Homework 3 - Student Report
CMPE 362 Spring 2016

Batuhan Demir - 2013400159

April 17, 2016

1 Questions & Answers

1.1 Question 1 :

ClearNoise.m In this question after combining two sound and getting noisy sound i applied filter. The filter i used a 'buttord' filter try to cut the noisy sound.

As seen in the frequency domain figures noisy magnitudes of the noisy sound is lower than human voice. So after combine of sounds by designing this filter i tried to disable street sound. Input SNR value i found is **2.0641**

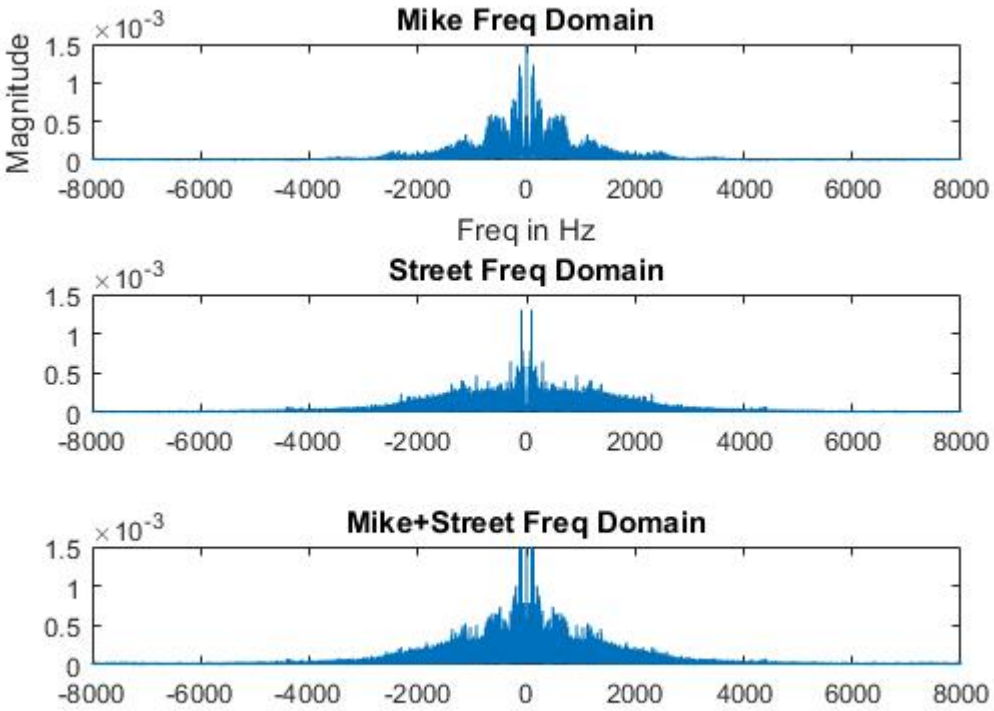

Figure 1: Frequency Domain Representations

```
[y1,F]=audioread('mike.wav');
[y2,F]=audioread('street.wav');

%%%%Create mike+street sound
y3 = [y1+y2];
audiowrite('noisy.wav',y3,F);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% FILTERING %%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Ws = 250/(F/2);
Wp = 3500/(F/2);
```

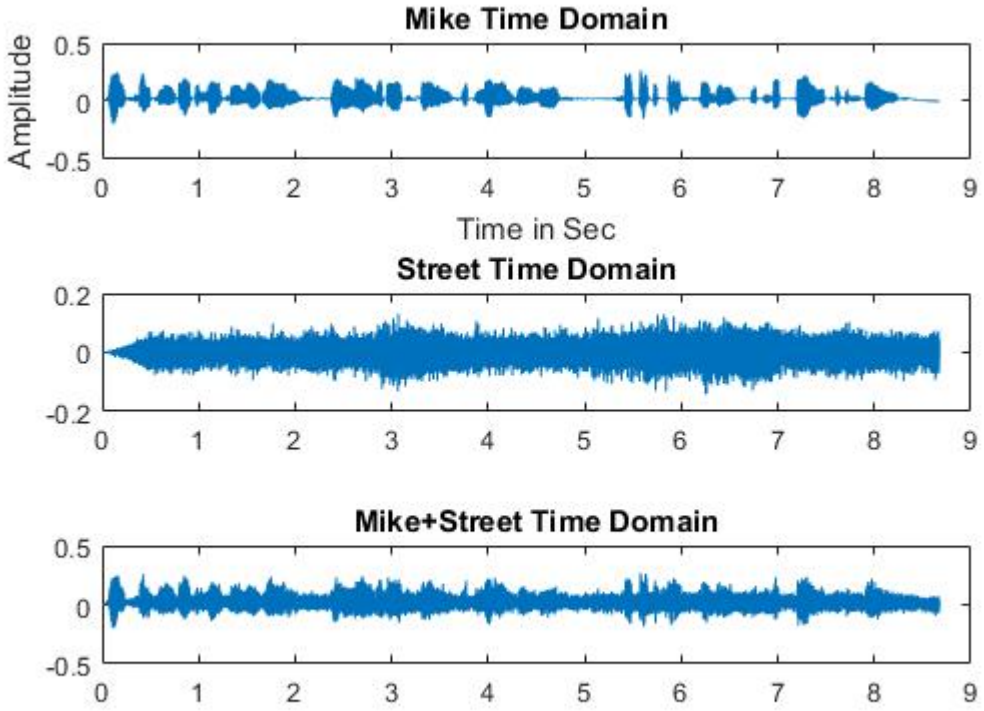

Figure 2: Time Domain Representations

```
[N,Wn] = buttord(Wp,Ws,0.5,50);
[b,a] = butter(N,Wn);
fout = filter(b,a,y3);

% Wn = 250/(F/2);                    % Normalized cutoff frequency
% [z,p] = butter(11,Wn,'high');  % Butterworth filter
%
% fout = filter(z, p, y3);

% sound(y3,F);
% pause(9);
% sound(fout,F);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% 1- FREQURNCY Domain %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure

N = size(y1,1);
df = F / N;
w = (-(N/2):(N/2)-1)*df;
y_1 = fft(y1(:,1), N) / N; %//For normalizing, but not needed for our analysis
yy = fftshift(y_1);
subplot(3,1,1)
plot(w,abs(yy));
xlim([-0.8e4 0.8e4]);
```

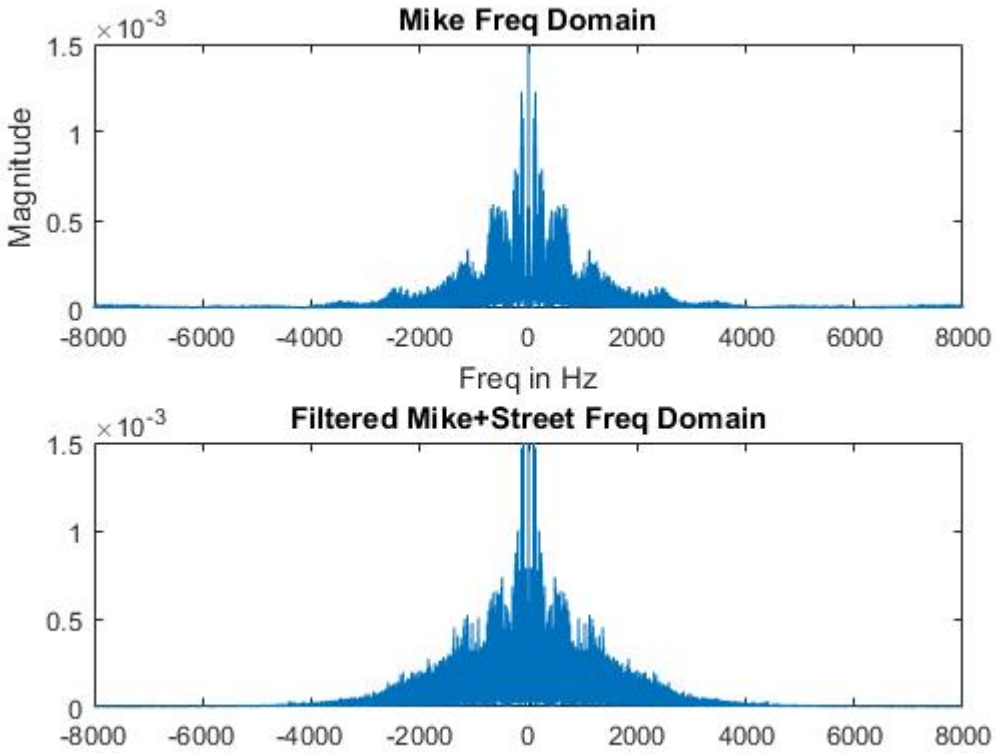

Figure 3: Actual and Filtered Freq. Domains

```
ylim([0 1.5e-3]);
xlabel('Freq in Hz');
ylabel('Magnitude');
title('Mike Freq Domain')
hold on;

N2 = size(y2,1);
df2 = F / N2;
w2 = (-(N2/2):(N2/2)-1)*df2;
y_2 = fft(y2(:,1), N2) / N2; %//For normalizing, but not needed for our analysis
yy2 = fftshift(y_2);
subplot(3,1,2)
plot(w2,abs(yy2));
xlim([-0.8e4 0.8e4]);
ylim([0 1.5e-3]);
title('Street Freq Domain')
hold on;

N3 = size(y3,1);
df3 = F / N3;
w3 = (-(N3/2):(N3/2)-1)*df3;
y_3 = fft(y3(:,1), N3) / N3; %//For normalizing, but not needed for our analysis
yy3 = fftshift(y_3);
subplot(3,1,3)
plot(w3,abs(yy3));
xlim([-0.8e4 0.8e4]);
```

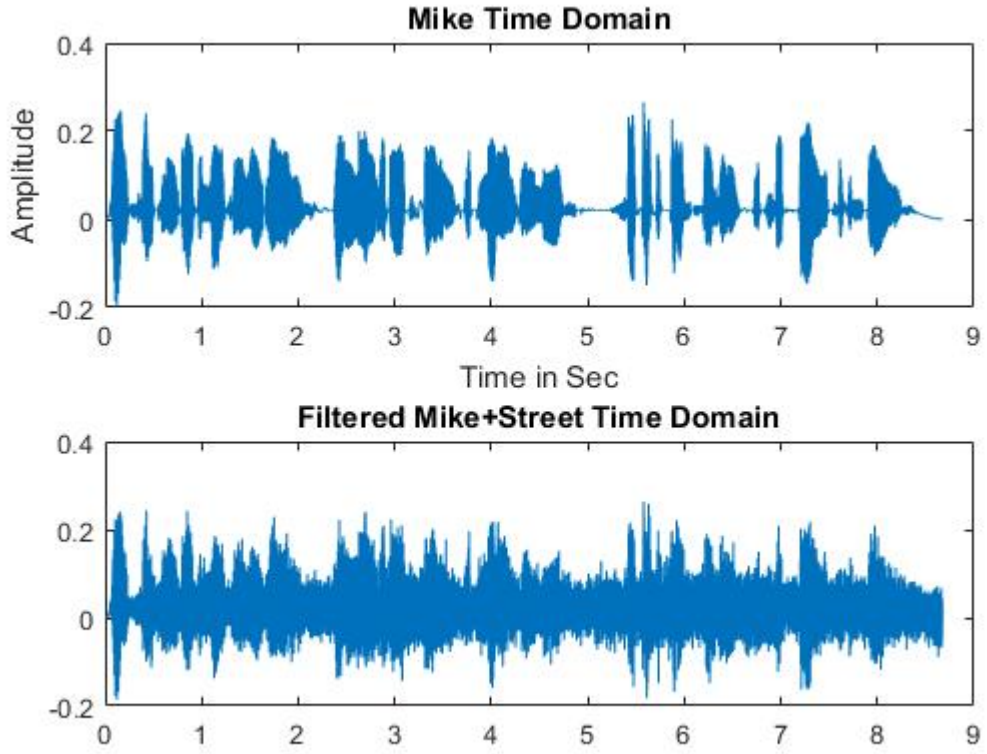

Figure 4: Actual and Filtered Time Domains

```
ylim([0 1.5e-3]);
title('Mike+Street Freq Domain')
hold on;



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% TIME Domain %%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure

t=0:1/F:(length(y1)-1)/F;
subplot(3,1,1)
plot(t,y1)
title('Mike Time Domain');
xlabel('Time in Sec');
ylabel('Amplitude');
hold on;

t1=0:1/F:(length(y2)-1)/F;
subplot(3,1,2)
plot(t1,y2)
title('Street Time Domain');
hold on;

t2=0:1/F:(length(y3)-1)/F;
```

```
subplot(3,1,3)
plot(t2,y3)
ylim([-0.2 0.4]);
title('Mike+Street Time Domain');
hold on;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%% 3- FREQUENCY domain%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure

subplot(2,1,1)
plot(w,abs(yy));
xlim([-0.8e4 0.8e4]);
ylim([0 1.5e-3]);
xlabel('Freq in Hz');
ylabel('Magnitude');
title('Mike Freq Domain');
hold on;

subplot(2,1,2)
NN3 = size(fout,1);
dff3 = F / NN3;
ww3 = (-(NN3/2):(NN3/2)-1)*dff3;
fout_1 = fft(fout(:,1), NN3) / NN3; %//For normalizing, but not needed for our analysis
ffout= fftshift(fout_1);
plot(ww3,abs(ffout));
xlim([-0.8e4 0.8e4]);
ylim([0 1.5e-3]);
title('Filtered Mike+Street Freq Domain')
hold on;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%% 4- TIME domain%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure

subplot(2,1,1)
plot(t,y1)
title('Mike Time Domain');
xlabel('Time in Sec');
ylabel('Amplitude');
hold on;

subplot(2,1,2)
t_fout=0:1/F:(length(fout)-1)/F;
plot(t_fout,fout)
title('Filtered Mike+Street Time Domain');
hold on;

%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%% SNR Value%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%

s = mean(y1.^2)/mean((fout-y1).^2);
snr = 10.^log(s);
```

1.2 Question 2 :

ClapDifferentiate.m Since i couldn't know the exact wav file to read, i read as '*.wav' but my program can only determines for 1 wav file. So this should be taken into consideration.

In the question, i first looked at frequency of the sound and then by trial and error method i determined that a clap contains high frequencies than snap. So i designed a highpass filter using *buttord* and *butter* functions. And then i look for max frequency in the domain. I determined my threshold as 60 Hz after filtering. If max frequency of the sound is below, it must be a snap, i say.

```
[stereo1, F] = audioread('*.wav');

mono1 = mean(stereo1,2);

% sound(stereo1,F);

figure
N = size(mono1,1);
df = F / N;
w = (-(N/2):(N/2)-1)*df;
y = fft(mono1(:,1), N) / N; %//For normalizing
yy = fftshift(y);
plot(w,abs(yy));
xlabel('Freq in Hz');
ylabel('Magnitude');
title('Freq Domain | Before')

% pause(2);

Ws = 5/(F/2);
Wp = 2200/(F/2);
[N,Wn] = buttord(Wp,Ws,0.5,10);
[b,a] = butter(N,Wn);
fout = filter(b,a,mono1);

% sound(fout,F);

figure
N = size(fout,1);
df = F / N;
w = (-(N/2):(N/2)-1)*df;
y = fft(fout(:,1), N) / N; %//For normalizing
yy = fftshift(y);
plot(w,abs(yy));
xlabel('Freq in Hz');
ylabel('Magnitude');
title('Freq Domain | After')

[max_value,idx] = max(fout);

if max_value<0.006
    disp('Snap sound detected')
else
    disp('Clap sound detected')
end
```

1.3 Question 3 :

Nand Despite of nearly hours of research on the Internet , i couldn't find particular solution to what we asked for, so i couldn't do this one.