EE386 Lab 4 - FIR Filtering and Image Processing

Dec. 17, 2018

Lab directed by Chris Hirunthanakorn

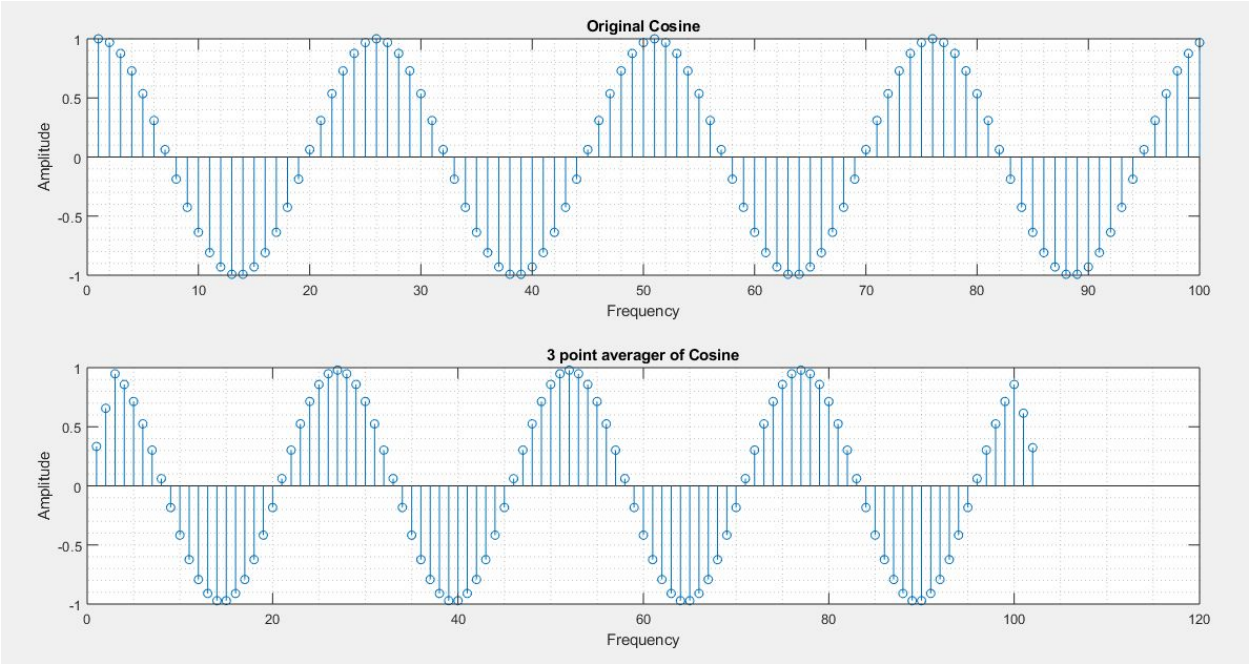
Lab conducted by Paco Ellaga

CSULB SID: 009602331

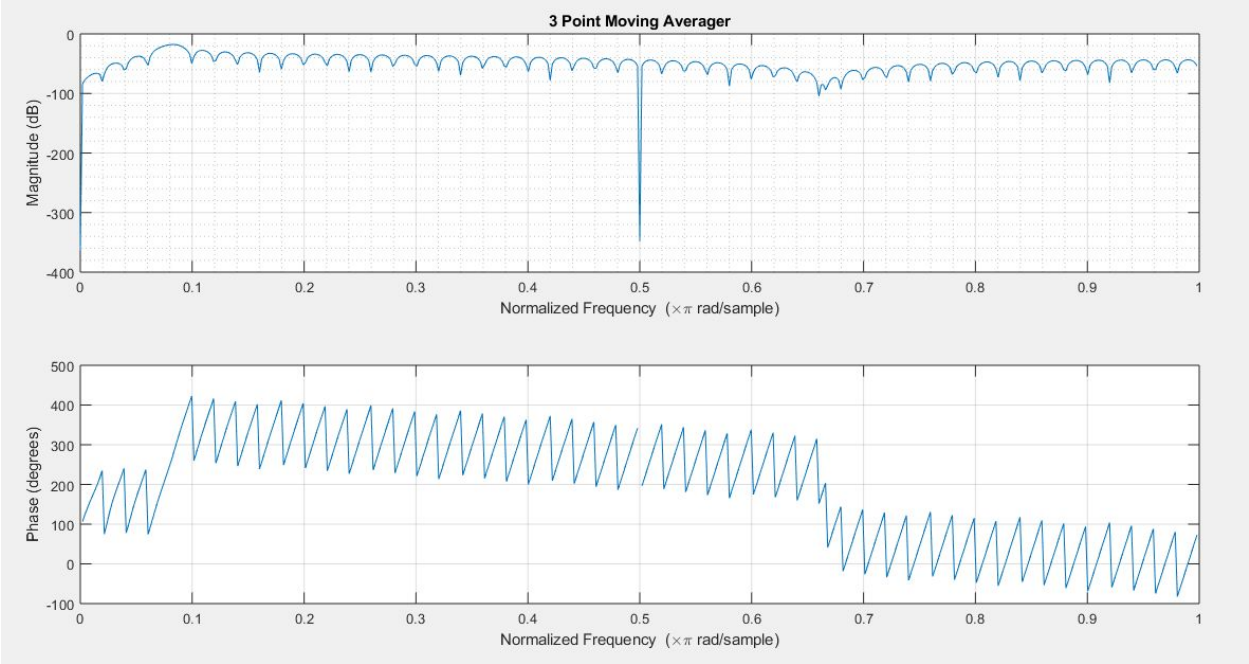
Experimenting with Moving Averagers

3-Point Averager

```
nn = linspace(0,99);        % 100 point vector from 0 to 99
x1 = cos(0.08*pi*nn);       % original signal
a3 = (1/3)*ones(3,1);       % points for 3 point averager
y1 = firfilt(a3,x1);        % filter for 3 point averager input
figure(1)
subplot(2,1,1)
stem(x1)
xlabel('Frequency'), ylabel('Amplitude')
title('Original Cosine'), grid minor
subplot(2,1,2)
stem(y1)
xlabel('Frequency'), ylabel('Amplitude')
title('3 point averager of Cosine'), grid minor
```



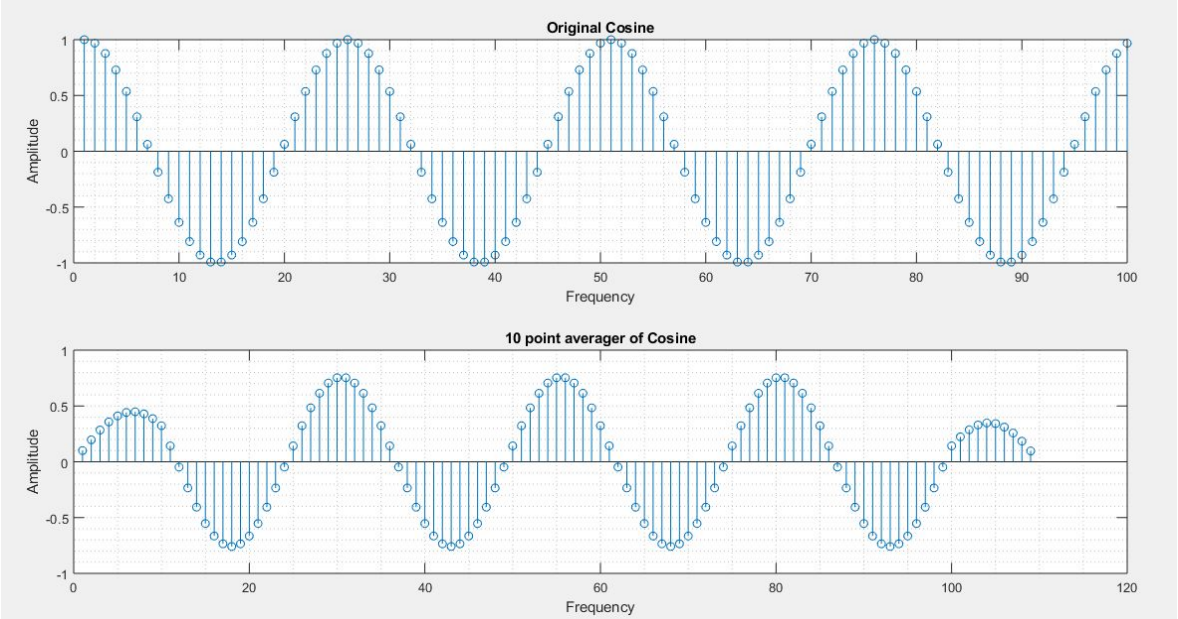
```
figure(2)
freqz(y1,nn)
title('3 Point Moving Averager'), grid minor
```



10-Point Averager

```
% repeating for 10 point averager
a10 = (1/10)*ones(10,1);    % points for 10 point averager
y2 = firfilt(a10,x1);       % filter for 10 point averager
figure(3)
subplot(2,1,1)
stem(x1)
xlabel('Frequency'), ylabel('Amplitude')
title('Original Cosine'), grid minor
subplot(2,1,2)
stem(y2)
xlabel('Frequency'), ylabel('Amplitude')
```

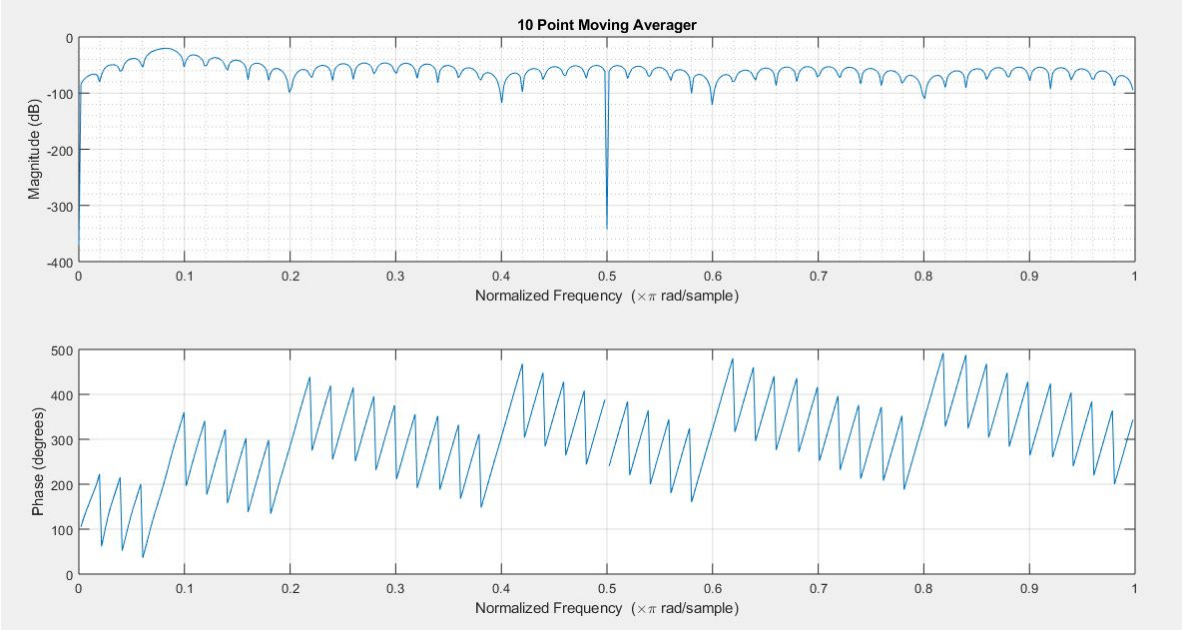
title('10 point averager of Cosine'), grid minor



figure(4)

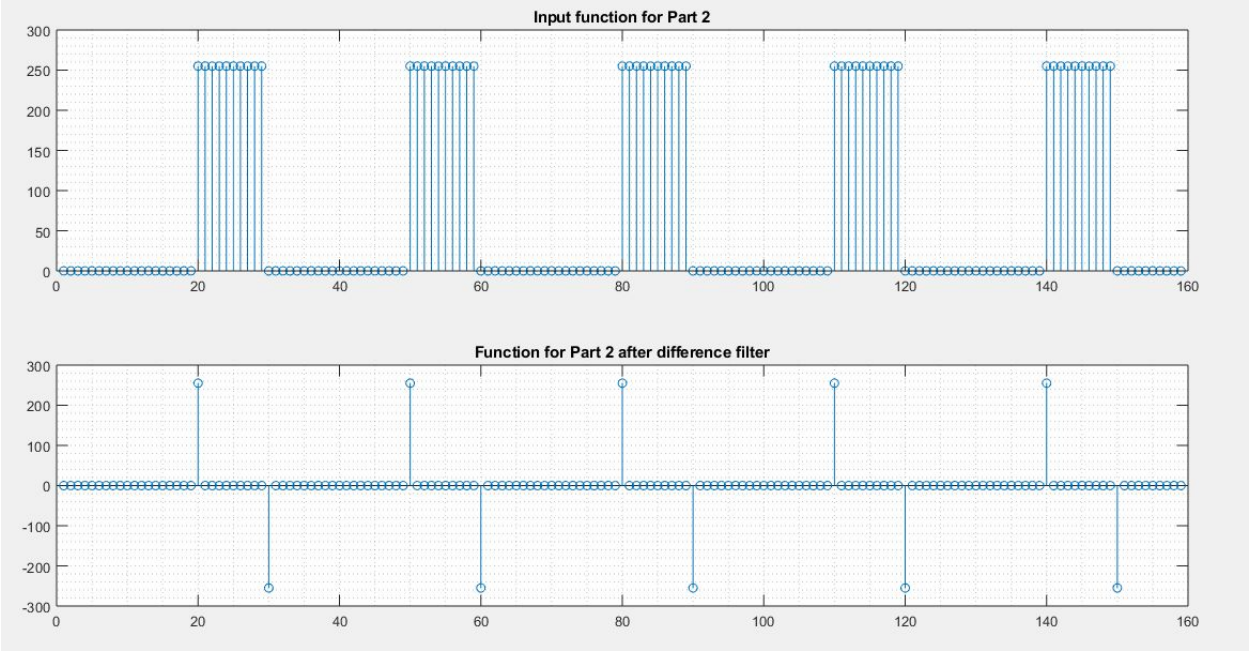
freqz(y2,nn)

title('10 Point Moving Averager'), grid minor



Difference Filter

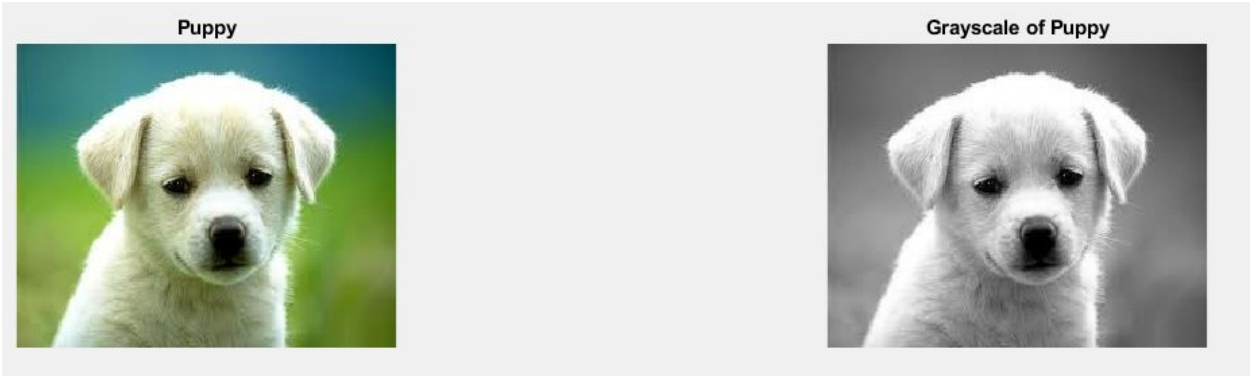
```
x2 = 255*(rem(1:159,30)>19);    % input function
y2 = filter([1,-1],1,x2);       % difference filter
figure(5)
subplot(2,1,1)
stem(x2), grid minor
title('Input function for Part 2')
subplot(2,1,2)
stem(y2), grid minor
title('Function for Part 2 after difference filter')
```

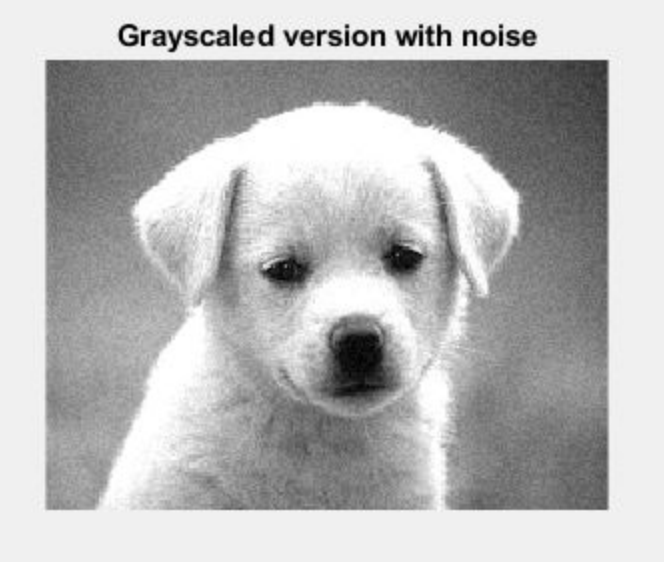



```
LIn = length(x2), LOut = length(y2)
Lin = 159
LOut = 159
```

2D Filtering on Images

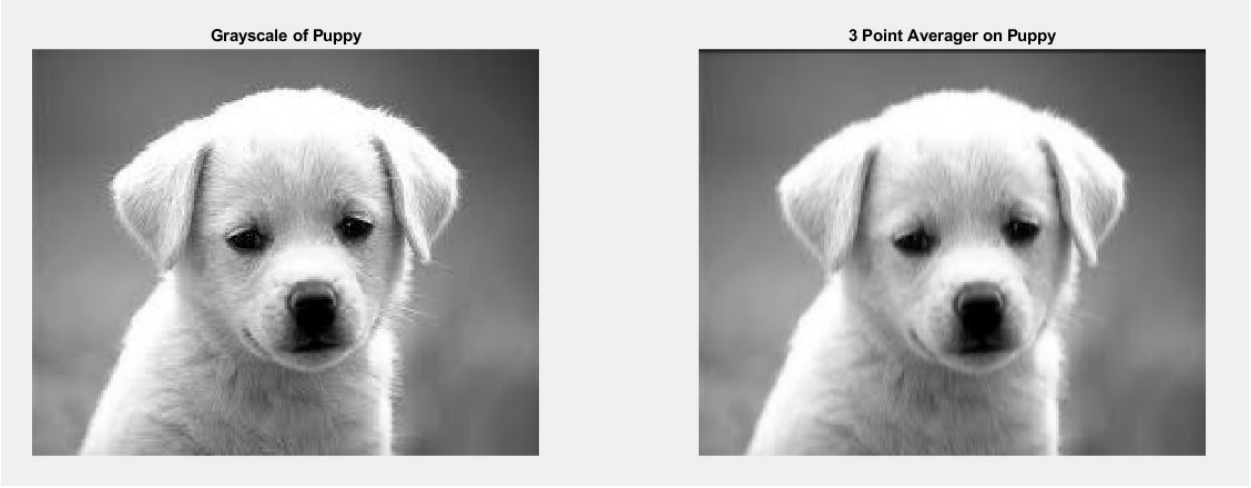
```
figure(6)
subplot(2,2,1)
Pup = imshow(Puppy);            % display of original image
title('Puppy')
GrayPup = rgb2gray(Puppy);
subplot(2,2,2)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
GrayPup = im2double(GrayPup);   % image converted to double
% calculation of the dynamic range
DynamicRange = max(max(GrayPup)) - min(min(GrayPup))
[rows,cols] = size(GrayPup);
noise = (DynamicRange/10)*rand(rows,cols);  % applying noise to range
NoisePup = GrayPup + noise;                 % applying noise to GrayPup
subplot(2,2,[3,4])
imshow(NoisePup);                   % display of grayscaled with noise
title('Grayscaled version with noise')
size(NoisePup)                      % checking size of picture
```

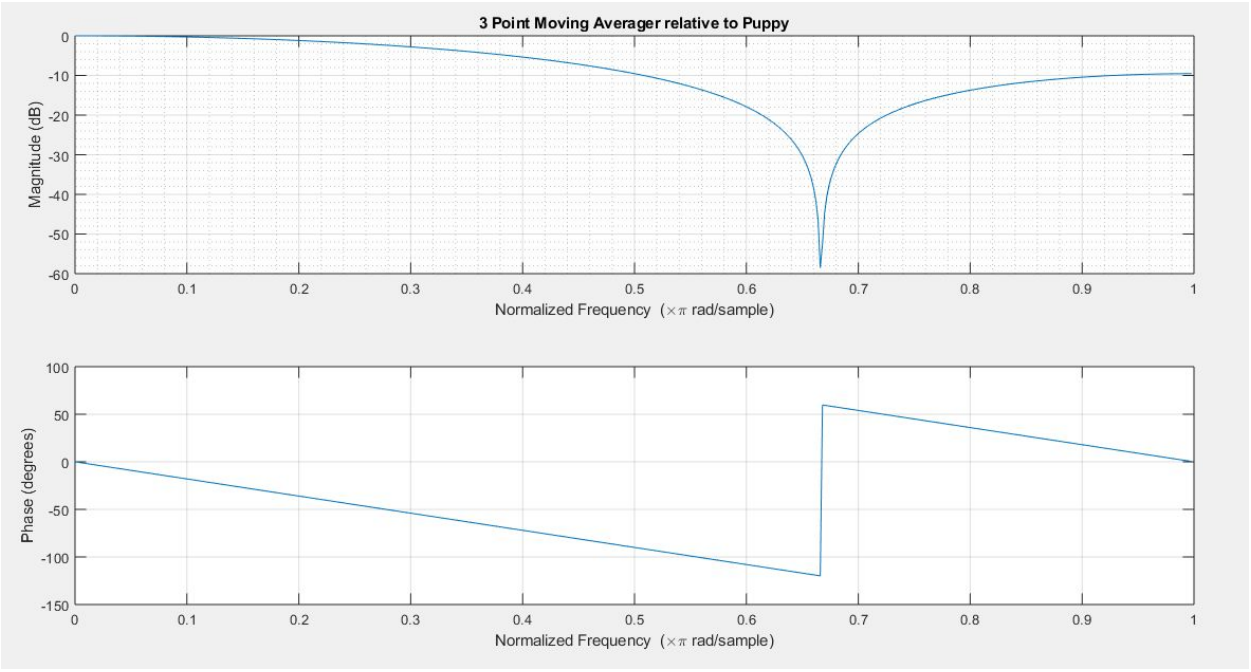




Part 3 - Question 1: 3 Point Averager on Puppy

```
b0 = (1/3)*ones(3,1);
Averager3Pup = filter(b0,1,GrayPup);
figure(7)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(Averager3Pup);           % display of 3-point version
title('3 Point Averager on Puppy')
B0 = [1/3 1/3 1/3];
figure(8)
freqz(B0);                      % frequency response
title('3 Point Moving Averager relative to Puppy'), grid minor
rmse = sqrt(immse(Averager3Pup,GrayPup))
size(Averager3Pup)              % checking size of picture
```





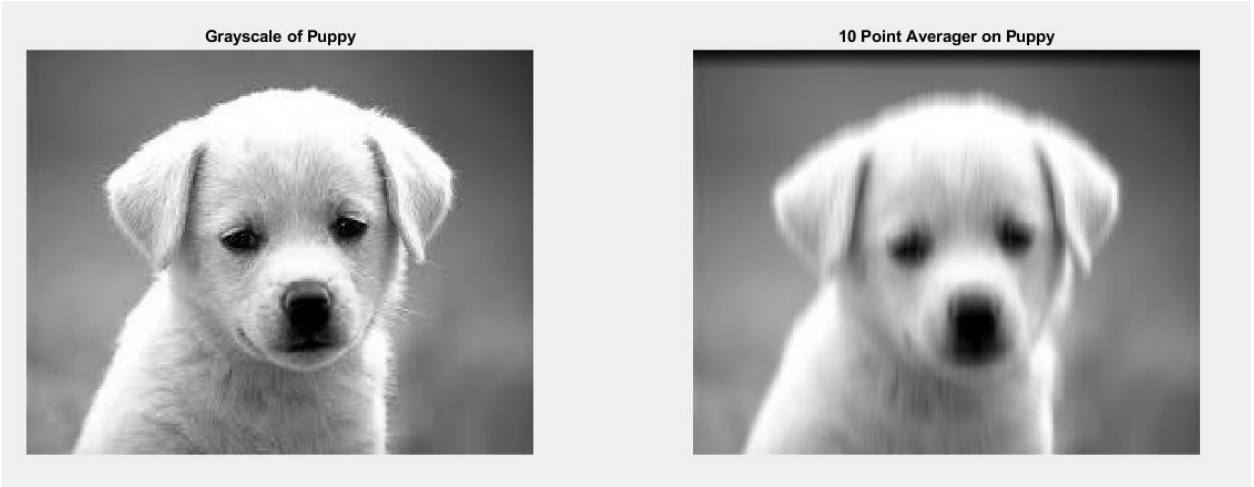
QUESTIONS:

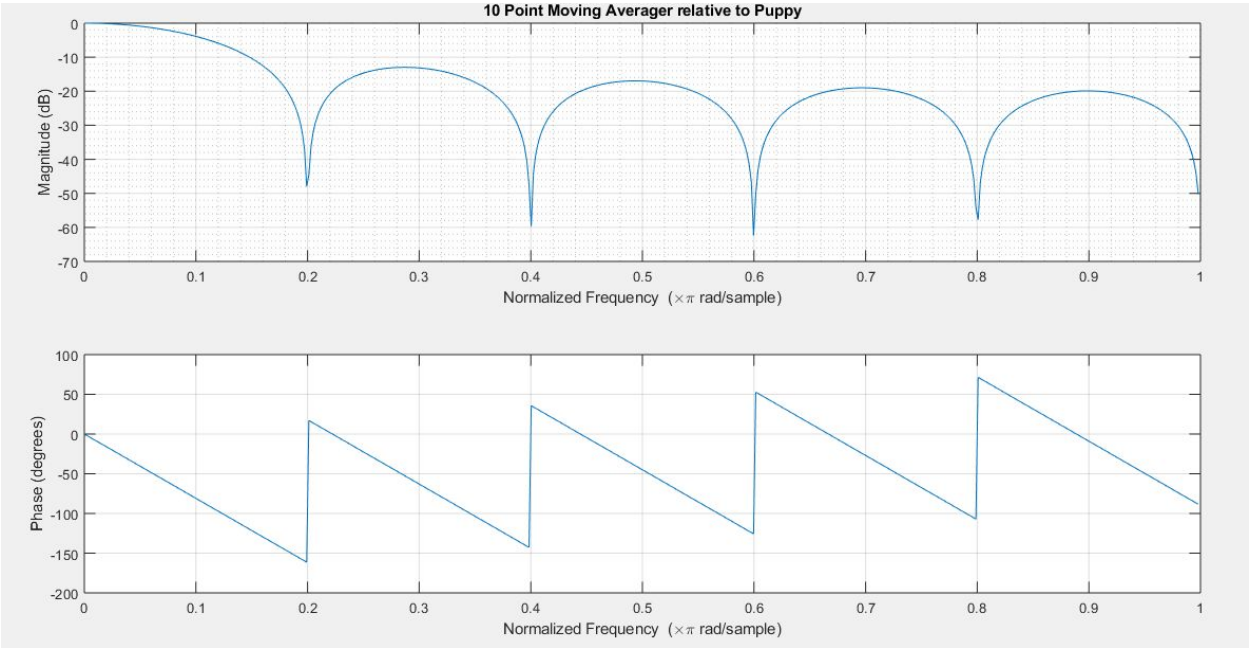
1. The 3 Point Averager makes it slightly blurry.
2. The size is 200x250 double
3. The filter is a low pass because it dips from zero to a lower frequency, pinches and remains dipped
4. The transient portion is the blurriness of the puppy, while the background remains mostly intact
5. The 3 Point averager passes frequencies below roughly 0.65.

6. The RMS error calculates out to 0.0350

Part 3 Question 2 - 10 Point Averager on Puppy

```
b1 = (1/10)*ones(10,1);
Averager10Pup = filter(b1,1,GrayPup);
figure(9)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(Averager10Pup);          % display of grayscaled version
title('10 Point Averager on Puppy')
B1 = (1/10)*ones(1,10);
figure(10)
freqz(B1);                      % frequency response
title('10 Point Moving Averager relative to Puppy'), grid minor
rmse = sqrt(immse(Averager10Pup,GrayPup))
size(Averager10Pup)             % checking size of picture
```





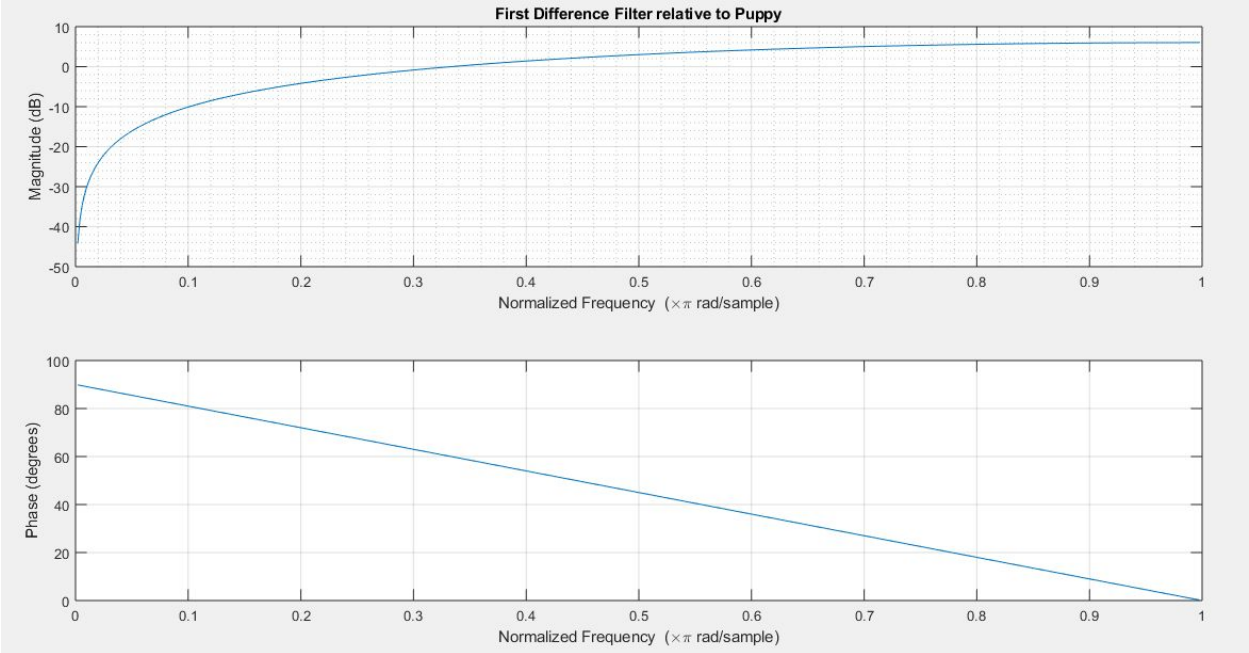
QUESTIONS:

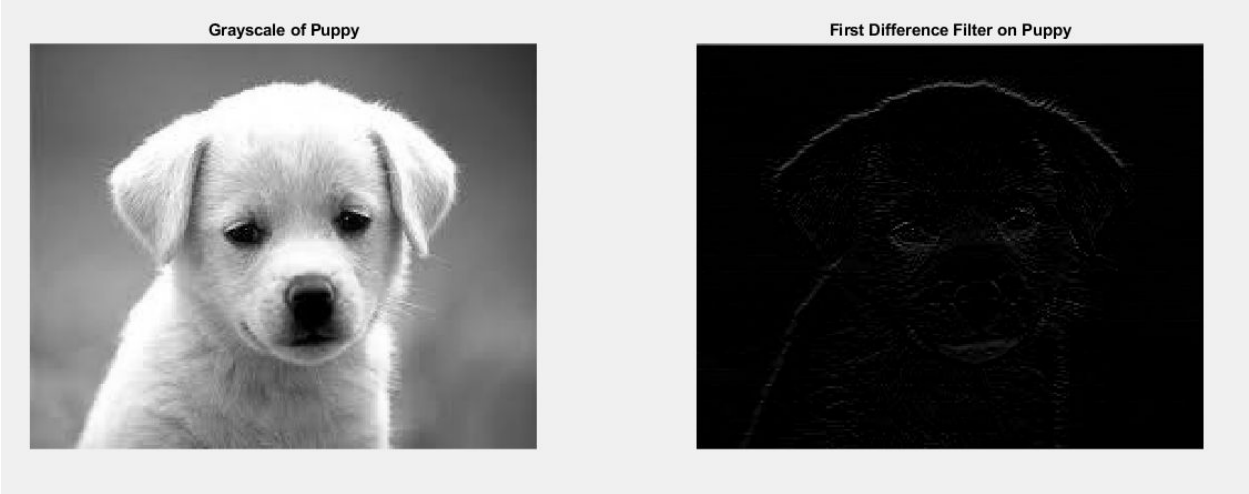
1. The 10 point averager makes it more blurry than the 3 point averager
2. The size is still 200x250 double
3. Since it's still an averager, it's still going to be a low pass
4. The transient part is the entire puppy where it's completely blurred out
5. The 10 point averager passes frequencies that are below roughly 0.2
6. The RMS error calculates out to 0.0895

Part 3 Question 3 - First Difference Filter on Puppy

```
bk0 = [1 -1];       % vector of coefficients
figure(11)
freqz(bk0,1);       % frequency response
title('First Difference Filter relative to Puppy'), grid minor
DiffPup = filter(bk0,1,GrayPup);
figure(12)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
```

```
subplot(1,2,2)
imshow(DiffPup);                % display of first difference filter
title('First Difference Filter on Puppy')
rmse = sqrt(immse(DiffPup,GrayPup))
```





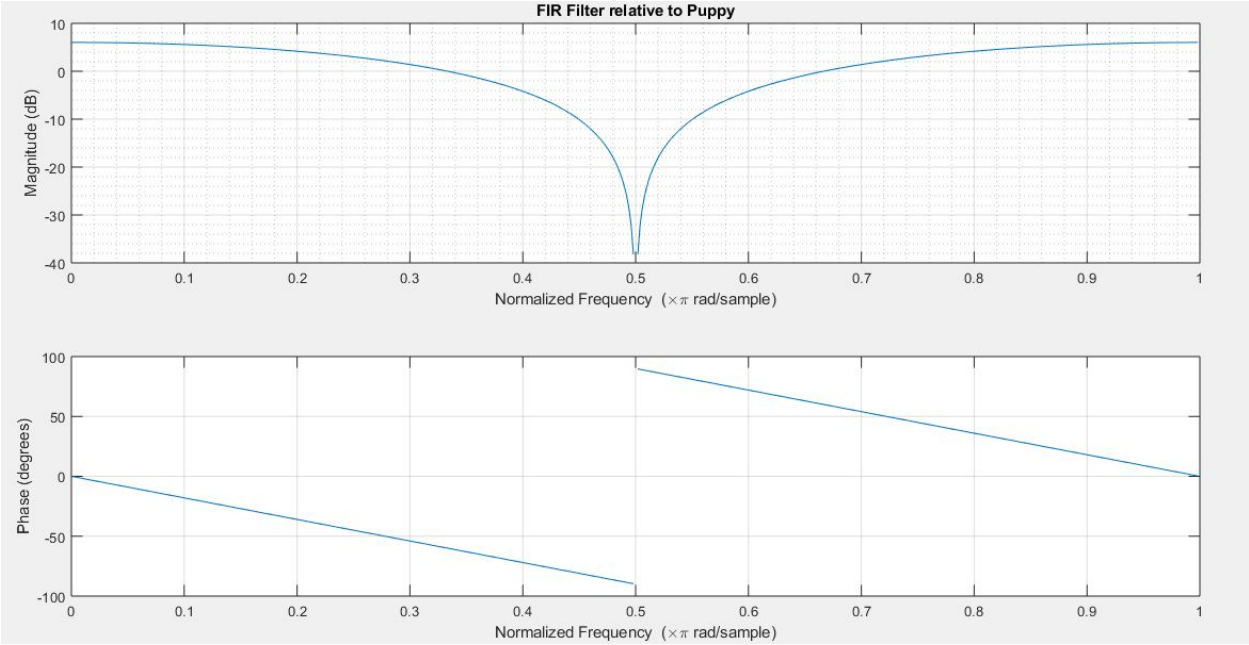
QUESTIONS:

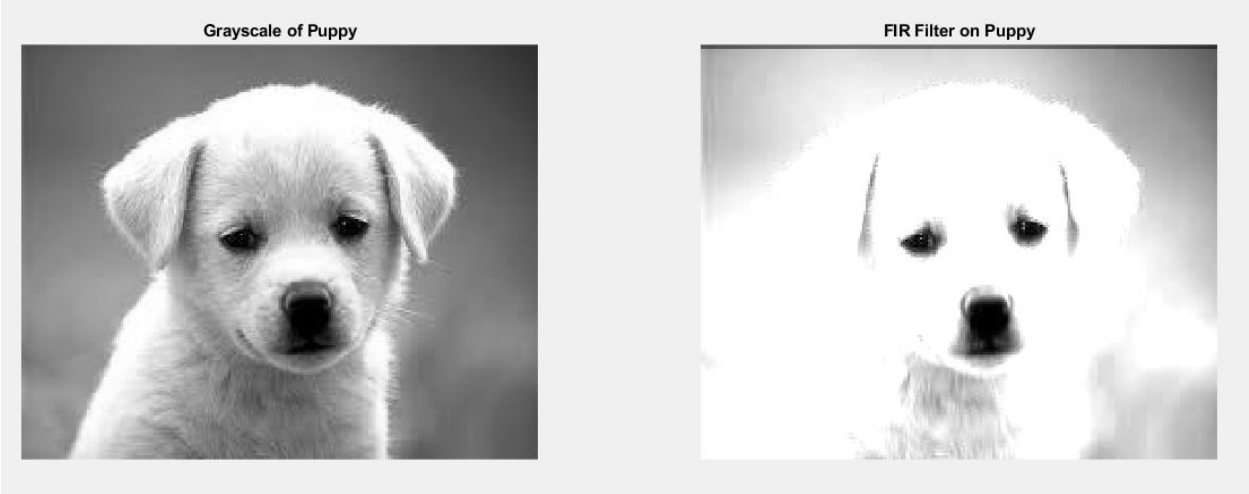
1. The contrast went all the way down to black for everything that was gray or white and captured everything that were edges.
2. The size is 200x250 double

3. The filter is a highpass filter allowing only the frequencies of the edges through
4. The transient part of the image is the edges of the image being highlight through the blackened portions.
5. The frequency flattens and caps around 6dB from ~-45dB.
6. 0.6204

Part 3 Question 4 - FIR Filter on Puppy

```
bk1 = [1,0,1];      % vector of coefficients
figure(13)
freqz(bk1,1);       % frequency response
title('FIR Filter relative to Puppy'), grid minor
FIRPup = filter(bk1,1,GrayPup);
figure(14)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(FIRPup);                 % display of first difference filter
title('FIR Filter on Puppy')
rmse = sqrt(immse(FIRPup,GrayPup))
size(FIRPup)                % checking size of picture
```



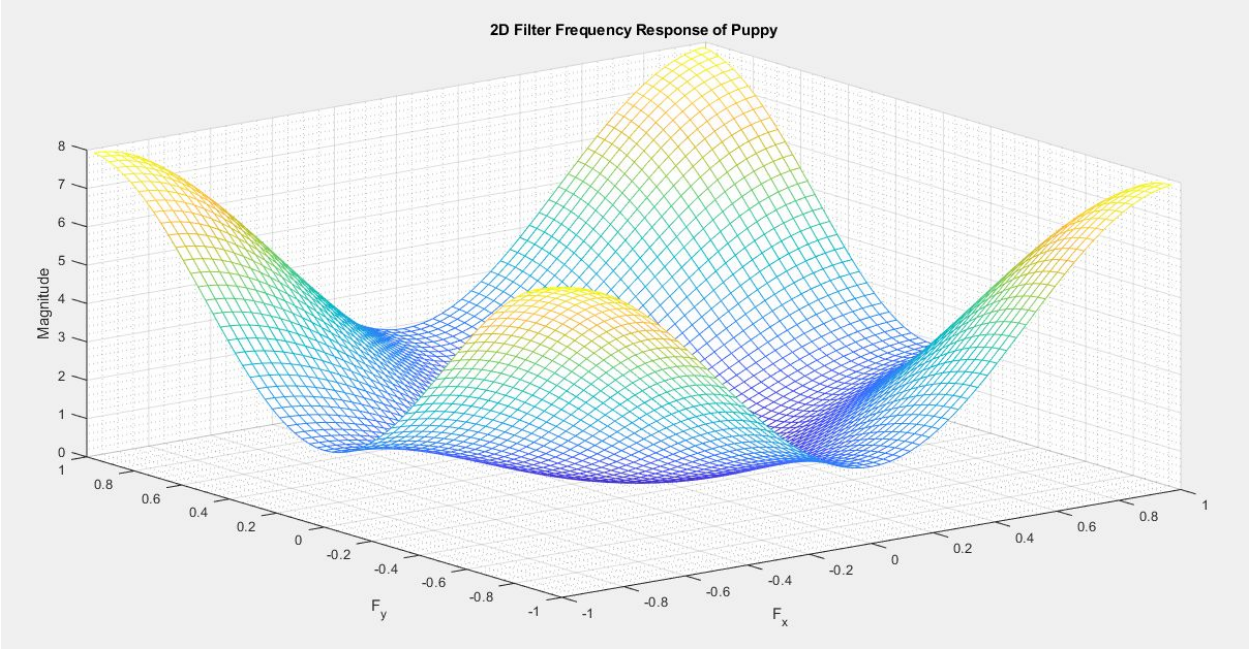


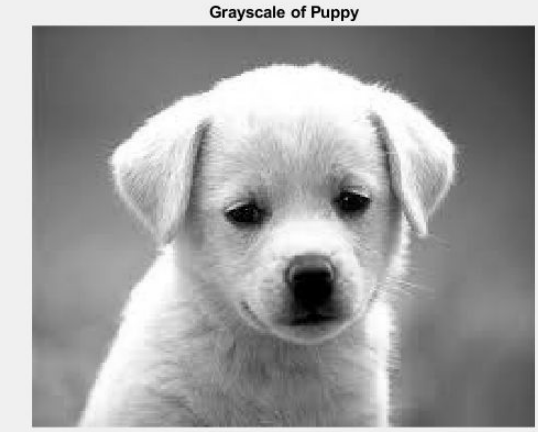
QUESTIONS:

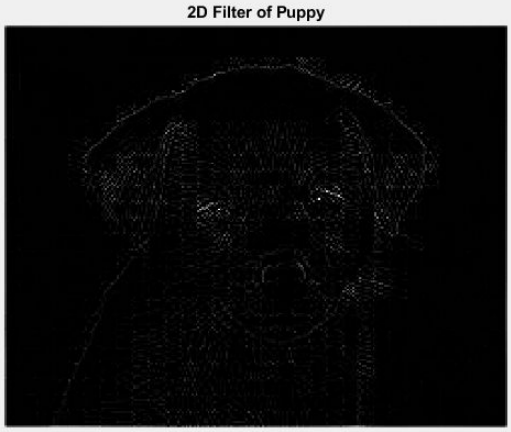
1. The opposite happened with the FIR filter than the First Difference Filter. It increased the contrast up to bright white and ignored the edges.
2. The size is 200x250 double
3. The filter is classified as a band stop. It is where the high and low are held, but certain one are removed altogether.
4. The transient portions are the grayed portion that are now whitened
5. The notch frequencies are displayed to be at 0.5
6. The RMS error is 0.6193.

Part 3 Questions 5 - 2D Filter on Puppy

```
bb = [0.25,-1,0.25;-1,3,-1;0.25,-1,0.25];   % vector of coefficients
figure(15)
freqz2(bb);                                 % frequency response
title('2D Filter Frequency Response of Puppy'), grid minor
TwoDPup = filter2(bb,GrayPup, 'same');
figure(16)
subplot(1,2,1)
imshow(GrayPup);                            % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(TwoDPup);                            % display of 2D filter on puppy
title('2D Filter of Puppy')
rmse = sqrt(immse(TwoDPup,GrayPup))
size(TwoDPup)                               % checking size of picture
```







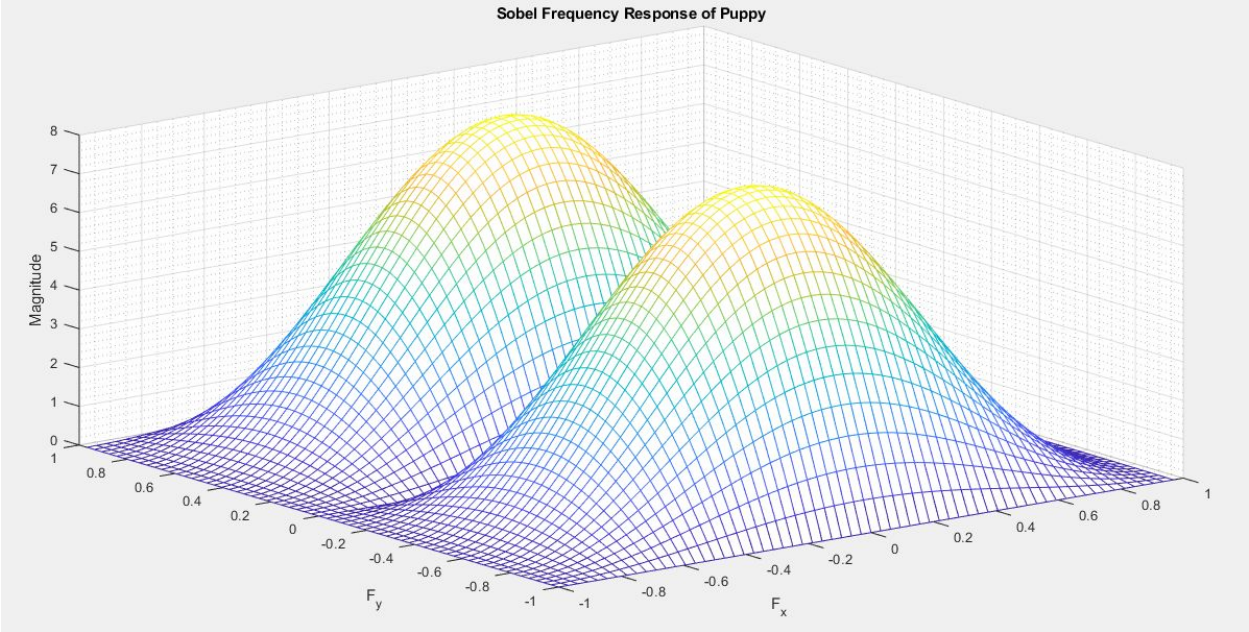
QUESTIONS:

1. The image darkened like the first difference filter and is only showing the edges
2. The size is 200x250 double
3. The filter is a high pass, edge detection filter.
4. The transient part is the edges being highlighted
5. The frequencies towards zero is nulled, but towards the negative and positive Fx and Fy are amplified.
6. The RMS error is 0.6191

Part 3 Question 6 - Sobel Filter on Puppy

```
s = [1,2,1;0,0,0;-1,-2,-1];              % vector of coefficients
figure(17)
freqz2(s);                               % frequency response
title('Sobel Frequency Response of Puppy'), grid minor
SobelPup = filter2(s,GrayPup, 'same');
figure(18)
subplot(1,2,1)
imshow(GrayPup);                         % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
```

```
imshow(SobelPup);                % display of sobel filter on puppy
title('Sobel Filter of Puppy')
rmse = sqrt(immse(SobelPup,GrayPup))
size(SobelPup)                   % checking size of picture
```



QUESTIONS:

1. The image is darkened and the the edges are highly outlined
2. The size is 200x250 double
3. The filter is of a band pass filter allowing for specific frequencies to pass through, but there are frequencies that are able to pass through.

4. The transient portion is the emphasised white edges in the darkened image
5. The filter is of a bandpass filter in one direction and acts a low pass across the other direction. This is separated by the two “hills”.
6. The RMS error is 0.6796

Appendix/ MATLAB CODE

```
cdflib.setFileBackward('BACKWARDFILEon');
mode = cdflib.getFileBackward;
% EE386 Project Lab 4
%
% MATLAB code conducted by Paco Ellaga (009602331)
%clear all; clc; close all;

%% Part 1 - Moving Averagers

nn = linspace(0,99);        % 100 point vector from 0 to 99
x1 = cos(0.08*pi*nn);       % original signal
a3 = (1/3)*ones(3,1);       % points for 3 point averager
y1 = firfilt(a3,x1);        % filter for 3 point averager input

figure(1)
subplot(2,1,1)
stem(x1)
xlabel('Frequency'), ylabel('Amplitude')
title('Original Cosine'), grid minor
subplot(2,1,2)
stem(y1)
xlabel('Frequency'), ylabel('Amplitude')
title('3 point averager of Cosine'), grid minor

figure(2)
freqz(y1,nn)
title('3 Point Moving Averager'), grid minor

% repeating for 10 point averager
a10 = (1/10)*ones(10,1);    % points for 10 point averager
y2 = firfilt(a10,x1);       % filter for 10 point averager

figure(3)
subplot(2,1,1)
```

```
stem(x1)
xlabel('Frequency'), ylabel('Amplitude')
title('Original Cosine'), grid minor
subplot(2,1,2)
stem(y2)
xlabel('Frequency'), ylabel('Amplitude')
title('10 point averager of Cosine'), grid minor

figure(4)
freqz(y2,nn)
title('10 Point Moving Averager'), grid minor

%% Part 2 - Difference Filters

x2 = 255*(rem(1:159,30)>19);    % input function
y2 = filter([1,-1],1,x2);       % difference filter

figure(5)
subplot(2,1,1)
stem(x2), grid minor
title('Input function for Part 2')

subplot(2,1,2)
stem(y2), grid minor
title('Function for Part 2 after difference filter')

LIn = length(x2), LOut = length(y2)

%% Part 3 a-d - 2D Filtering on Images
%
% uiopen('C:\Users\chron\Desktop\Puppy.png',1)

figure(6)
subplot(2,2,1)
Pup = imshow(Puppy);            % display of original image
title('Puppy')
```

```
GrayPup = rgb2gray(Puppy);
subplot(2,2,2)
imshow(GrayPup);              % display of grayscaled version
title('Grayscale of Puppy')

GrayPup = im2double(GrayPup);   % image converted to double

% calculation of the dynamic range
DynamicRange = max(max(GrayPup)) - min(min(GrayPup))

[rows,cols] = size(GrayPup);
noise = (DynamicRange/10)*rand(rows,cols);  % applying noise to range
NoisePup = GrayPup + noise;               % applying noise to GrayPup
subplot(2,2,[3,4])
imshow(NoisePup);                  % display of grayscaled with noise
title('Grayscaled version with noise')
size(NoisePup)                     % checking size of picture

%% Part 3 - Questions 1 - 3Point Averager on Puppy

b0 = (1/3)*ones(3,1);
Averager3Pup = filter(b0,1,GrayPup);

figure(7)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(Averager3Pup);           % display of 3-point version
title('3 Point Averager on Puppy')

B0 = [1/3 1/3 1/3];
figure(8)
freqz(B0);                      % frequency response
title('3 Point Moving Averager relative to Puppy'), grid minor
```

```
rmse = sqrt(immse(Averager3Pup,GrayPup))
size(Averager3Pup)                % checking size of picture

%% Part 3 Question 2 - 10Point Averager on Puppy

b1 = (1/10)*ones(10,1);
Averager10Pup = filter(b1,1,GrayPup);

figure(9)
subplot(1,2,1)
imshow(GrayPup);                  % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(Averager10Pup);            % display of grayscaled version
title('10 Point Averager on Puppy')

B1 = (1/10)*ones(1,10);
figure(10)
freqz(B1);                        % frequency response
title('10 Point Moving Averager relative to Puppy'), grid minor
rmse = sqrt(immse(Averager10Pup,GrayPup))
size(Averager10Pup)               % checking size of picture

%% Part 3 Question 3 First Difference Filter on Puppy

bk0 = [1 -1];       % vector of coefficients
figure(11)
freqz(bk0,1);       % frequency response
title('First Difference Filter relative to Puppy'), grid minor

DiffPup = filter(bk0,1,GrayPup);
figure(12)
subplot(1,2,1)
imshow(GrayPup);                  % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
```

```
imshow(DiffPup);                % display of first difference filter
title('First Difference Filter on Puppy')
rmse = sqrt(immse(DiffPup,GrayPup))
size(DiffPup)               % checking size of picture

%% Part 3 Question 4 - FIR Filter on Puppy

bk1 = [1,0,1];      % vector of coefficients
figure(13)
freqz(bk1,1);       % frequency response
title('FIR Filter relative to Puppy'), grid minor

FIRPup = filter(bk1,1,GrayPup);
figure(14)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(FIRPup);                 % display of first difference filter
title('FIR Filter on Puppy')
rmse = sqrt(immse(FIRPup,GrayPup))
size(FIRPup)               % checking size of picture

%% Part 3 Questions 5 - 2D Filter on Puppy

bb = [0.25,-1,0.25;-1,3,-1;0.25,-1,0.25];   % vector of coefficients
figure(15)
freqz2(bb);                         % frequency response
title('2D Filter Frequency Response of Puppy'), grid minor

TwoDPup = filter2(bb,GrayPup, 'same');
figure(16)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
```

```
imshow(TwoDPup);                % display of 2D filter on puppy
title('2D Filter of Puppy')
rmse = sqrt(immse(TwoDPup,GrayPup))
size(TwoDPup)              % checking size of picture

%% Part 3 Question 6 - Sobel Filter on Puppy

s = [1,2,1;0,0,0;-1,-2,-1];              % vector of coefficients
figure(17)
freqz2(s);                          % frequency response
title('Sobel Frequency Response of Puppy'), grid minor

SobelPup = filter2(s,GrayPup, 'same');
figure(18)
subplot(1,2,1)
imshow(GrayPup);                % display of grayscaled version
title('Grayscale of Puppy')
subplot(1,2,2)
imshow(SobelPup);               % display of sobel filter on puppy
title('Sobel Filter of Puppy')
rmse = sqrt(immse(SobelPup,GrayPup))
size(SobelPup)             % checking size of picture
```