**Lab 5： System, Convolution and Filter**

|  |  |
| --- | --- |
| **Author** | Name：郑聿钦 刘栩华 Student ID:11911615 11911906 |
| **Introduction**  1. Use filter to generate speech-shaped noise. Utilizing the speech spectrum as frequency response to filter white noise, we obtain corresponding speech-shaped noise (SSN).  2. Use filter to extract speech envelope waveform. After full-wave or half-wave rectification, speech envelope waveform could be extracted from signal with a low-pass filtering.  **Lab results & Analysis**：      The spectra was plotted as above figure.  From formula:  SNR could be adjusted to -5dB by multiplying a certain coefficient.      Since the original speech envelope waveform is too dense to be distinguished, only part of the whole envelope (from 1 to 1.2×104) was plotted to characterize these 3 low-pass filter.  Owing to more high-frequency components, the higher cutoff frequency is, the steeper and denser the envelope is.      Since the original speech envelope waveform is too dense to be distinguished, only part of the whole envelope (from 1 to 1.2×104) was plotted to characterize these 2 low-pass filter.  Owing to the stronger filtering capability, the higher the order of filter is, the flatter and looser the envelope is. | |
| **Experience:**  1. Filter could be used in speech signal processing, such as generating speech-shaped noise and extracting speech envelope waveform.  2. Considering the different properties of various filters, proper filter need to be generated according to the specific condition. | |
| **Score** |  |

filename='D:\Download\Lab5-System Convolution and Filtering\C\_01\_02.wav';

[x0, fs] = audioread(filename);

player = audioplayer(x0, fs); %play(player);

x=x0';

signal=repmat(x,1,10);

N=length(x);

noise = 1-2\*rand(1,N);

[Pxx,f] =pwelch(signal,[],[],512,fs);

figure(1);

plot(f,Pxx);

xlabel('frequency/Hz'),ylabel('power');

b = fir2(3000,f/(fs/2),sqrt(Pxx/max(Pxx)));

[h,wh] = freqz(b,1,128);

SSN= filter(b,1,noise);

SNR=20\*log10(norm(x)/norm(SSN));

SSN=SSN\*10^((5+SNR)/20);

y=x+SSN;

y=y\*norm(x)/norm(y);

figure(2);

for i=1:1:3

[b1,a]=butter(2,i\*100/(fs/2));

env=abs(filter(b1,a,y));

plot(10000:12000,env(10000:12000));

hold on;

end

legend('100Hz','200Hz','300Hz');

figure(3);

for i=[2,6]

[b1,a]=butter(i,200/(fs/2));

env=abs(filter(b1,a,y));

plot(10000:12000,env(10000:12000));

hold on;

end

legend('2^n^d-order','6^t^h-order');