

De-noising the Speech Signal With FIR Filter Based on Matlab

Yanxin Wei^{1, a}, Xiujuan Fan^{1, a}

Beijing Institute of Fashion Technology, Beijing, China, 100009

^aweiyaxin4800@outlook.com, ^agxyfanxj@bift.edu.cn

Keywords: Matlab, FIR filter, window function, Kaiser window

Abstract. With the advent of the information age, signal transmission has been involving many kinds of fields. During the process of voice signal transmission, there would be inevitably doped with some noise signal, but which would not happen if the very designed filter has been used. In this paper, the window function method is used to select the fast Fourier transform interval, so as to design a FIR digital low-pass filter to achieve the doping in the speech signal in the random noise elimination, and analyze the time domain and frequency domain characteristics of voice signal before and after denoising, compare the quality difference of voice signal before and after filtering. This paper aims to verify the validity of the designed FIR digital low-pass filter's function by Matlab simulation.

1 Introduction

Various parts in modern communication systems are inseparable from the different forms of signal transmission. During the actual signal processing, signal generation, the completion of digital-to-analog conversion, and signal transmission process will be more or less affected by the impact of the external environment and then bringing the distortion. And most of the distortion is very serious, so that there would be great difficulties for signals to recover and then hide in the noise.

This problem has a huge impact on the transmission quality of the communication system. Therefore, how to remove the noise off the signal is a key point to solve.

This paper mainly discusses how to design a suitable FIR filter by Matlab software, so as to achieve better filtering effect.

2 FIR filter

2.1 The characteristics of FIR filter

FIR filter is a finite impulse response filter, also known as non-recursive filter, is the most basic component in digital signal processing system. Its biggest feature is a very strict linear phase, and can guarantee any amplitude-frequency characteristics. At the same time its unit sampling response is finite, so which can be confirmed that the FIR filter is a stable system.

Precisely because its unit impulse response is finite length, fast Fourier transform (FFT) algorithm can be used to filter signal, which can greatly improve the operation rate.

2.2 FIR filter designing method

Generally, the essence in designing process of FIR filter is that in accordance with the requirements of the frequency characteristics to achieve the frequency characteristics of indicators with the corresponding algorithm. Specifically, we use a certain approximation criterion to select each parameter $h(n)$ in the filter transfer function $H(z)$, that is to calculate the unit sampling response and the order N based on the optimal clamping criterion, so as to satisfy the required frequency characteristic's requirements. Common FIR digital filter designing methods are divided into the following three: window function method, frequency sampling method and Chebyshev approximation method.

Among these three methods, the window function method is the most widely used, and the simplest. It can be calculated through the known window function and then use Matlab software

simulation to complete the specific operation. Although using the window function method to complete the designed FIR filter characteristics comes out relatively rough indicators, which would be less practical for some of the harsh demanding design, however, in the field studied by this paper, the window function method is still the most convenient designing method.

3 window function designing method

Window function method, is also known as Fourier series method. The basic idea of the window function is to select one suitable ideal frequency-selective filter and then truncate its impulse response to obtain a linear phase and causal FIR digital filter.

Designing steps:

- (1) Given the required frequency response function $H(e^{j\omega})$.
- (2) According to the formula $H(\omega) = \sum_{n=0}^{(N-1)/2} a(n)\cos(\omega n)$ Eq.1 to solve the formula $h_d(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(e^{j\omega}) e^{j\omega n} d\omega$ Eq.2.
- (3) Upon the requirements of minimum transition bandwidth and minimum attenuation of the stopband, using the following Table 1 and Table 2 to select the shape of the window $\omega(n)$ and the size of N, which is usually determined by several trials.
- (4) Find the unit impulse response of the designed FIR filter. $h(n) = h_d(n) n = 0, 1, \dots, N - 1$ Eq.3.
- (5) By $H(e^{j\omega}) = \text{DTFT}[h(n)]$, to detect the result whether to meet the designing requirements or not, if not, re-designing is needed then.

Table1: Comparison of basic parameters of commonly used window functions

Window Function	Side Lobe Peak Range/dB	Transition Band	Minimum Attenuate in Stopband/dB _s
Rectangular Window	-13	$4\pi/N$	-21
Triangular Window	-25	$8\pi/N$	-25
Hanning Window	-31	$8\pi/N$	-44
Hamming Window	-41	$8\pi/N$	-53
Blackman Window	-57	$12\pi/N$	-74
Kaiser Window ($\alpha=7.685$)	-57	$10\pi/N$	-80

Table2 Influence of Kaiser Window Parameters on Filter Performance

β	Transition Band	Passband Ripple/dB	Minimum Attenuate in Stopband/dB
2.120	$3.00\pi/N$	± 0.27	-30
3.384	$4.46\pi/N$	± 0.0868	-40
4.538	$5.86\pi/N$	± 0.0274	-50
5.658	$7.24\pi/N$	± 0.00868	-60
6.764	$8.64\pi/N$	± 0.00275	-70
7.865	$10.0\pi/N$	± 0.000868	-80
8960	$11.4\pi/N$	± 0.000275	-90
10.056	$12.8\pi/N$	± 0.000087	-100

4 Design and Simulation Based on Matlab

In order to design a FIR filter to achieve the de-noising of the speech signal processing, following steps must be carried out:

- (1) collecting and sampling the speech signal;
- (2) adding random noise to the speech signal;

- (3) designing the parameters of FIR filter;
- (4) completing the design and simulation via Matlab;
- (5) verifying whether the simulation results meet the requirements.

The speech signal de-noising flow chart is as below:

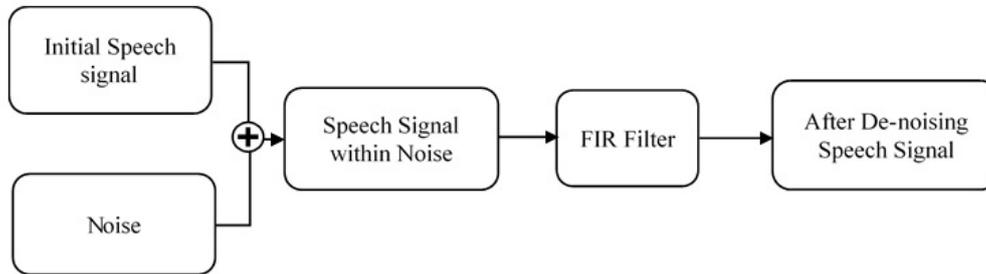


Fig.1 Speech signal de-noising flow chart

4.1 Speech signal collection and sampling

Use recording software to record some speech signal when the voice recorder is speaking in the normal status during the daily life, with the duration of 10 seconds or less, and ensuring at .wav format. Sample the speech signal by function wavread () of the Matlab software, then draw the sampling frequency and points, by using the function audioplayer () to play the original speech signal again. Then use the function fft () on the sampling point N to go through Fourier transform, make the speech signal transform from the time domain to the frequency domain.

Below are the main procedures:

```

[y, fs, bits] =wavread('C:\Users\1.wav');
audioplayer (y, fs); % Play back the speech signal
n=length(y); % The number of points of the selection transformation point
y_p=fft (y, n); % Fourier transform, transform to the frequency domain
f=fs*(0: n/2-1)/n; % The frequency of the corresponding point
  
```

After running the results shown in Figure.2:

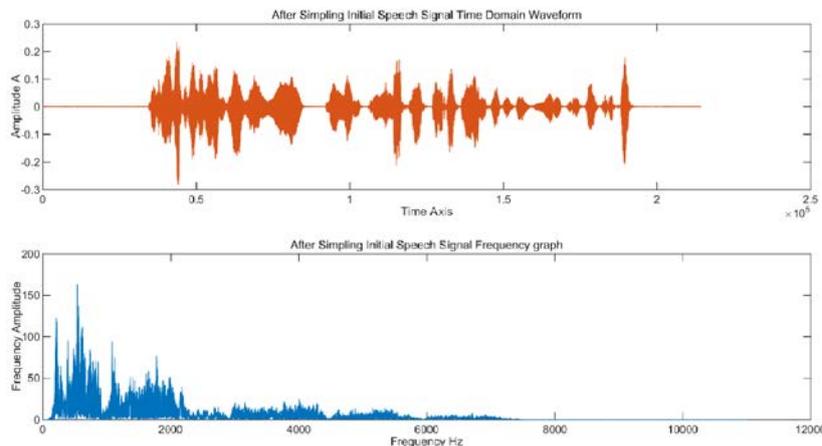


Fig.2 Time domain / frequency domain diagram of the original speech signal

By observing the time-domain diagram from the simulation results, it can be seen that the speech signal lasts about 10 seconds long and the amplitude A is not more than 0.5.

From the above frequency domain diagram of the simulation results, which can be seen that the frequency of the speech signal is mainly concentrated in the low frequency part, the maximum amplitude of the speech signal frequency does not exceed 400.

4.2 Noisy speech signal

With the statement “noise = 0.1 * randn () “in Matlab, to produce a noise signal which length equals to the original speech signal, then superimpose them to simulate and obtain a mixed speech signal including the noise and the original information.

The main procedures are as follows:

```
L=length(y); % Calculates the length of the audio signal
noise=0.1*randn(L,2); % Generates random noise signals of equal length
y_z=y+noise; % The two signals are superimposed
Through the simulation results shown in Figure.3:
```

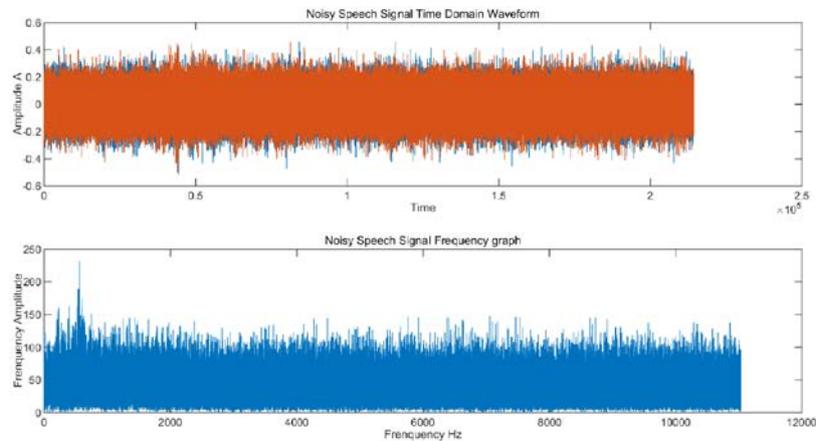


Fig.3 Time domain / frequency domain after adding noise

By observing the simulation results, the effect is very obvious when adding noise into the signal by using Matlab software. Which can be clearly seen in the time domain graph that the original speech signal almost could not be revealed, deeply buried in the noise.

4.3 Designing FIR filter

Under normal circumstances, the frequency of human speech is about 300-3400HZ. But the collected speech signal is often mixed with a lot of noise, which is very difficult to filter the noise while ensuring the quality of the speech signal. Therefore, it is particularly important to choose the indicators in the design of the filter, which relates to the final filtering effect.

The performance requirements which can be determined by connecting the actual situation are: Passband boundary is 3400HZ. Stopband boundary is 3800HZ, stopband attenuation is less than 80dB, passband ripple is less than 3dB.

Since the stop-band attenuation is less than 80 dB, Kaiser window is chosen to design the FIR low-pass filter. Kaiser window can show comprehensive reflection of the exchange relationship between the main lobe and the side lobe, and is free to choose the attenuation of the proportion between the main lobe and side lobe.

Executing the special program in Matlab as below:

```
fp=1700; fc=2000; As=100; Ap=1;
wc=2*pi*fc/fs; wp=2*pi*fp/fs;
wdel=wc-wp;
beta=0.112*(As-8.7);
N=ceil((As-8)/2.285/wdel);
wn= kaiser (N+1, beta);
ws=(wp+wc)/2/pi;
b=fir1(N, ws, wn);
```

Using Matlab software to draw the corresponding normalized and normalized frequency waveform results shown in Figure.4:

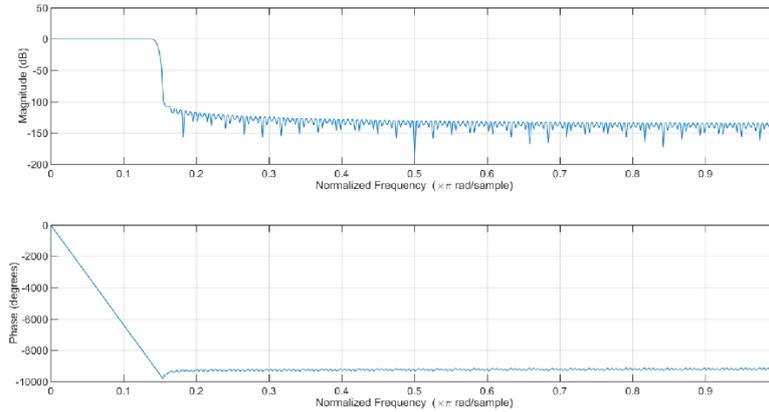


Fig.4 Normalized and normalized frequency waveforms

As the figure above tells, the frequency of the filter in the pass band within the phase shows as a straight line, which indicates that the filter is linear phase.

4.4 Simulation result verification

Finally, use the function `fir1` of Matlab to filter the voice signal.

The program is as follows: `b = fir1 (N, ws, wn);`

The results between the time domain and the spectrum pattern before and after the speech signal passing through the low-pass filter are compared in Figure.5.

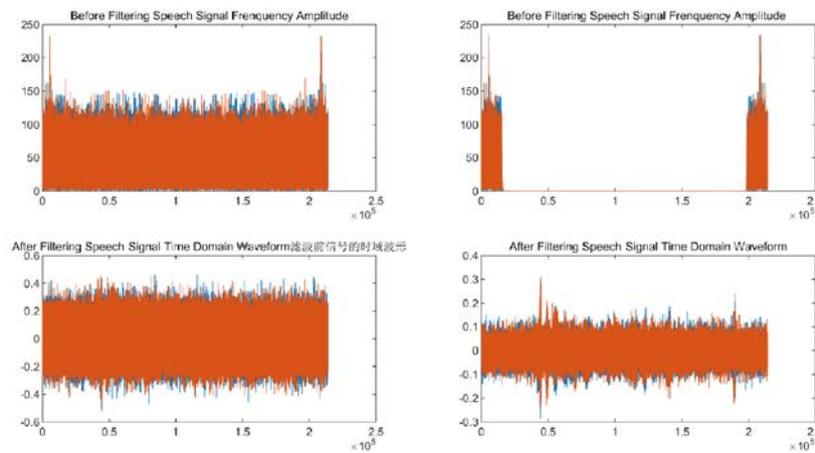


Fig.5 Kaiser window design of the low-pass filter spectrum

As can be seen from Figure.5, after low-pass filter and Kaiser window’s processing, the speech signal has been significantly optimized. Compared with the original speech signal, no matter the time domain or the frequency domain, the filtered signal’s waveform is basically the same as the original one.

Use the sound function, respectively play the original speech signal, noisy speech signal and filtered speech signal. With the comparison which can be known that the original speech signal is very clear, speech information content is integrity. After adding random noise by Matlab, the speech signal becomes obviously noisy, the original speech content becomes blurred which is not easy to identify. And after Kaiser window function and low-pass filter filtering the mixed speech signal, it obviously becomes much more clearer, the vast majority of the noise has been filtered through the filter. The original speech information has been more clearly distinguishable, which meets the requirements of the initial design of the filter.

4.5 Conclusions

By observing the simulation results and playing back the speech signal before and after de-noising, it can be concluded that the designed FIR filter can meet the requirements of the original speech signal de-noising. Therefore, it is the simplest way to denoise the speech signal

when the de-noising requirement is not too strict, and it can achieve a good de-noising effect.

5 Summary

In the real communication system, the information to be transmitted is often susceptible to noise pollution, it's very necessary to filter the noise before output the speech signal.

This paper introduces the basic characteristics of FIR filters and three basic design methods. The definition of window function and the basic design steps are also introduced in detail. And a 10-second voice signal is recorded by software, which is used to be analyzed by Matlab software in time-domain and frequency-spectrum. And then use the window function method to design the FIR low-pass filter to determine the relevant performance indicators, at the end go through the Matlab simulation.

The feasibility of the method is verified by simulation experiments, which provides the simulation basis for the speech signal filtering method selection in real life.

Reference

- [1] Peiqing Cheng. *Digital Signal Processing Tutorial* [M]. Third Edition. Beijing: Tsinghua University Press, 2007: 323-370
- [2] Digital Filtering of Speech Signal Based on Matlab [J]. *Electro acoustic Technology*. 2013.30.9: 65-72
- [3] Zhao Gang Et al. Discussion of digital filter design [J]. *Nanjing University*, 2003.2-5
- [4] Nianxi Xue. *Application of Matlab in digital signal processing* [M]. Beijing: Tsinghua University Press, 2003: 113-124.
- [5] [US] Vinay K. Ingle, John G. Proakis. *Digital signal processing (Matlab version)* [M]. Xi'an: Xi'an Jiaotong University Press, 2013: 34-55
- [6] Peiqing Cheng. *Digital filtering and fast Fourier transform* [M]. Beijing: Tsinghua University Press, 1988: 67-75
- [7] Song. J, Yang, L, Gong. K, et. all. Technical Review on Chinese Digital Terrestrial Television Broadcasting Standard and Measurements on Some Working Modes [J]. *IEEE Transactions on Broadcasting*, 2007, 53(1): 1-7.