024


My DFT IMAGE 3
100
200
300
400
500
600
700
100
200
300
400
500
600
700
800
900
1000



TIME COMPARISON

1. **IMPLEMENTED DFT VS INBUILT FFT**

A. size = 256*256

DFT - Elapsed time is 0.131528 seconds.
I_FFT - Elapsed time is 0.076505 seconds.
REC_FFT - Elapsed time is 0.119624 seconds.

Time requirements are approximately the same for inbuilt fft and recursive fft, but dft takes more time.

B. size = 1024*1024

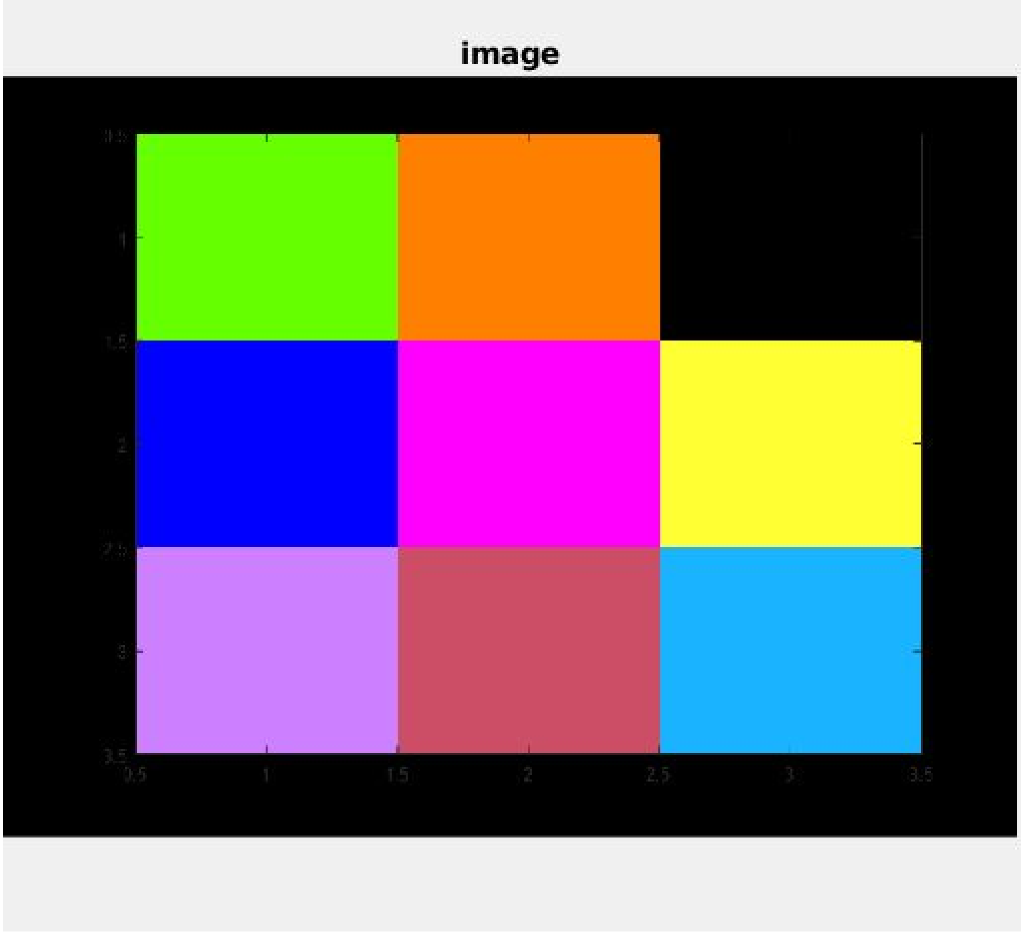
DFT - Elapsed time is 0.878335 **seconds.**
I_FFT - Elapsed time is 0.160339 **seconds.**
REC_FFT - Elapsed time is 0.472599 **seconds.**

C. size = 738*1024

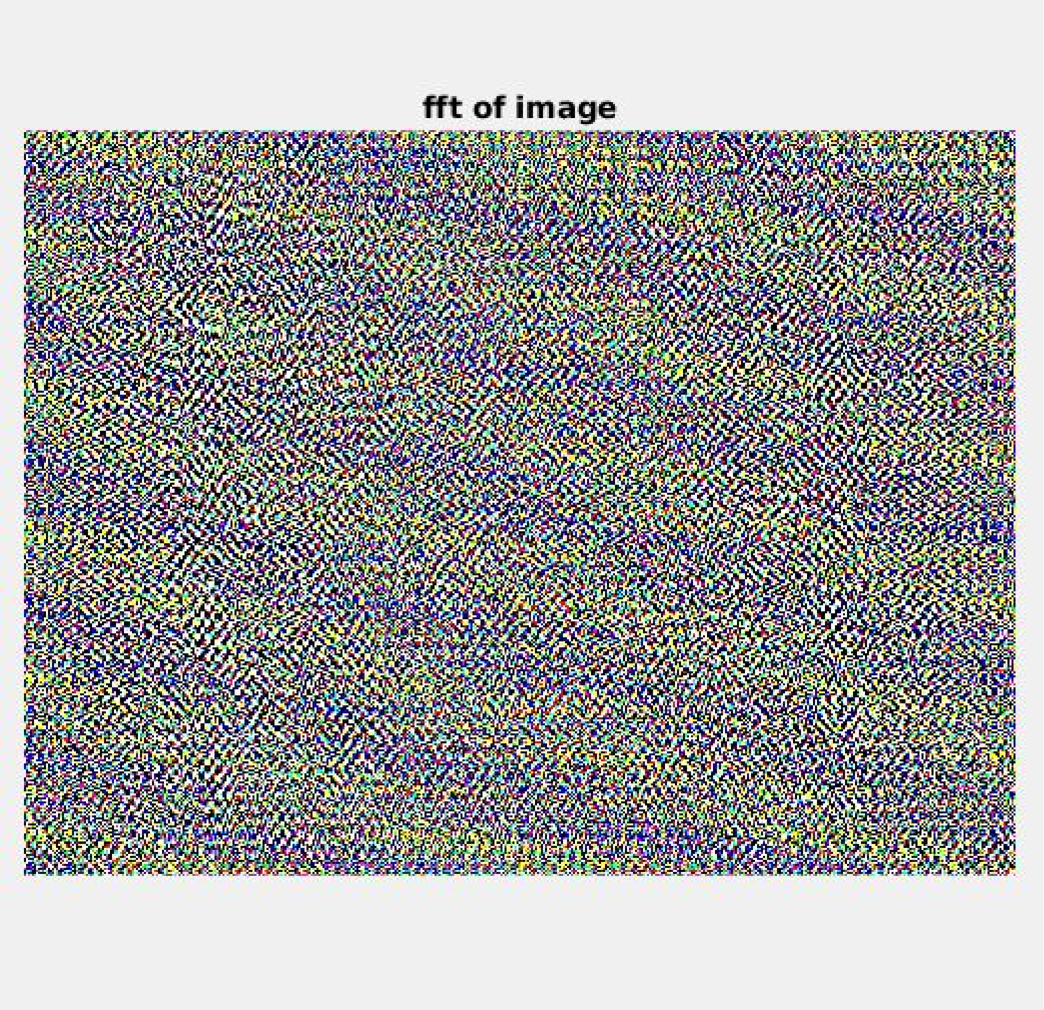
DFT - Elapsed time is 0.563272 seconds.
I_FFT - Elapsed time is 0.100844 seconds.
REC_FFT - Matrix dimensions do not agree.

---xxxxxxxxxxxxxx--

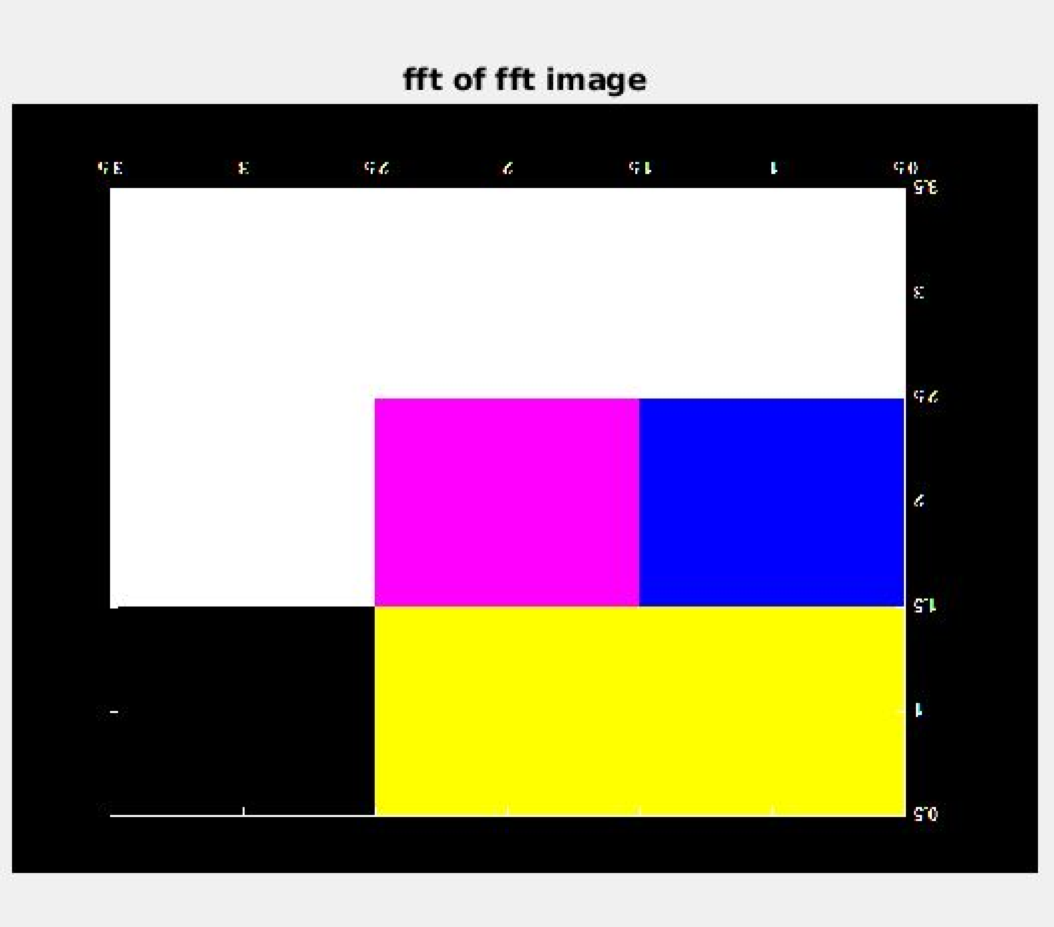
QUESTION 6



Original Image

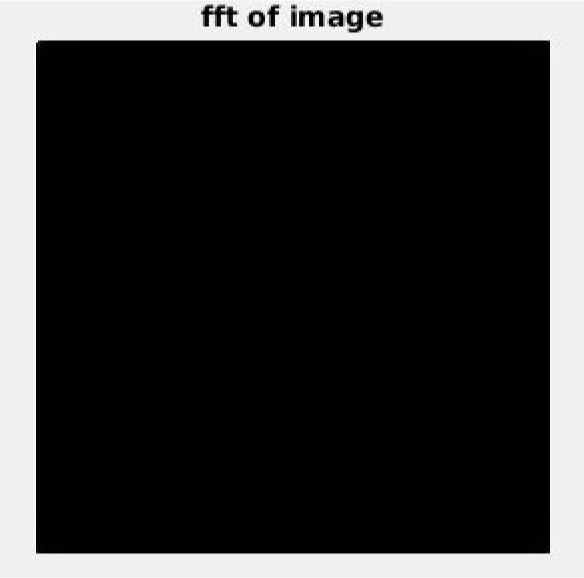


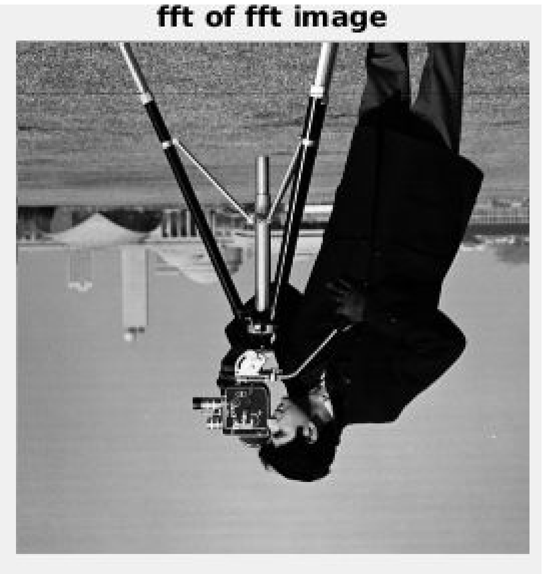
Fft Of the Original Image



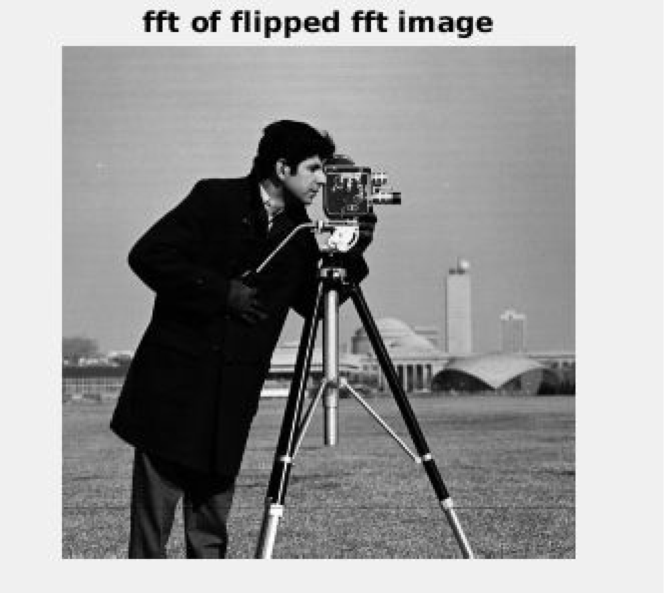
Fft of fft of image

The most common observation is that the fft of fft is double flipped .





In the frequency domain (i.e after taking the first fft, we manually flip the image wrt to both x and y direction since we know (by observation)) that the final image is flipped in both x and y.

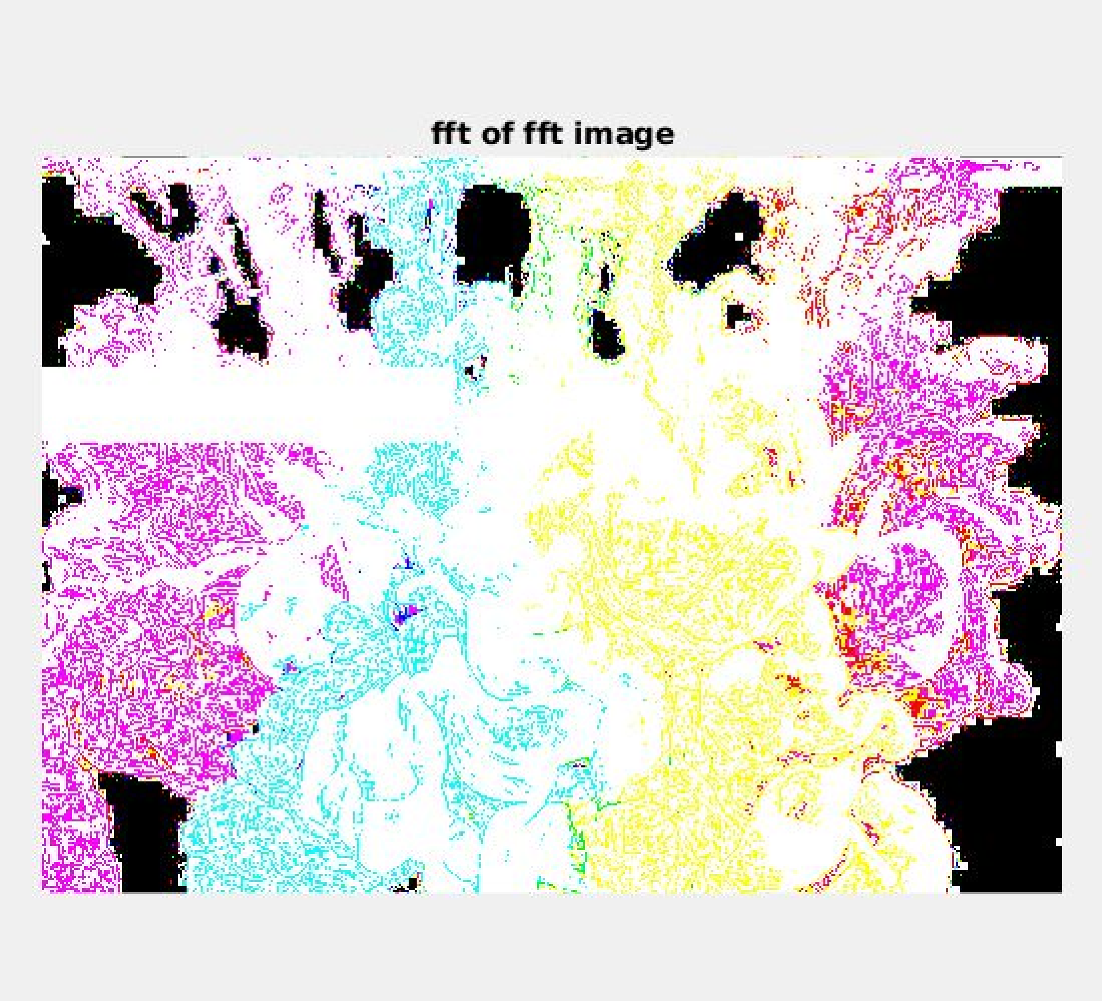


fft of flipped fft image

This effectively makes the image of the same orientation.

image
gettyimages
Mark Mawson

fft of image

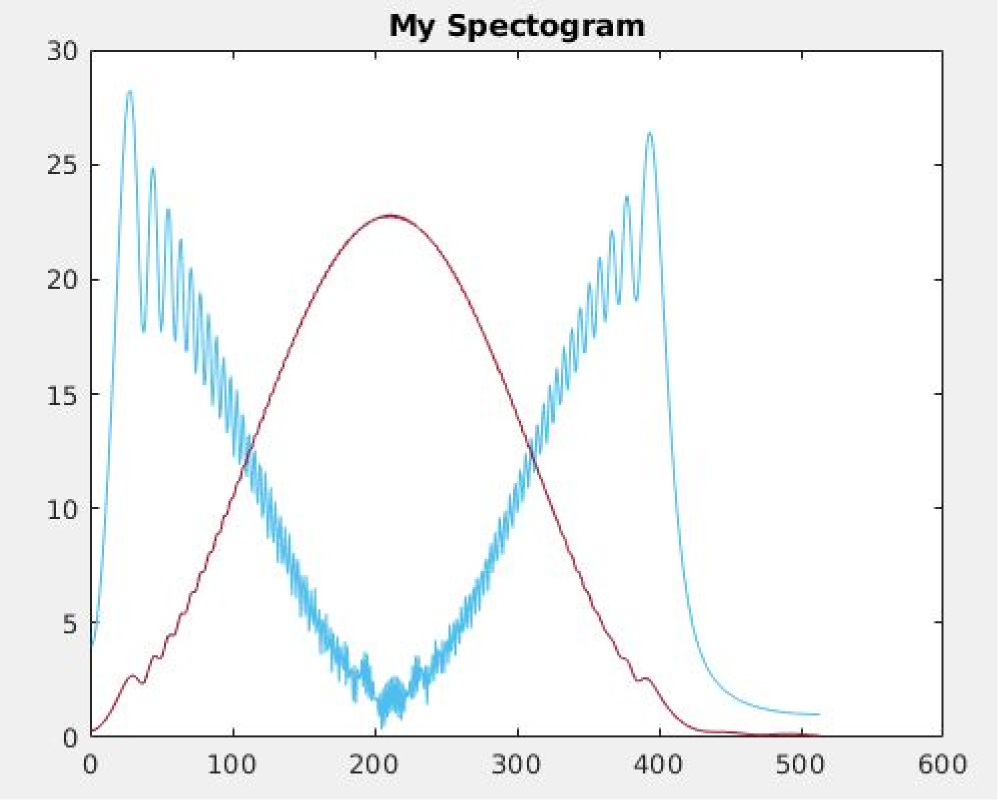


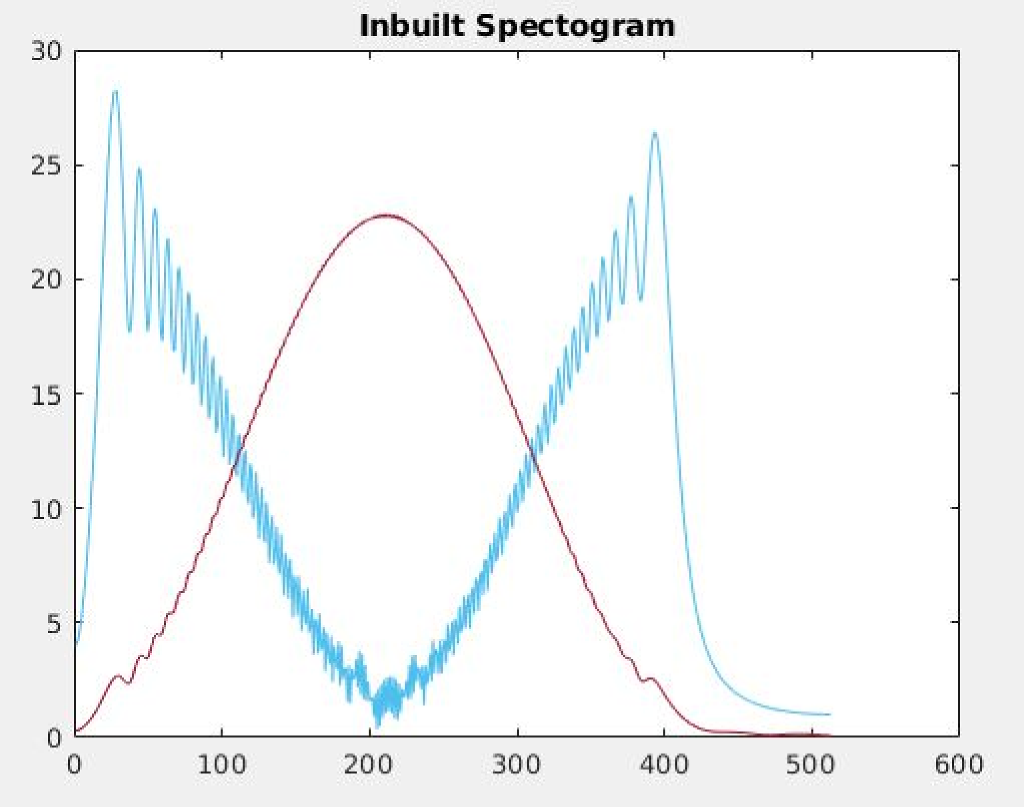
Q.What does fft of an image denote ?
A.It denotes the amplitudes and frequencies of the sines/cosines that, when added up, will give you the original image.

--xxxxxxxxxxxxxx---

QUESTION 7

PART 1

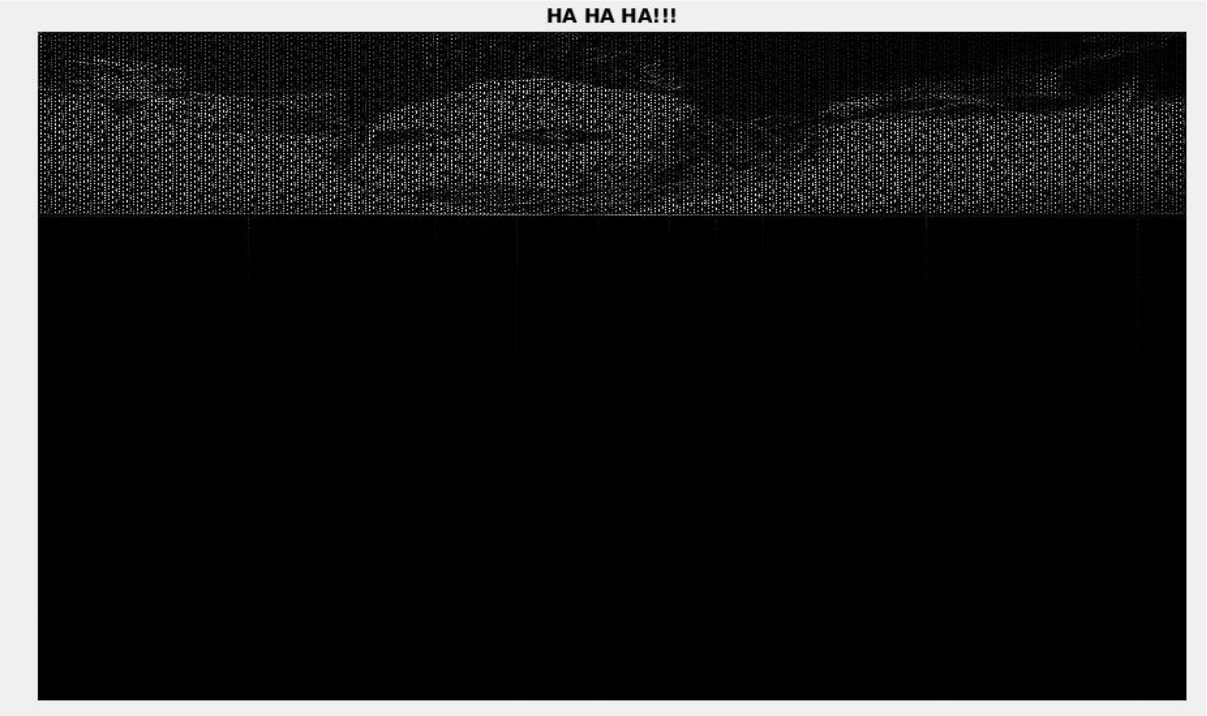




Hamming -> rect function.

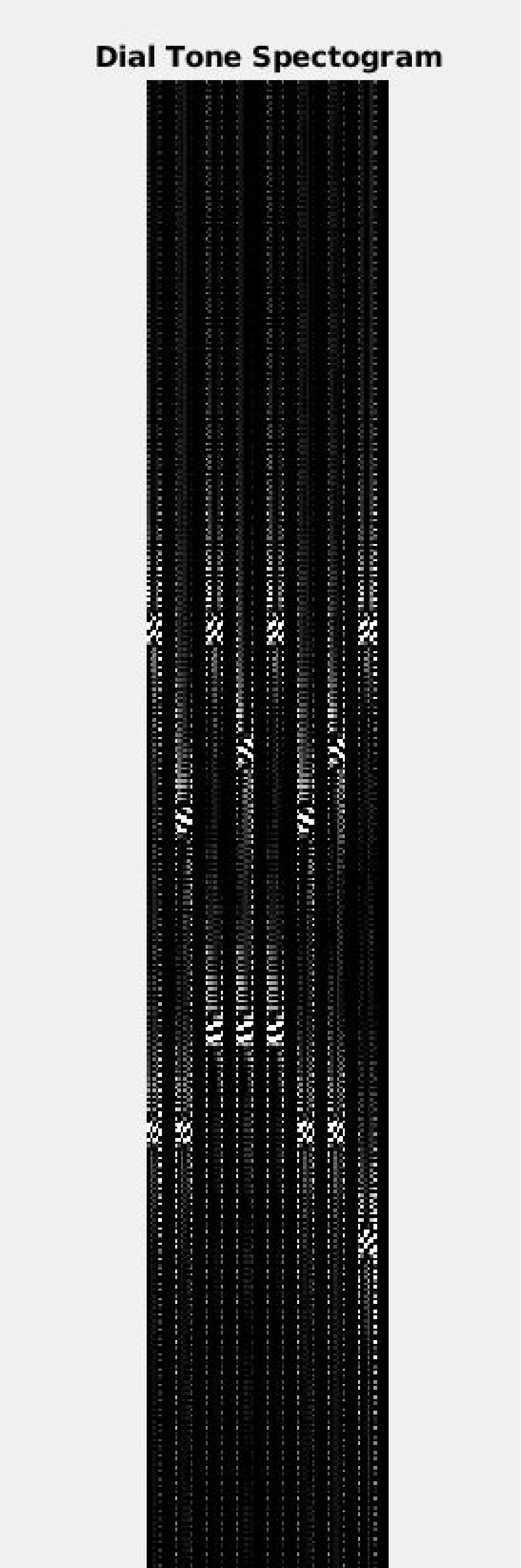
Takes window size,stride and number of points at which fft needs to be calculated.

PART 2



Just drew the spectrogram and found the key is 'JOKER'

PART 3



Dial Tone Spectogram

We can see clearly the eight tones.

---xxxxxxxxxxxxxx---