

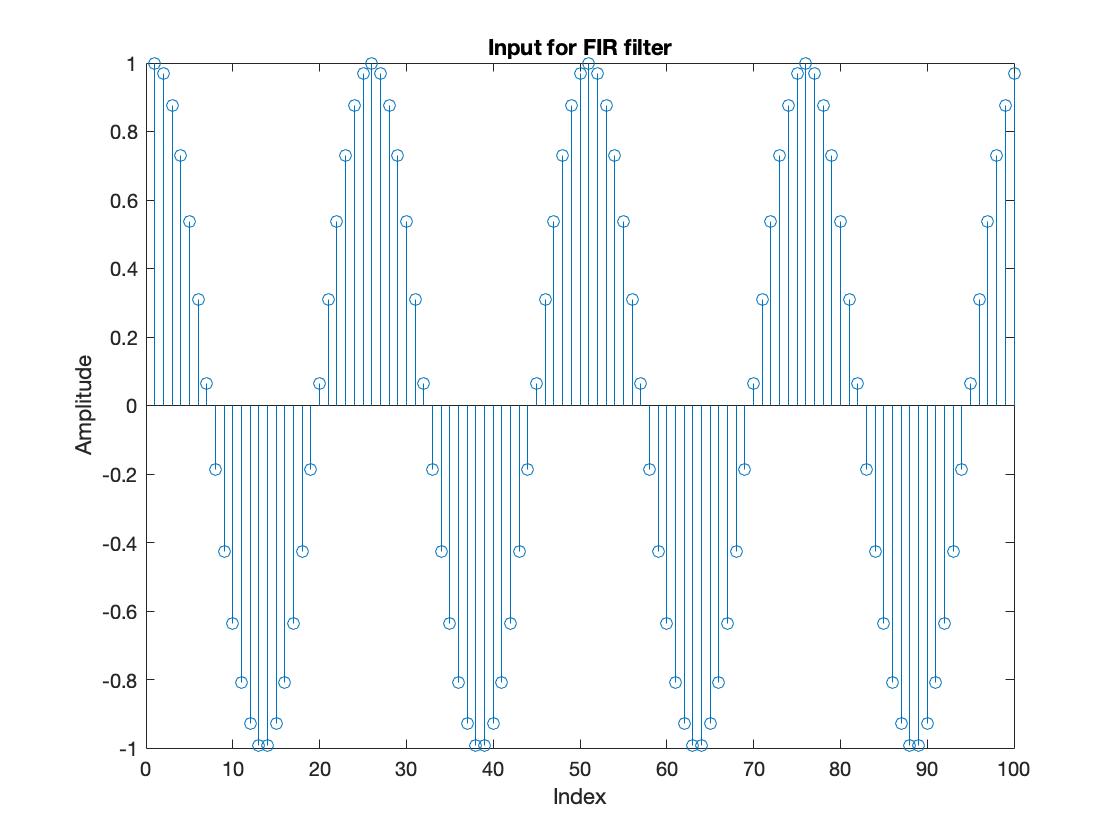
EE 386 Lab 4

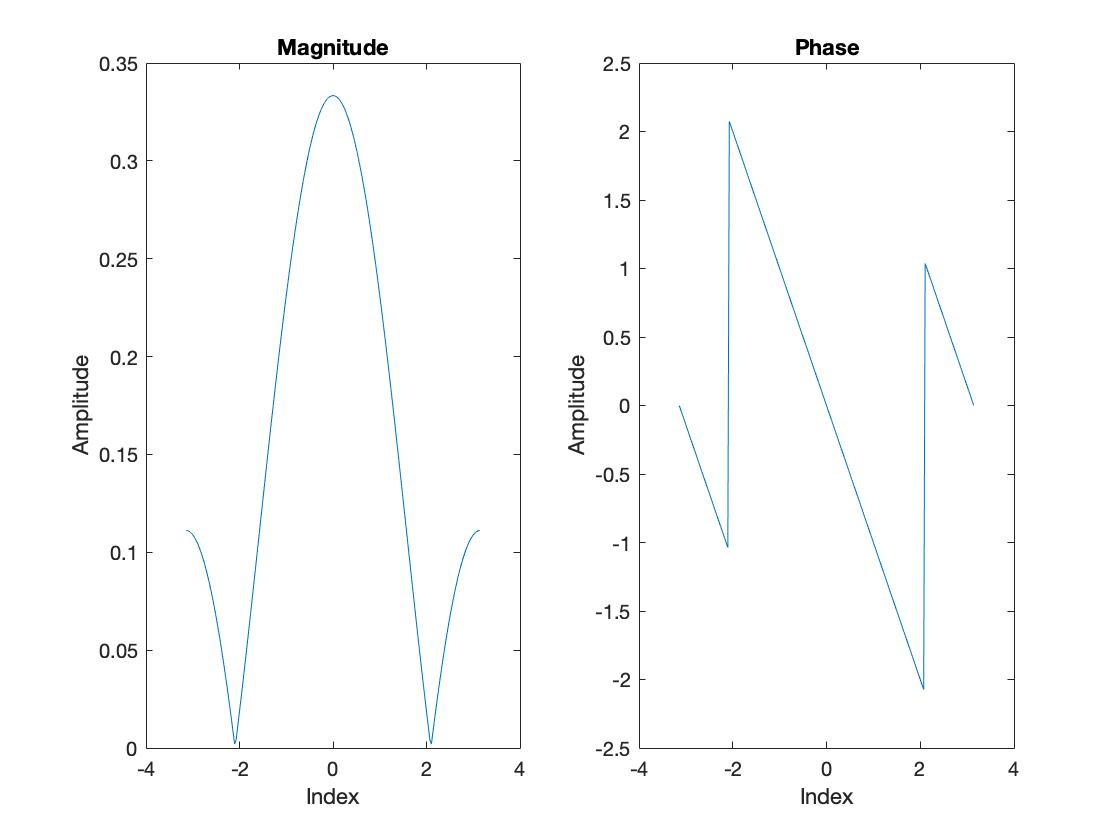
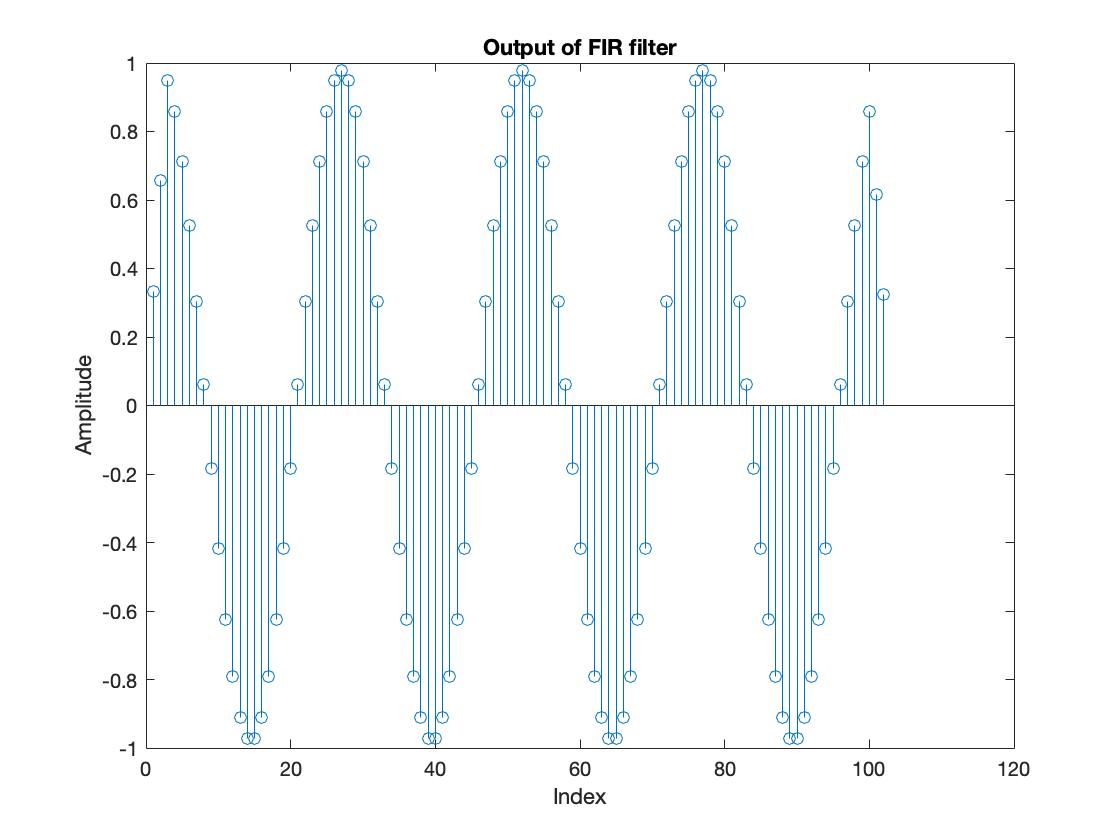
FIR Filtering and Image Processing

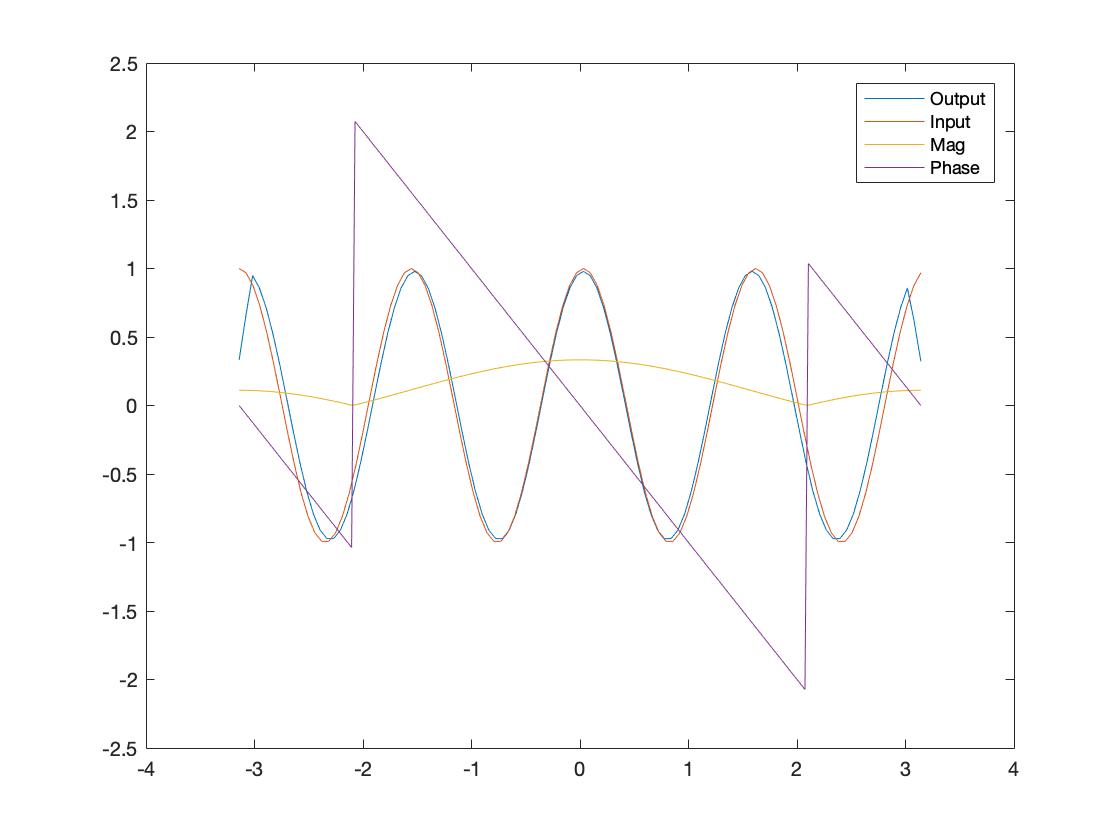
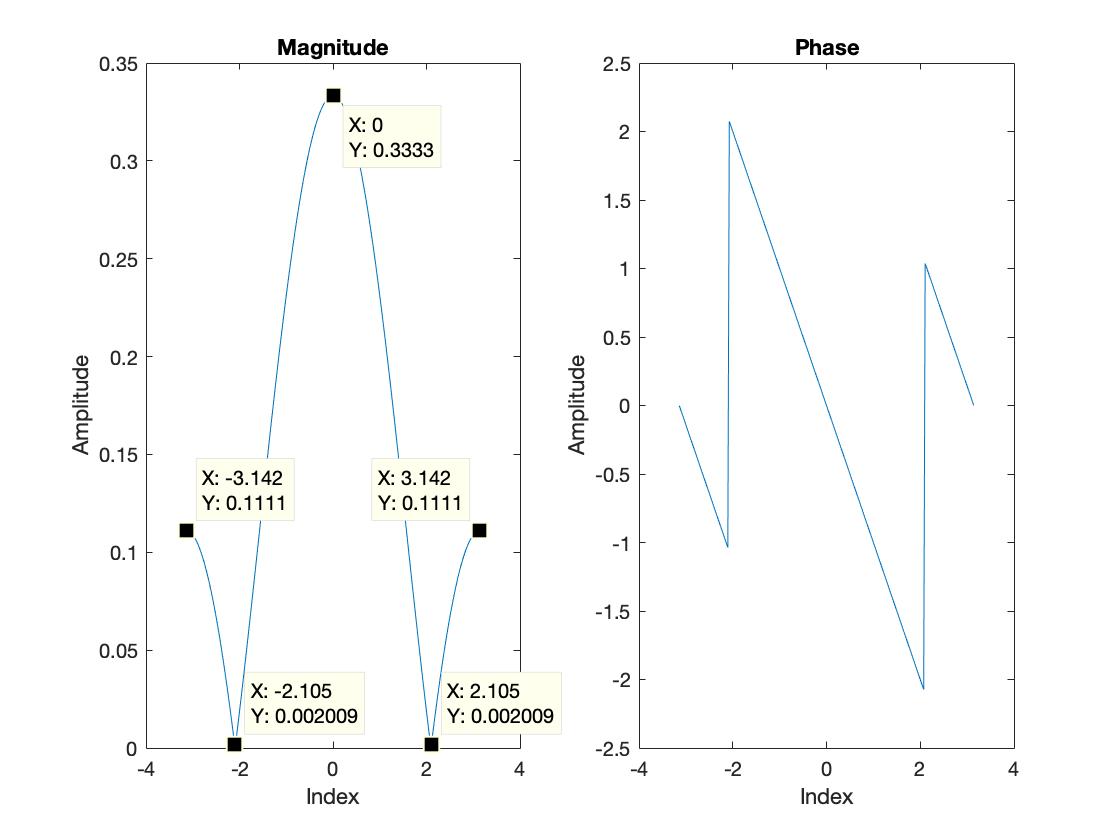
By. Alex Jackson and Christopher W. Fingers

Lab Instructor Chris Hirunthanakorn

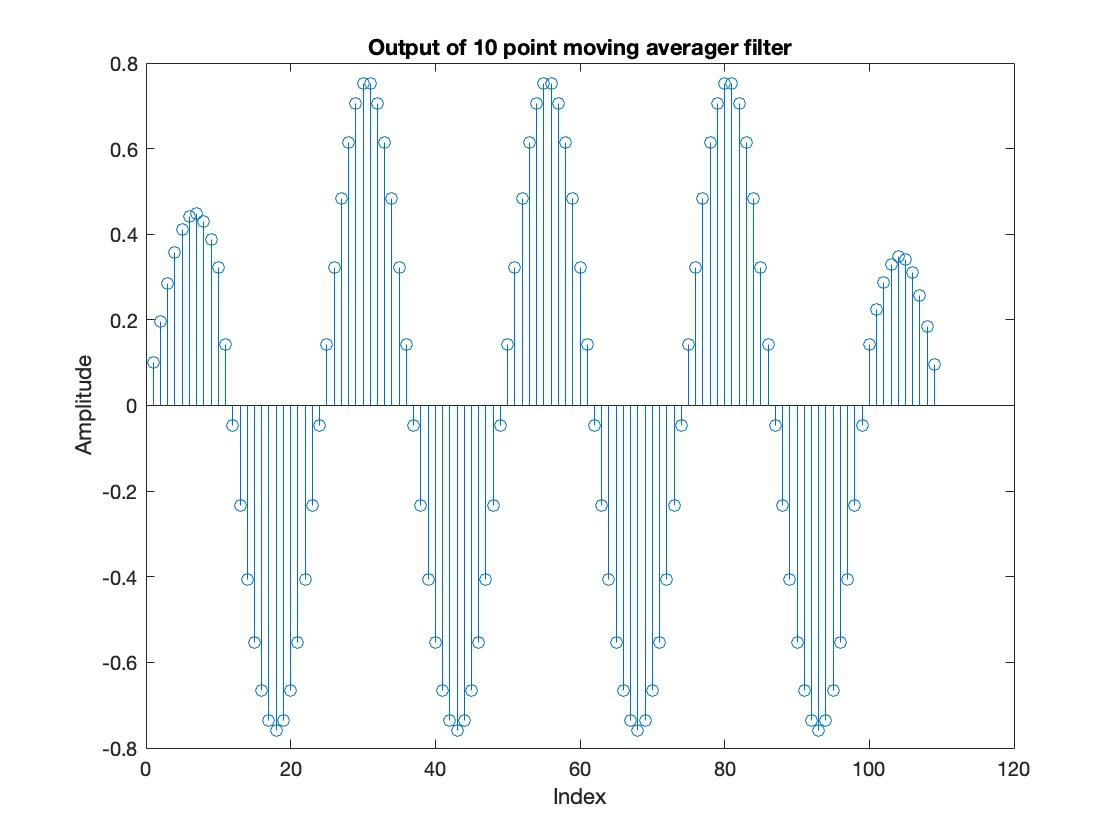
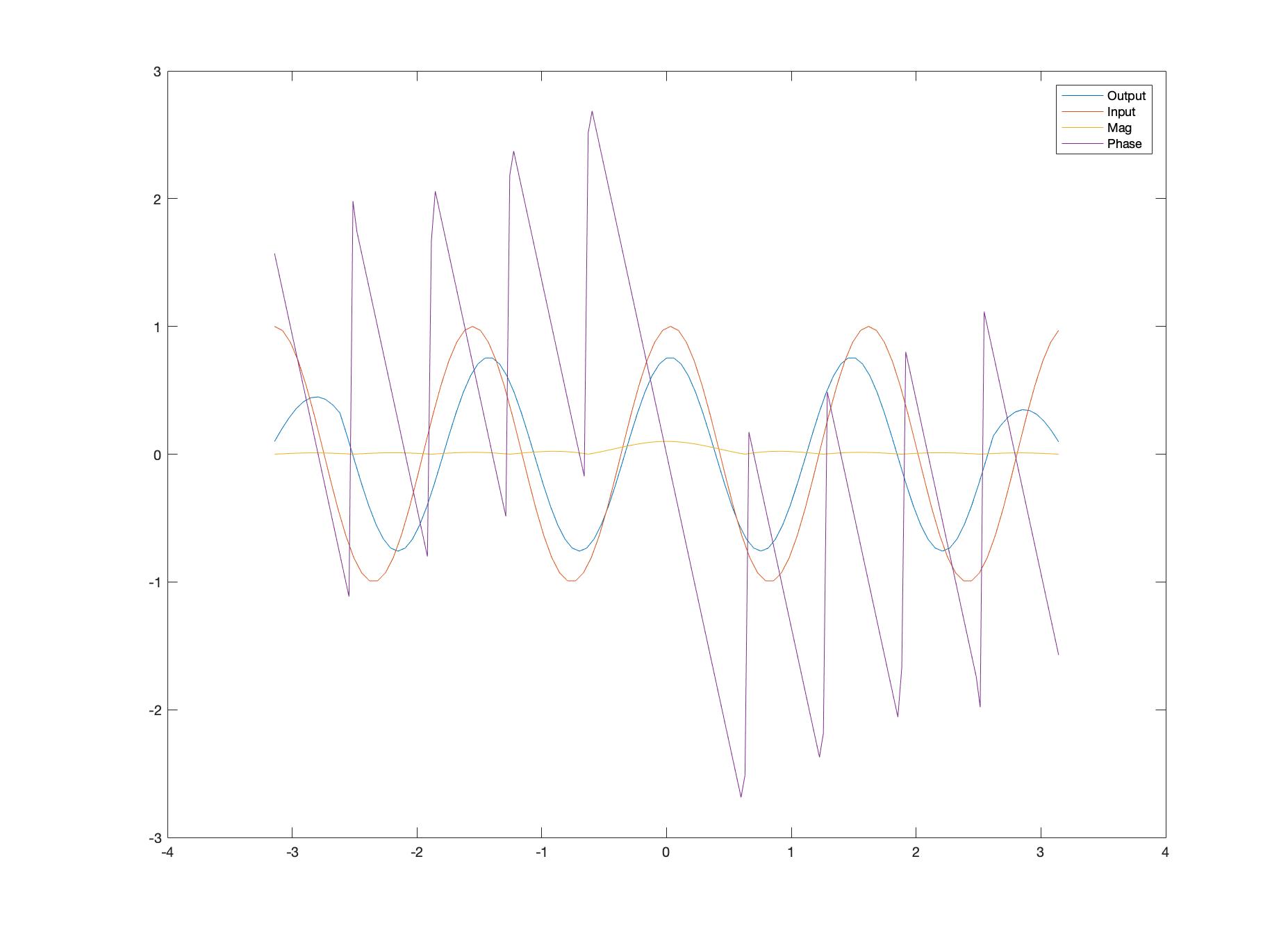
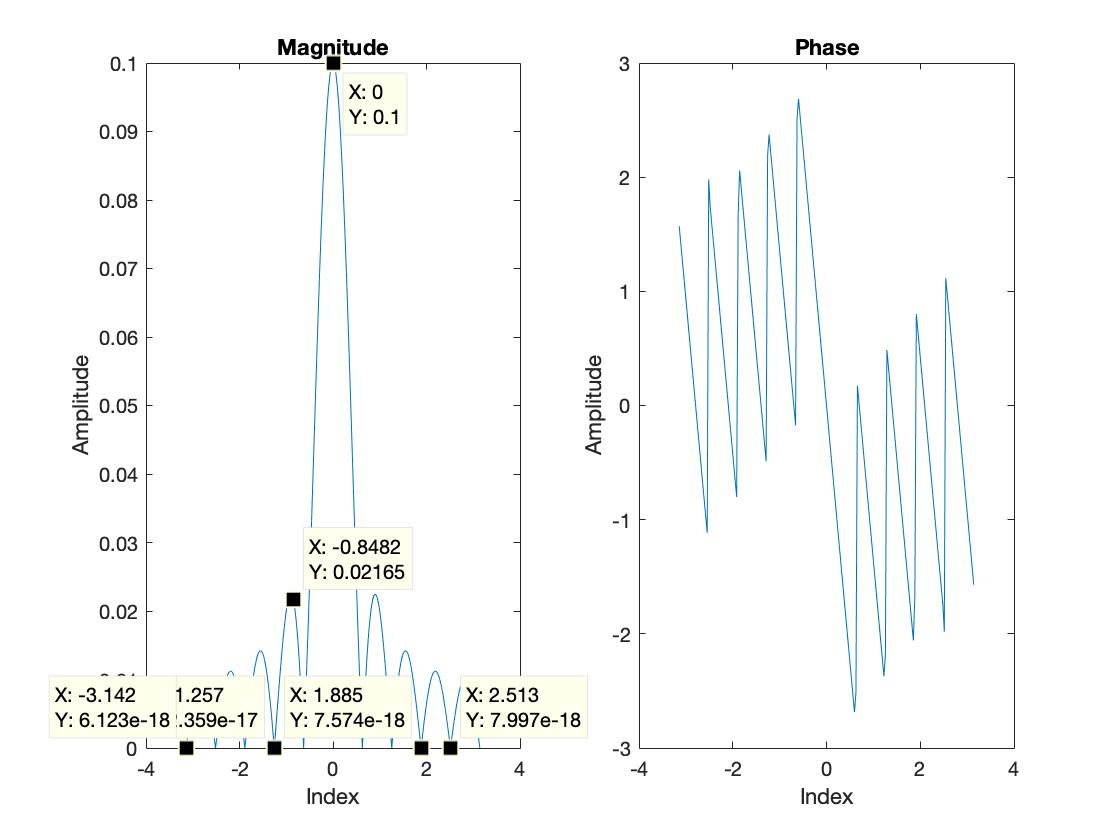
**Part 1:**



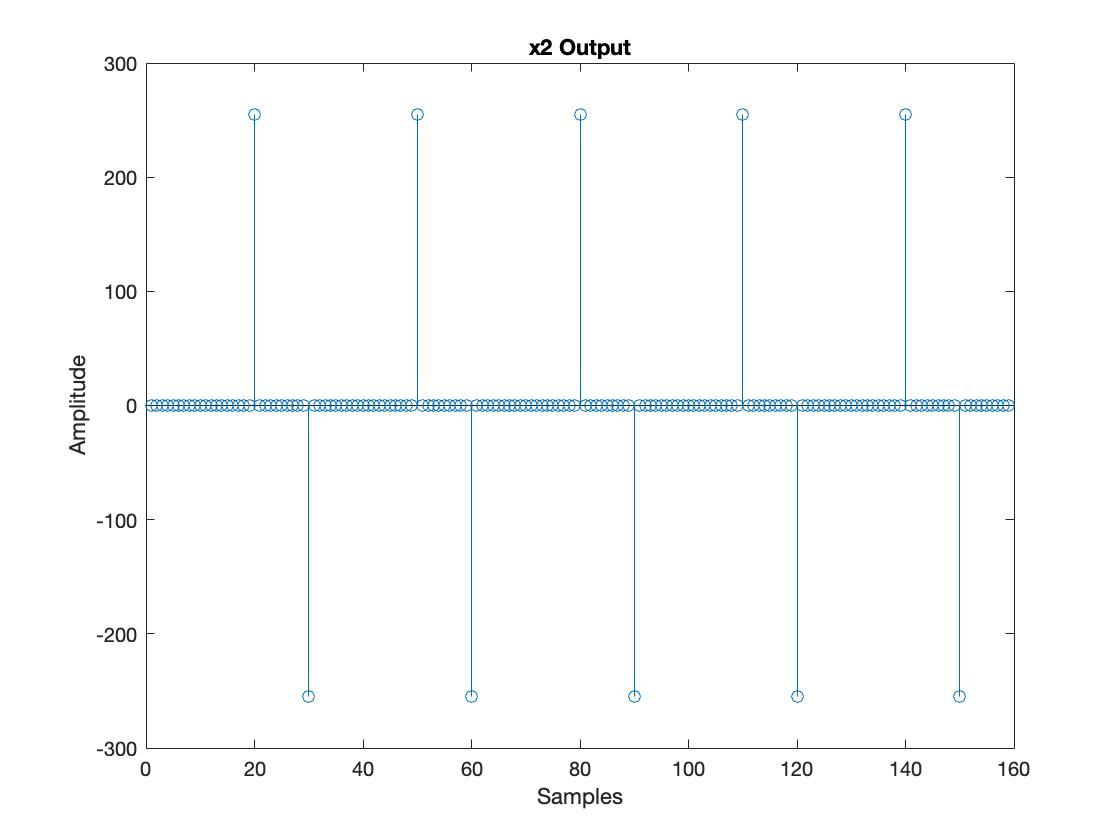
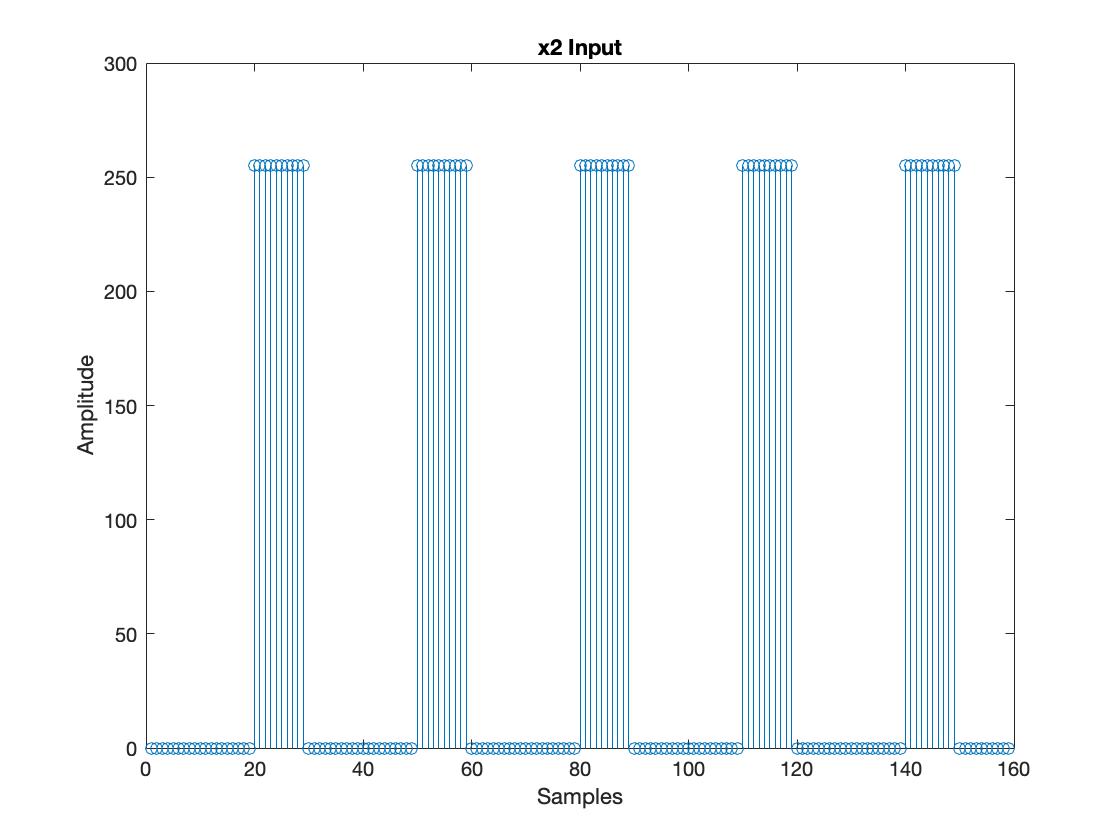




For this system, there is a shift to the right and little time delay looking at the input and output of the system. When the phase begins with a negative slope, we know that the system is introducing a delay of one sample looking at the slope of the phase.

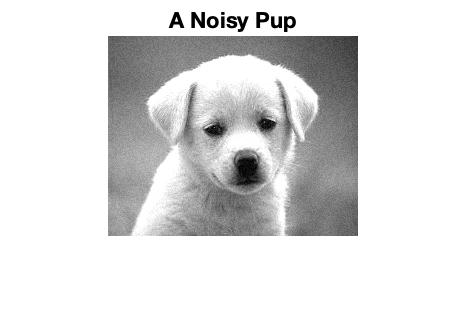
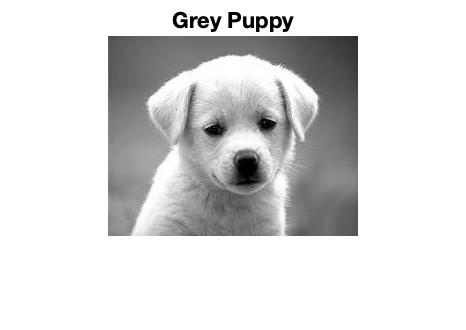
 

Looking at the input and the output, the phase and amplitude of the system are altered according to the filter applied. The output is shifted to the right with a time delay. This is seen in the phase of system looking at where the phase begins. Since there is a negative slope of about negative 4.63. This means that there is a delay of about 4 samples.

**Part 2:**

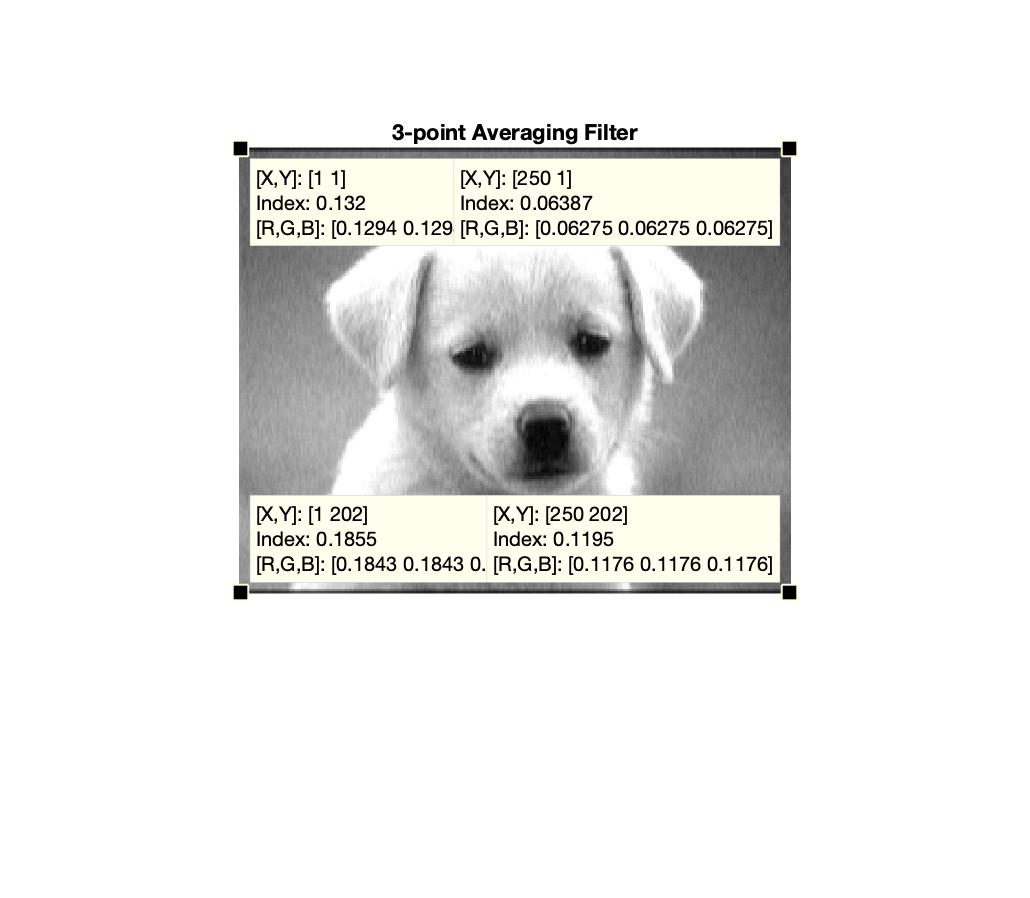
The input for the system appears the way it does because it is taking the remainder values from a series of numbers ranging from 1:159 and dividing each by 30. If the remainder value is greater than 19 then a 1 is stored for that number. This number is then multiplied by 255 after the remainder value is found. The output for this system is taking the collection of remainder values greater than 19 and breaking it down to the first and last value in each collection of remainder values. Each remainder value that is greater than 19 is multiplied by 255 then passed through the difference filter. The first input value is multiplied by the first part of the filter 1 and subtracted with the next remainder value. This is why the first value is a positive 255 amplitude and the next different amplitude that is not 0 is - 255. Both the input x2 and output have the same number of samples taken, but the output has the -255 one space after the input ends. This is because of how the difference filter works where it will take the next value after -255 which is 0 and adding together.

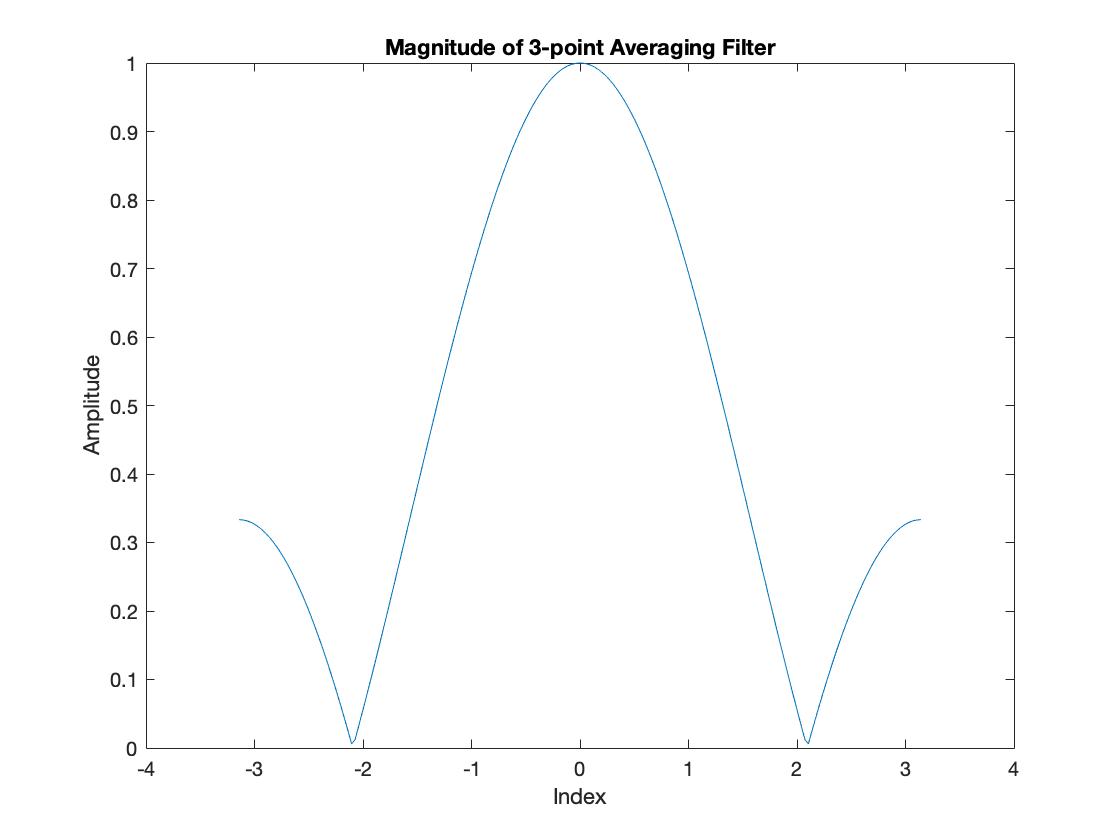
**Part 3:**



RMS value = 0.6584

**3 Point Moving Averager:**





1. The 3-point moving averager smooths out the noise seen in the original image. Convolving the filter with the noisy image, the image becomes more blurry and less sharp comparing to the noisy image.
2. The size of the new image is 202x250 comparing to the original image of size 200x250.
3. Observing the output the filter is a low pass filter.
4. The transient part of the image is the bottom and top row of the image where it is black. Here there are lower values of the index intensity and RGB values. This parts of the image are the transient response.
5. For this 3 point averager filter the magnitude of the transfer function looks like an absolute value graph with a positive and negative slope and peak at 1.
6. RMS = 0.6537

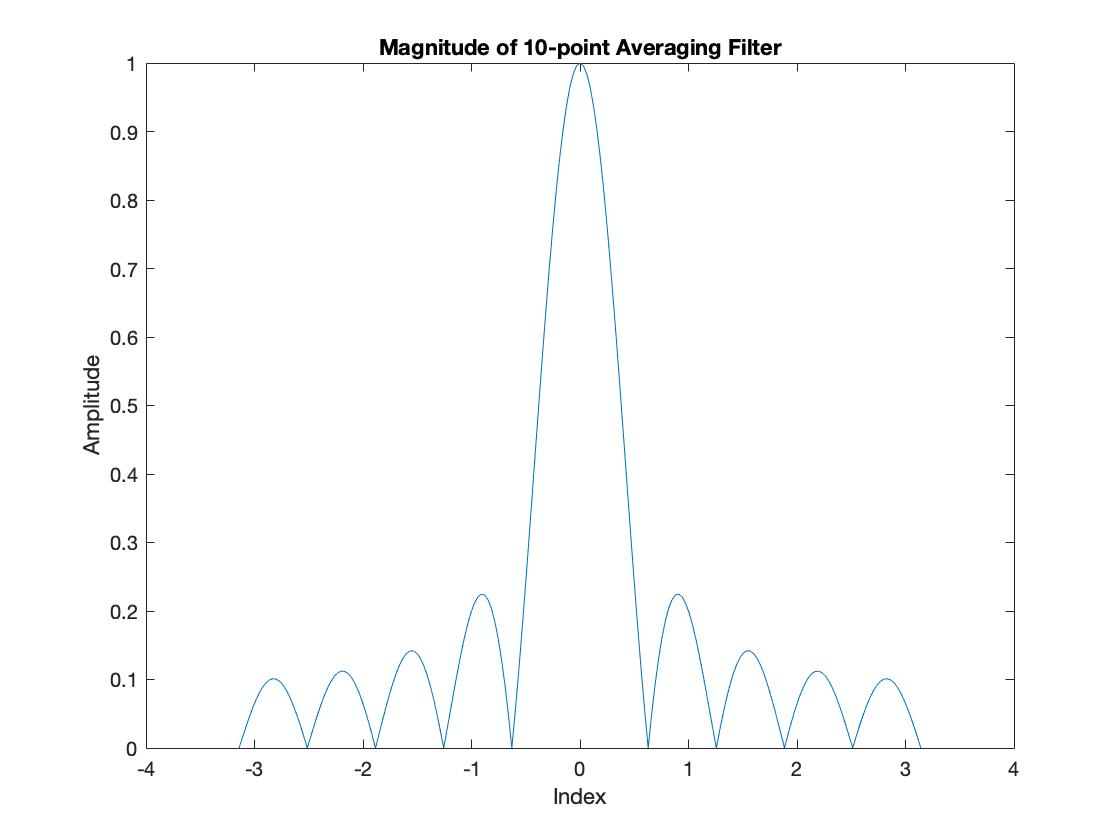
Out of the

Out of the other varied signals, the 3 point moving averager has the lowest error when compared to the original systems average RMS value.

**10-Point Moving Averager:**



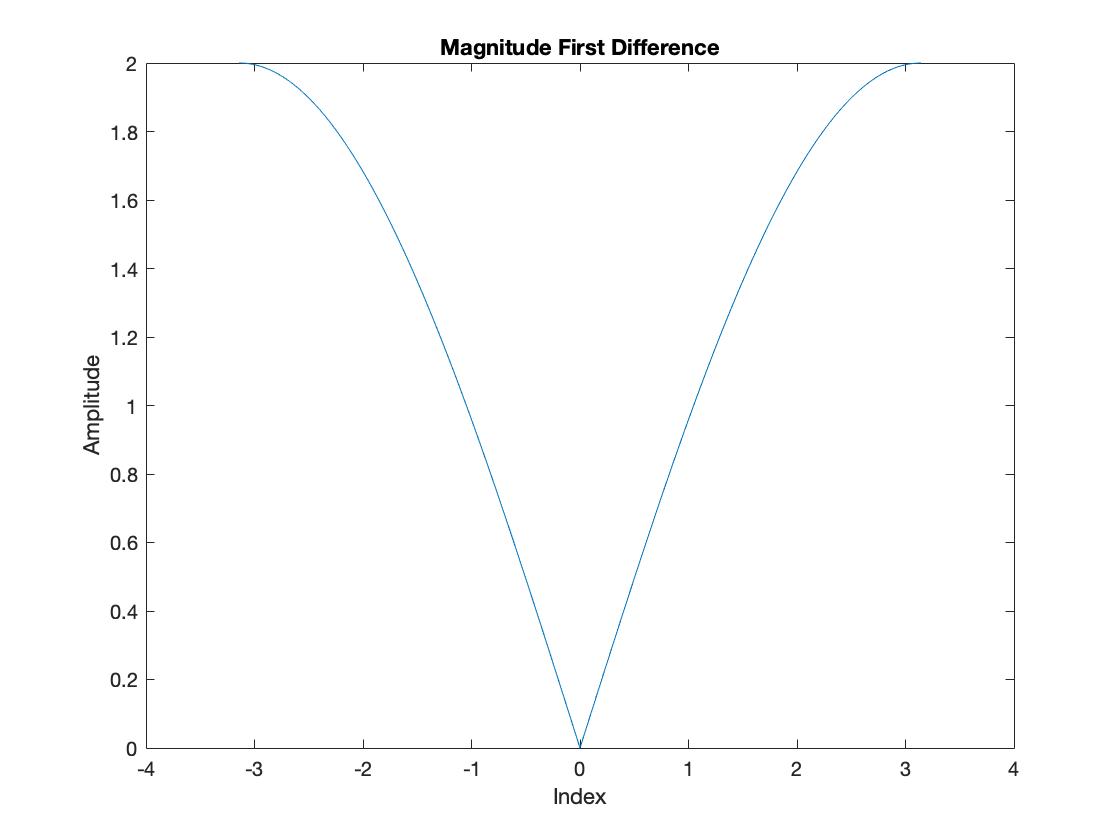




1. This image is a lot more blurry than the 3-point moving averager. Increasing the averager, increases the blurriness of the image.
2. The new size of the image is 209x250. The amount of columns for the images do not change, however the amount of rows does change. For this image the rows increased from 200 to 209. Comparing to the 3 point moving averager, how much the size of the image is increased depends on the averager. This is because the length of the convolved signal is the length of the two vectors then subtracting 1. This fits for both of the cases for the two averaging filters.
3. This filter is also a lowpass filter. Most of the averaging filters are low pass filters.
4. For this image, again the transient response is similar to the 3 point moving averager. The black top and bottom line of the image is the transient response. For this image since the size is increased even more the transient response is increase.
5. Comparing the magnitude of the 10 point to the 3 point, the 10 point has more side lobes. This is the transient response and output of the filter. The more lobes the higher the averager, and vice versa.
6. RMS=0.6384

**First Difference:**



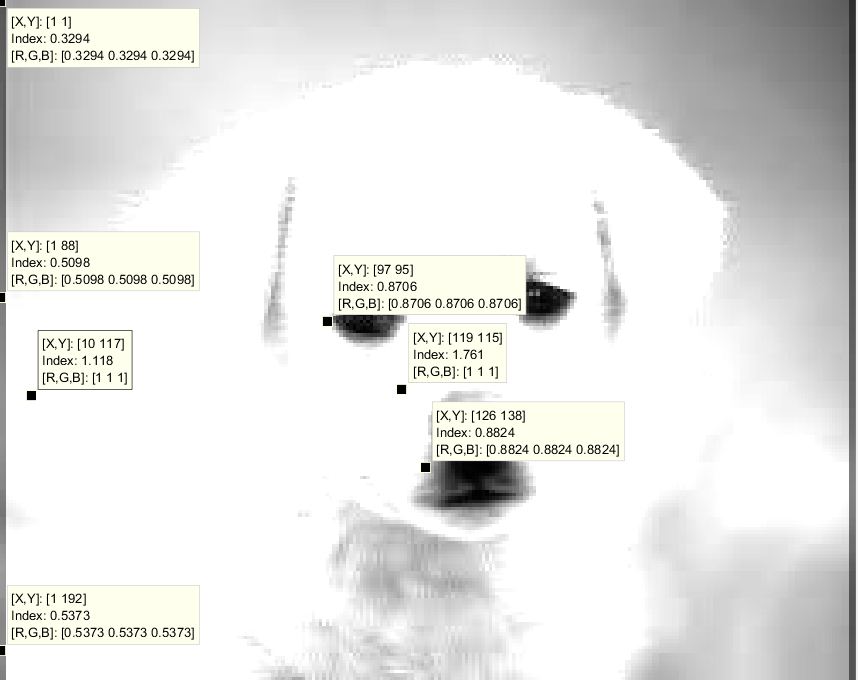


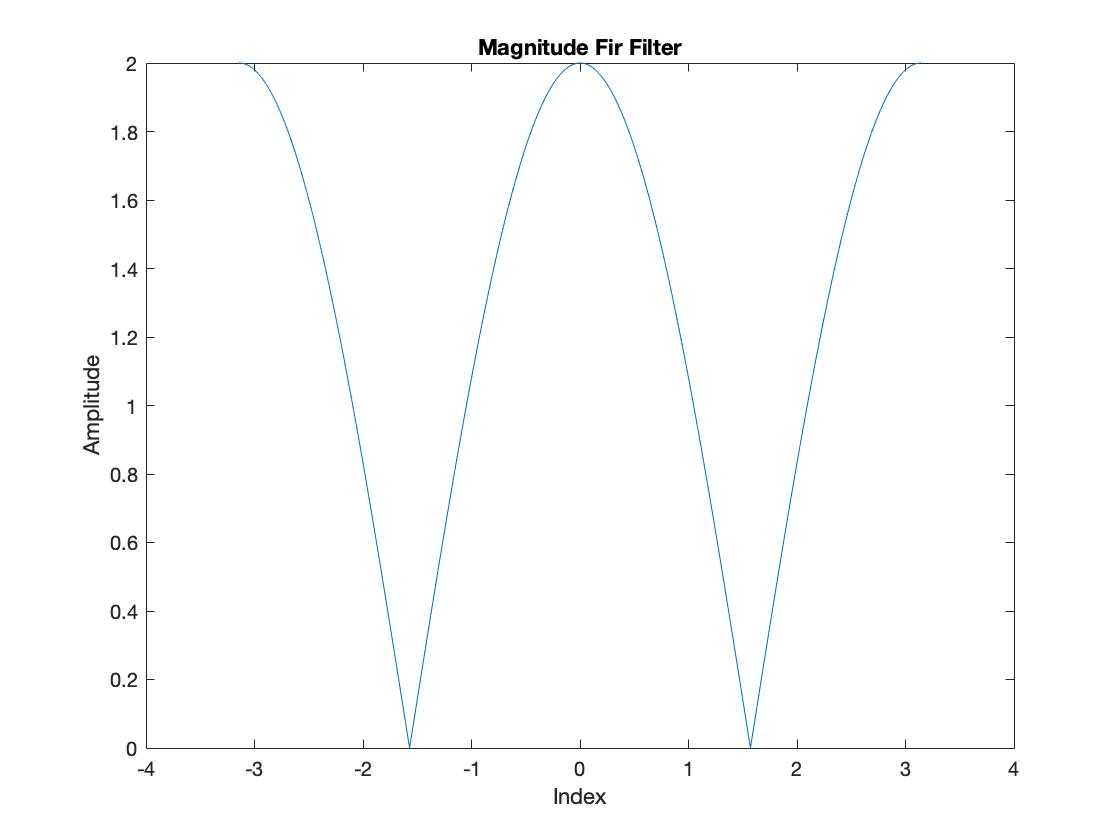


1. The first difference filter is mainly used to define the edges of an image, but moving from left to right. That is why the edges of the puppy on the left side is visible, but the right side of the puppy is not. The puppy is nearly invisible, but there is a line that outlines the edges of the puppy’s side, eyes, head, etc.
2. The size of the image is 200 x 251. This extra column comes up because of how the edges are being defined from left to right. The extra column appears because during the convolution of the two signals, the combined size -1 of the signals is the resulting output.
3. This is a high pass filter.
4. The transient response for a first difference filter occurs at the first column of the image and when an edge appears.
5. The main difference for this filter when compared to the others in terms of magnitude is that there is mainly only 2 slopes starting from zero and moving up to 2.
6. RMS = 0.0354

**FIR Filter:**

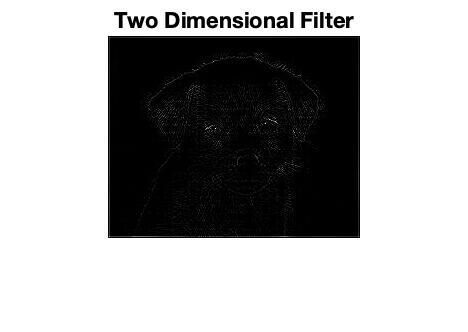


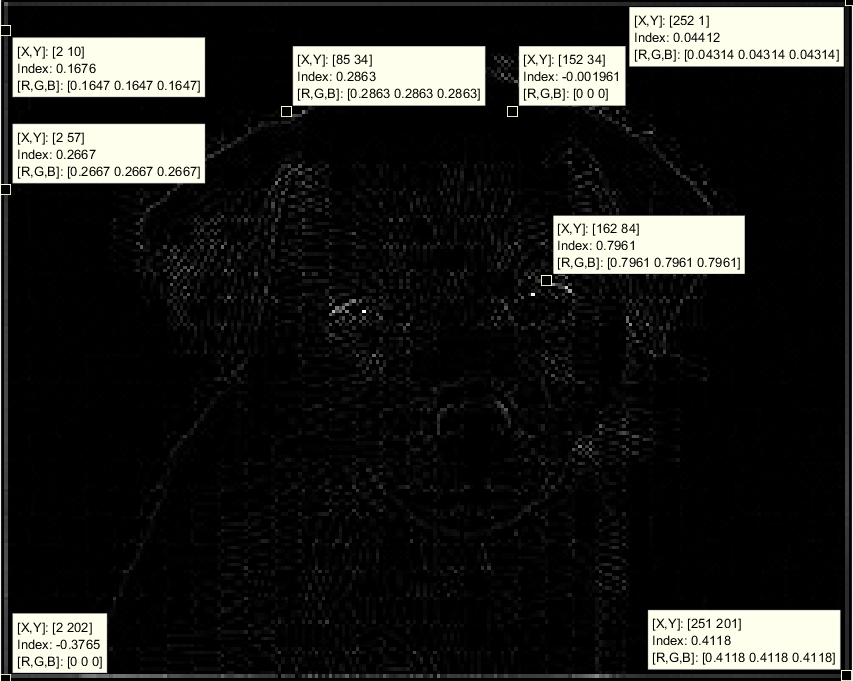


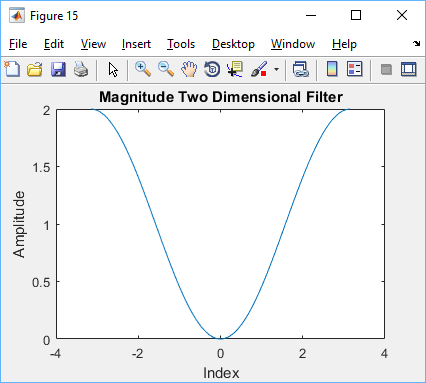


1. This filter magnifies the different shades of the filter. Areas that are more lighter are now completely white and darker areas are more well defined. The puppy has well defined eyes and nose that shows well with this filter, however the whiter areas of the puppy are hard to define.
2. This image is 200 x 252 in size.
3. This is a low pass filter.
4. The transient response for this filter occurs at the left and right sides of the image and during the transitional periods when going from white to black.
5. As shown by the output, this filter brightens up a picture and accentuates the different shades. The different shades of grey are much more noticeable with this filter and clearly defines the darker areas from the lighter ones.
6. RMS= 1.2090

**Two Dimensional Filter:**



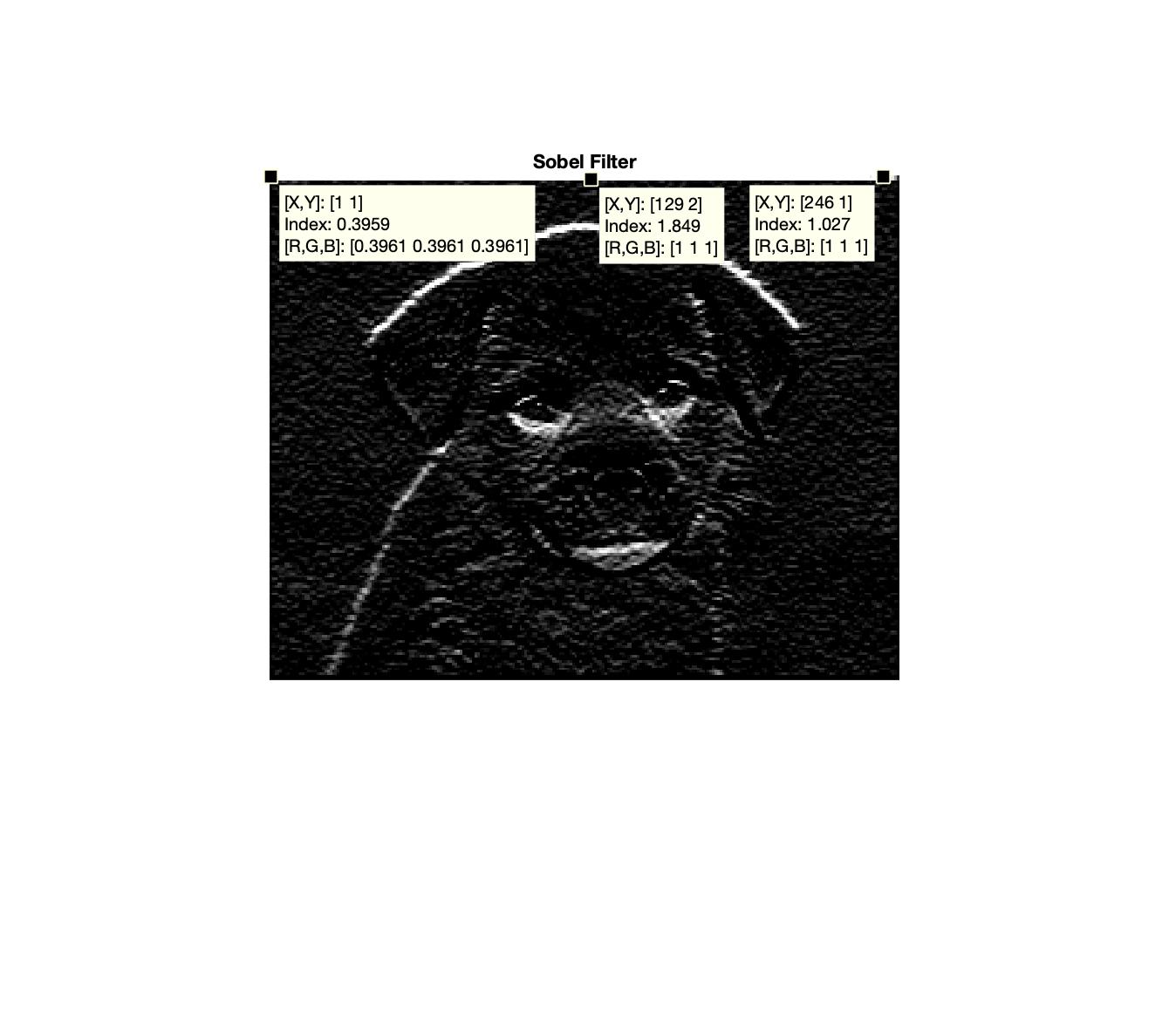


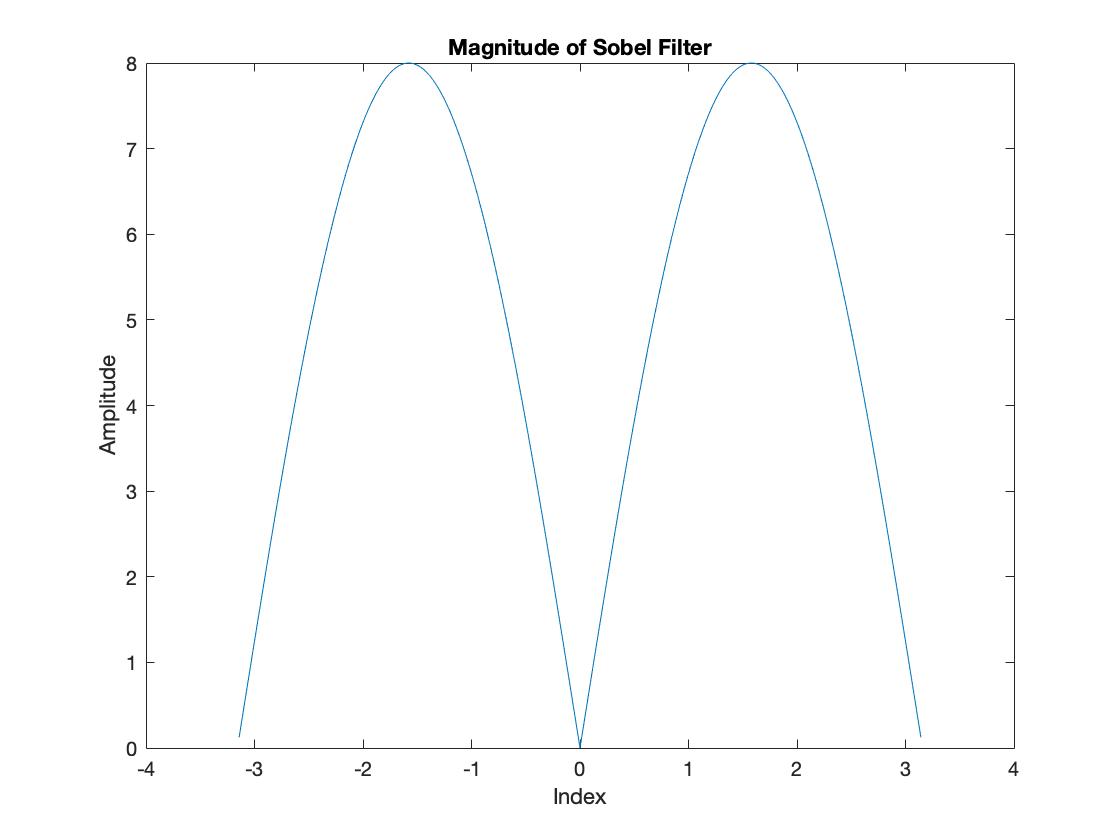


1. The two dimensional filter gives a very slight outline to the puppy and blacks out the background. This is similar to the difference filter, but not as well defined. The image is hard to make out what it is, but there is a faint outline that makes up the dogs head and body.
2. The image is a size of 202x252, increase both the rows and columns by n-1 and m-1.
3. Based on the output this is a bandpass filter. The edges are somewhat defined, however there is still some edges that are not very visible. There is also some points that are being picked up by the filter in areas of varying shade.
4. The transient response starts at all sides of the picture and when there is an edge on the puppy.
5. In terms of magnitude, this shows a gradual increase up to an amplitude of 2 and instead of having a sharp jump from 0 amplitude up, the 2d filter gradually decreases to 0 then increases back up to the desired amplitude.
6. RMS = 0.0602

**Sobel Filter:**

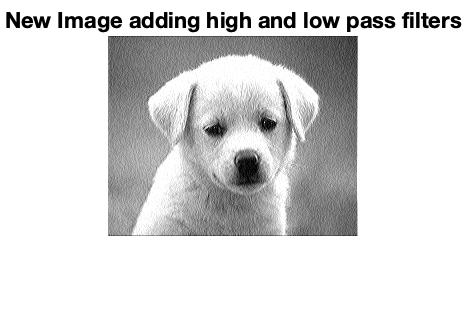






1. For this filter, the edges of the puppy is more defined. The horizontal lines and edges are more defined than the vertical lines. This is because we only convolved the image one way instead of doing both the row and column. Doing both directions would increase the sharpness and define the edges even more so than this image.
2. The new size of this image is 202x252. For this filter, both the row and columns are increased by 2. This is because the matrix of the sobel filter has a transient response in both directions versus just the one direction.
3. Based on the output the filter is a high pass filter. Typically high pass filters detect edges, such as this filter output. This is also similar to first difference filter since it is an edge detector, but of higher quality. The point of edge detection is to detect the change and discounties throughout the image.
4. It is harder to detect the transient part of the image. However, based on the previous filters, we can assume the transient response is one line of the the right and left and top and bottom of the image. After looking at the detection points the white bar towards the top of the image can be seen as the transient response as seen in the image.
5. This filter is similar to the difference filter, this makes sense since it displays the edge of the image. First difference filters are edge detectors as seen above. In terms of magnitude the sobel filter displays two main lobes excluding any frequency at zero. These two lobes go from zero to pi and is mirrored going from zero to negative pi.

6. RMS = 0.357



To obtain the original image, we can add a high pass with a low pass and obtain the original image.

Our code does not use filter2, but conv2 to produce the correct image, however the ‘same’ input that is used in filter2 returns the appropriate size transfer function when performed. This is needed because during convolution the size of the output vector is increased by the same amount as the size of the convolved figure -1. This will make it so there are too many single points on the y axis when compared to the x axis and the graph will not be able to be plotted in Matlab.

**Matlab Code:**

%% Lab 4

% Part 1 (Alex)

% a

nn=0:99;

x1=cos(.08\*pi\*nn);

bn=ones(3,1)/3;

filt=firfilt(bn, x1);

% b

% input and output of filter

figure(1)

stem(x1);

title('Input for FIR filter')

xlabel('Index')

ylabel('Amplitude')

figure(2)

stem(filt);

title('Output of FIR filter')

xlabel('Index')

ylabel('Amplitude')

% c

% measure the magnitude and phase

w\_hat=-pi:pi/100:pi;

H=freqz(bn,1,w\_hat);

figure(3)

subplot(1,2,1)

plot(w\_hat, abs(H))

title('Magnitude')

xlabel('Index')

ylabel('Amplitude')

subplot(1,2,2)

plot(w\_hat, angle(H))

title('Phase')

xlabel('Index')

ylabel('Amplitude')

% f

% 10 point moving averager

%nn=-pi:pi/100:pi;

nn=0:99;

bk=ones(10,1)/10;

x1=cos(.08\*pi\*nn);

yn=conv(bk,x1);

figure(4)

stem(yn);

title('Output of 10 point moving averager filter')

xlabel('Index')

ylabel('Amplitude')

w\_hat=-pi:pi/100:pi;

H=freqz(bk,1,w\_hat);

figure(5)

subplot(1,2,1)

plot(w\_hat, abs(H))

title('Magnitude')

xlabel('Index')

ylabel('Amplitude')

subplot(1,2,2)

plot(w\_hat, angle(H))

title('Phase')

xlabel('Index')

ylabel('Amplitude')

figure(1) % comparing phase

w\_hat1=-pi:pi/50.5:pi;

plot(w\_hat1,filt)

hold on

plot(-pi:pi/49.5:pi,x1)

hold on

plot(w\_hat, abs(H))

hold on

plot(w\_hat,angle(H))

legend('Output','Input','Mag','Phase')

figure(2) % comparing phase

w\_hat1=-pi:pi/54:pi;

plot(w\_hat1,yn)

hold on

plot(-pi:pi/49.5:pi,x1)

hold on

plot(w\_hat, abs(H1))

hold on

plot(w\_hat,angle(H1))

legend('Output','Input','Mag','Phase')

%% Part 2 (Chris)

% A

x2=255\*(rem(1:159,30)>19);

% B Fir filter

bb = [1 -1]

output = filter(bb,1,x2);

figure(6)

stem(x2)

xlabel('Samples');

ylabel('Amplitude');

title('x2 Input');

figure

stem(output)

xlabel('Samples');

ylabel('Amplitude');

title('x2 Output');

%% Part 3

% rgb2gray grays out the picture of the puppy and imshow displays the

% image.

GreyScaledPuppy = rgb2gray(Puppy);

GreyScaledPuppyDouble=im2double(GreyScaledPuppy);

figure(7)

imshow(GreyScaledPuppyDouble)

title('Grey Puppy')

HighValue = max(max(GreyScaledPuppy))

LowValue = min(min(GreyScaledPuppy))

noise= .1.\*rand(200,250);

Noiseypup=noise + GreyScaledPuppyDouble;

figure(8)

imshow(Noiseypup)

title('A Noisy Pup')

%% Part 3

pup=rgb2gray(Puppy);

figure(9)

imshow(pup)

pup=im2double(pup);

noise=.1\*rand(200,250);

jig=pup+noise;

figure(10)

imshow(jig)

%% Different Filters Section

% Part 1 (Alex)

filt1=conv2(bn,jig); % 3 point

figure(11)

imshow(filt1)

title('3-point Averaging Filter')

H1=freqz(bn,1,w\_hat);

figure(5)

plot(w\_hat, abs(H1))

title('Magnitude of 3-point Averaging Filter')

xlabel('Index')

ylabel('Amplitude')

RMS1= rms(filt1)

RMS2= sum(RMS1)/length(RMS1)

%% part 2 (Alex)

filt2=conv2(bk,jig); % 10 point

figure(12)

imshow(filt2)

title('10-point Averaging Filter')

H2=freqz(bk,1,w\_hat);

figure(5)

plot(w\_hat, abs(H2))

title('Magnitude of 10-point Averaging Filter')

xlabel('Index')

ylabel('Amplitude')

RMS3 = rms(filt2);

RMSr = sum(RMS3)/length(RMS3)

%% Part 3 (Chris)

FdF=[1 -1];

differenceconv=conv2(GreyScaledPuppyDouble,FdF);

figure(13)

imshow(differenceconv)

title('First Difference')

H3=freqz(FdF,1,w\_hat);

figure(5)

plot(w\_hat, abs(H3))

title('Magnitude First Difference')

xlabel('Index')

ylabel('Amplitude')

RMS4=rms(differenceconv);

RMSr1=sum(RMS4)/length(RMS4)

%% Part 4 (Chris)

FcF=[1 0 1];

Cconv=conv2(GreyScaledPuppyDouble,FcF);

figure(14)

imshow(Cconv)

title('Fir Filter')

H4=freqz(FcF,1,w\_hat);

figure(5)

plot(w\_hat, abs(H4))

title('Magnitude Fir Filter')

xlabel('Index')

ylabel('Amplitude')

RMS5 = rms(Cconv);

RMr2 = sum(RMS5)/length(RMS5)

%% Part 5 (Chris)

bb=[0.25,-1,0.25;-1,3,-1;0.25,-1,0.25];

Dconv=conv2(GreyScaledPuppyDouble,bb);

figure(15)

imshow(Dconv)

title('Two Dimensional Filter')

H = freqz2(bb,[201 1]);

plot(w\_hat, abs(H))

title('Magnitude Two Dimensional Filter')

xlabel('Index')

ylabel('Amplitude')

RMS6 = rms(Dconv);

RMSr3 = sum(RMS6)/(length(RMS6))

%% part 6 (Alex)

w\_hat=-pi:pi/100:pi;

s= [1, 2, 1; 0, 0, 0; -1, -2, -1];

filt6=conv2(jig,s);

H = freqz2(s,[201 1]);

figure(17)

imshow(filt6)

title('Sobel Filter')

figure(18)

plot(w\_hat,abs(H))

title('Magnitude of Sobel Filter')

xlabel('Index')

ylabel('Amplitude')

RMS7 = rms(filt6);

RMSr4 = sum(RMS7)/length(RMS7)

% Adding high pass with low pass to get image

y1 = filter2(FdF, jig) + filter2(bn, jig);

imshow(y1)

title('New Image adding high and low pass filters')