

语音信号的 MATLAB 分析与处理

黄春燕 景妮洁 祝红梅

(杭州电子科技大学自动化学院 杭州 310018)

摘要 MATLAB 是十分强大的用于数据分析和处理的工程实用软件,利用其来进行语音信号的分析、处理和可视化十分便捷。文中介绍了在 MATLAB 环境中如何驱动声卡采集语音信号和语音信号采集后的文档处理方法,并介绍了 FFT 频谱分析原理及其显示、MATLAB 中相关函数的功能、滤波器的设计和使用。在此基础上,对实际采集的一段含噪声语音信号进行了相关分析处理,结果表明利用 MATLAB 处理语音信号十分简单、方便且易于实现。

关键词 语音信号, MATLAB, 数字滤波, 滤波器

中图分类号 TN912 **文献标识码** A

Analysis and Processing of Speech Signal Based on MATLAB

HUANG Chun-yan JING Ni-jie ZHU Hong-mei

(School of Automation, Hangzhou Dianzi University, Hangzhou 310018, China)

Abstract As an engineering software with very powerful functions of data analysis and processing, MATLAB is used for speech signal analysis, processing and visualization conveniently. This paper introduced the principle of FFT, the functions of MATLAB and the design and usage of the filter designing firstly. And then, an actual speech signal with noise was analyzed and processed by MATLAB. The result showed that MATLAB can execute analysis and processing of the speech signal simply and conveniently.

Keywords Speech signal, MATLAB, Digital filtering, Filter

语音信号处理技术的发展体现在生活中的各个方面,其重要性和现实意义也早已超乎大众的想象,早期的 BB 机到电话、大哥大到手机和智能机的这一系列发展表明语音信号处理技术正迅速发展。目前,在语音信号处理方面已经取得了非常大的进展,语音信号处理技术在计算机接口与人机交互方面应用得最为广泛^[1]。语音信号处理技术涵盖得非常广泛,包括语音识别、说话人的识别和确定、语种的识别和确认、关键词的检测和确认、语音的合成和语音编码等。目前,最富前景和挑战性的就是语音识别的技术,该技术在现实生活中已经有了具体的实际应用,其在车载导航、视频监控、网络视讯等人机交互领域有着非常广泛的应用^[2]。而其中高性能的连续语音、大词汇量语音识别系统是当前语音识别技术的前沿代表^[3];同时,语音信号的准确性识别也是亟待解决的主要任务^[4]。本文选用语音处理技术作为研究项目,是一次非同寻常的尝试。另外,采用 MATLAB 来进行一系列的研究处理。MATLAB 是一个功能强大的计算软件,在 MATLAB 中可以非常方便、快捷地处理很多技术^[5]。此次在 MATLAB 中完成相关滤波器的设计,因为 MATLAB 在参数调整上非常简便快捷,这在很大程度上减轻了研究者的工作量,对滤波器的优化也非常有利^[6]。

1 语音信号的采样和量化

首先通过电脑的录音机来录制一段为“笃学力行守正求

新 杭州电子科技大学欢迎您”的音频,再通过格式工厂软件将“.wma"文件转化成“.wav"文件,因为 MATLAB 只能处理“.wav"结尾的音频信号^[7]。通过程序 `[x,fs,bits]=wavread('bishe.wav')` 来读取语音信号时, MATLAB 默认的 $f_s=44100$ Hz。通常,人们可以听到频率范围为 20 Hz~20 kHz 的声音^[8],而通常语音频率的最高值为 3.4 kHz。一般语音信号的频率为 300 Hz~3400 Hz。由采样定理可知,抽样频率大于等于两倍信号的最高频率,信号可无失真恢复,因此取采样频率 $f_s=8$ kHz。然后利用 `sound(x,fs)` 实现对声音的重放^[9]。可以发现,这时语音信号发生了混叠,已经完全无法听到任何一个清晰发声的声音。接着使用 MATLAB 中默认的采样频率,即 $f_s=44100$ Hz 来对音频信号进行采样,再用 `sound` 函数进行声音的重放,此时与原声非常接近,因此本次研究使用 MATLAB 默认的 $f_s=44100$ Hz 作为采样频率。

我们通过以下程序得到所录制的语音信号的采样值 x ,其中 f_s 表示采样的频率,由于并没有单独设置该采样,因此采样频率为 MATLAB 默认的 $f_s=44100$ Hz,这完全符合我们之前对比后所想要设置的值。

```
[x,fs,bits]=wavread('D:\大学\毕业设计\毕业设计\13062205 黄春燕 毕设论文\语音\bishe.wav');%原始的语音信号属于列向量,此时的 x 是一个二维矩阵。
```

```
L=length(x)-1;%L 是我们采集的语音信号的数据点数。
```

本文受杭州电子科技大学课堂教学改革创新项目(KT1710)资助。

黄春燕(1994—),女,主要研究方向为信号与处理,E-mail:njjing@hdu.edu.cn;景妮洁(1985—),女,讲师,主要研究方向为信号分析与处理,E-mail:njjing@hdu.edu.cn(通信作者);祝红梅(1982—),女,高级工程师,主要研究方向为检测技术。

2 语音信号的FFT处理

对采集的语音信号进行快速傅里叶变换,以将采集的语音信号转化为频率域上的值,从而绘制出它在频率域上的波形。利用 $y=fft(x)$ 进行快速傅里叶变换^[10]时的程序如下:

```
x1=fft(x,fs);%求得所录制的语音信号的频谱
```

再利用绘图函数来绘制原始语音信号在时域上的波形和其在频域上的波形。程序可以设置为如下形式:

```
tx=(0:L-1)/fs;%可以得到原语音信号的数据点的时刻
```

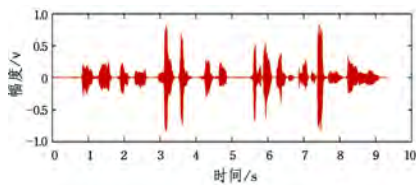
```
xf=fs/(length(x1)-1);
```

$xf1=xf*(0:length(x1)-1)$; %得到在频域波形图上的频率的刻度

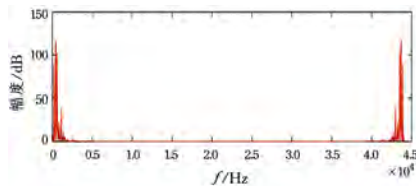
```
subplot(211);plot(tx,x);title('原信号波形图');xlabel('时间/s');ylabel('幅度/v');%在2行1列的制图中,第一张图放置原始语音信号在时域上的波形。
```

```
subplot(212);plot(xf1,abs(x1));title('原信号频谱图');xlabel('f/Hz');ylabel('幅度/dB');%在2行1列的制图中,第二张图放置原始语音信号在频率域上的波形。
```

此时,可以得到原始语音信号在时间域上的波形和在频率域上的波形^[11],如图1所示。分析语音信号的时域波形图和频域波形图可以发现,语音信号的频谱能量主要聚集在较低的频率段上,而较高频率段上的能量非常少。通过 $sound(x,fs)$ 进行语音重放发现,声音的还原度非常高,仍然可以清晰地听见“笃学力行守正求新 杭州电子科技大学欢迎您”这几个字,不存在声音混叠的现象。



(a) 原始信号波形图



(b) 原始信号频谱图

图1 原始信号波形图和频谱图

3 语音信号的加噪处理

由于本身录制的音频信号几乎是不含噪声的,因此为了之后设计滤波器的效果更明显,这里需要进行添加噪声的处理。在 MATLAB 中,可以通过程序设计给原始的语音信号添加不同的噪声,包括高斯白噪声、单频正弦噪声、多频正弦噪声以及一些其他类型的噪声^[12]。

高斯白噪声指在幅度变化上服从高斯分布,而在功率谱密度上呈现均匀分布的一种噪声。我们可以调用 WAG 函数和 $AWGN$ 函数来直接产生,也可以调用 $randn$ 来产生一个高斯分布序列。

单频正弦噪声指在幅度上呈现正弦变化的一种噪声。可以先设置好需要添加的正弦噪声,再将其与采集的语音信号叠加,即可得到添加了单频正弦噪声后的语音信号。程序如下:

```
d1=[0.8*sin(2*pi*5000*t)]';%添加的向量是一个行向量,也是一维矩阵
```

```
x1=x+d1;%叠加形成单频正弦
```

多频正弦噪声是将多个振幅相同但频率不同的呈现出正弦变化的两个单频噪声叠加到一起形成的。可以预先设置好所要添加的多个正弦噪声,再将其与采集的语音信号叠加,即可得到加了多频正弦噪声后的语音,程序如下所示:

```
d1=[0.8*sin(2*pi*5000*t)]';%添加的向量是一个行向量,也是一维矩阵
```

```
d2=[0.8*sin(2*pi*20*t)]';
```

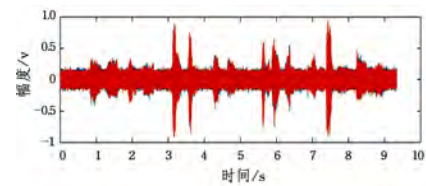
```
x2=x+d2+d1;%叠加后形成了多频正弦噪声
```

若需添加一个均匀的噪声,那么就可以采用 $randn$ 函数来生成一个高斯白噪声。程序设计如下:

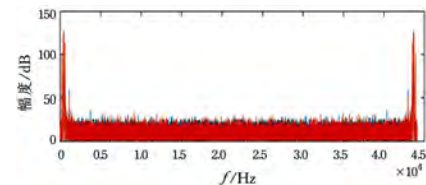
```
noise=0.05*randn(L,2);%用于生成一个同等长度的均匀随机噪声信号。
```

```
X_Z=x+noise;%将噪声信号与原始采集信号叠加,形成加噪后的语音信号。
```

用 $randn(L,2)$ 来产生一个随机矩阵,将其作为噪声加入到原始的语音信号中,用系数 0.05 对这个噪声进行一定程度的削弱,可以使得之后的滤波产生更加明显的效果。同理,绘制出添加噪声之后的信号在时间域上的波形和在频率域上的波形,如图2所示。



(a) 添加随机噪声后的信号时域波形图



(b) 添加随机噪声后的信号频谱图

图2 添加随机噪声后的时域波形图和频谱图

可以很明显地看到,加噪处理之后的时域波形和频谱图都有了一个均匀的增加,符合之前预想的结果。通过 $sound(X_Z,fs)$ 进行回放语音,可以清楚地听到“笃学力行守正求新 杭州电子科技大学欢迎您”这几个字,同时也可以听到等强度的噪声一直贯穿整个音频。接着设计数字滤波器,然后使用滤波器来对已经添加了噪声的语音信号进行滤波处理。

4 滤波器的设计

(1) FIR 滤波器的设计

首先使用 $hamming$ 窗设计 FIR 的低通滤波器,其核心程序如下:

```
wde1=ws-wp;%求过渡带宽
```

```
N=ceil(6.6*pi/wde1);%求滤波器的阶数
```

```
wn=(ws+wp)/2;%计算滤波器的长度
```

```
b=fir1(N,wn/pi);%得到滤波器的系数
```

再利用 $freqz$ 函数来绘制低通滤波器的幅度-频率上的

响应波形,如图 3 所示。

```

[h,w]=freqz(b,1,fs);%得到滤波器的频率响应
figure(1);subplot(221);plot(w/pi,abs(h));xlabel
('归一化频率(w/pi rad/sample)');
ylabel('幅度/v');title('hamming 窗低通滤波器的幅频
响应');%绘制幅频图

```

利用设计的滤波器^[13]进行滤波,并绘制出其与滤波前的对比图,使滤波的效果更加明显。利用 *plot* 画出滤波前后语音信号在时间域上的波形图和其在频率域上的波形图,从而得到如图 7 所示的低通滤波器滤波前后的波形对比。通过频率转换和更改性能参数,可以得到高通、带通和带阻滤波器的幅频响应,如图 4—图 6 所示。通过频率转换公式和更改性能参数,可以得到高通滤波器、带通滤波器和带阻滤波器滤波在时间域上的波形及其在频率域上的前后波形,如图 7—图 10 所示。

对滤波后的频谱图进行对比分析后可以发现,hamming 窗低通滤波器的滤波效果是最好的^[14]。用 *sound* 函数在 MATLAB 中对滤波后的语音信号进行重放可以发现,低通滤波器的噪声最小且最少,但也仍然可以听见噪声,与滤波前的信号相比,噪声有所减弱,录制的“笃学力行守正求新 杭州电子科技大学欢迎您”仍然可以清楚听见。

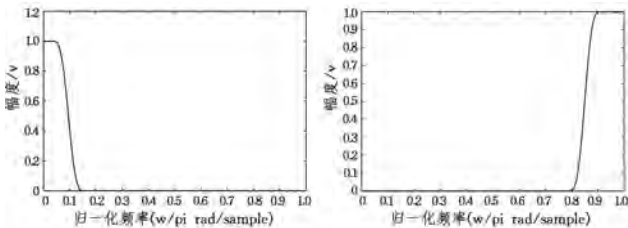


图 3 hamming 窗低通滤波器的幅频响应

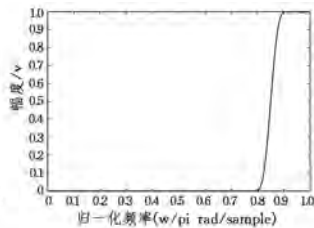


图 4 hamming 窗高通滤波器的幅频响应

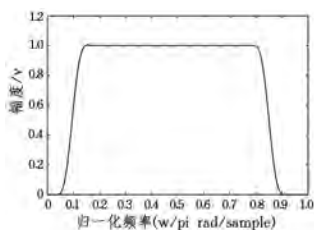


图 5 hamming 窗带通滤波器的幅频响应

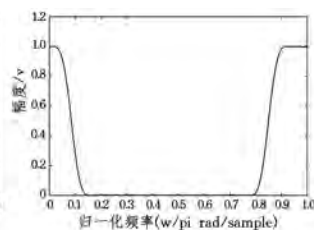


图 6 hamming 窗带阻滤波器的幅频响应

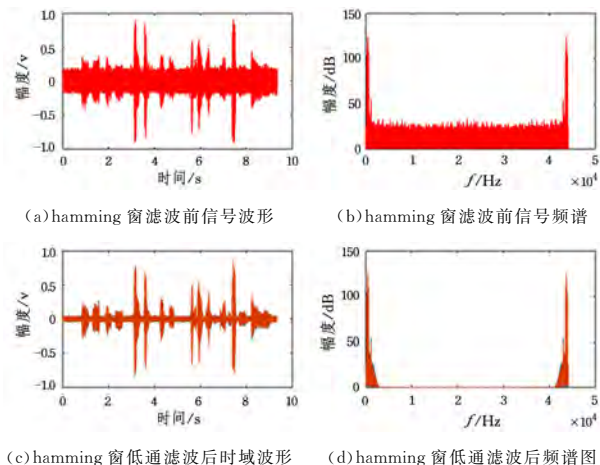


图 7 hamming 低通滤波前后波形对比

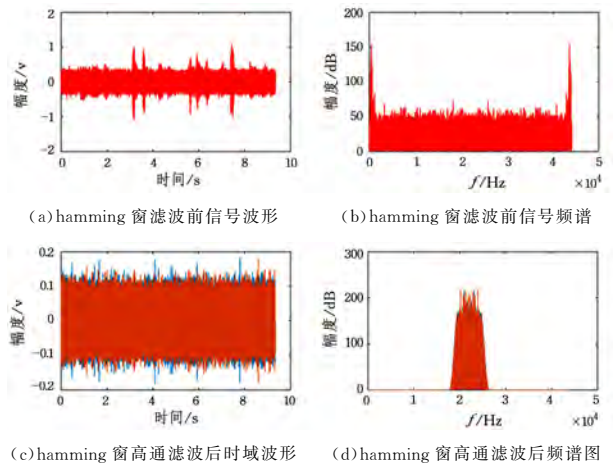


图 8 hamming 窗高通滤波前后波形对比

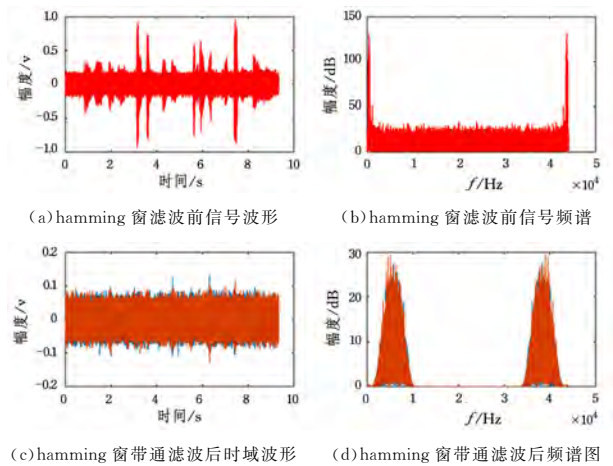


图 9 hamming 窗带通滤波前后波形对比

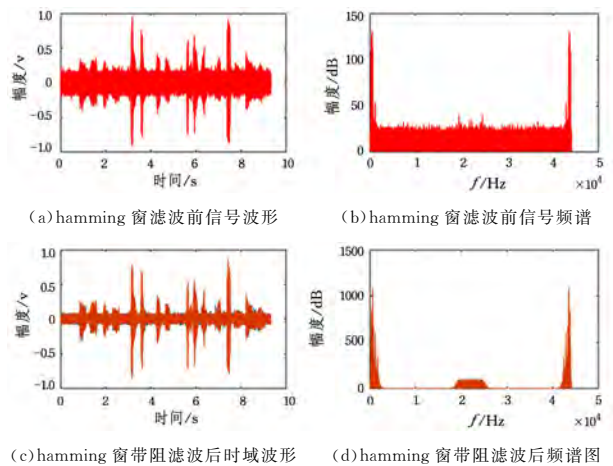


图 10 hamming 窗带阻滤波前后波形对比

通过高通滤波器滤波后可以发现,时间域上的波形和频率域上的波形都有一些变化,通过语音回放可以发现,已经很难听清声音,噪声也比较明显。通过带通和带阻滤波器滤波后可以发现,时间域上的波形和频率域上的波形都变化不大。通过语音回放可以发现,带通滤波器滤波后的语音声音似乎有些变化,与原本发出的声音相比已经不太一样。通过带阻滤波器滤波后的语音可以发现,其与原本的声音相比相差不大,但噪声依旧比较明显,基本没有减弱。

(2) IIR 滤波器设计

首先利用双线性变换法来设计椭圆低通滤波器,由之前的分析知,低通滤波器的性能指标设置为: $\omega_p = 0.045\pi$,

$\omega_s = 0.145\pi, R_p = 0.25, R_s = 50\text{ dB}$.

设计椭圆滤波器时可以采用的程序如下:

`[N,wc]=ellipord(wp,ws,rp,rs);`%用于获得滤波器的截止频率 ω_c 和阶数 N

`[b1,a1]=ellip(N,rp,rs,wc);`%双线性变换法设计 IIR 滤波器,得到滤波器系数 $b1,a1$

`yn=filter(b1,a1,X_Z);`%用设计好的滤波器对加噪的语音进行滤波,得到滤波后的 yn

`yk=fft(yn);`%进行傅里叶变换

通过 plot 绘图可以得到低通滤波器的时域波形和频谱图,如图 11 和图 12 所示。通过频谱分析可以发现,椭圆低通滤波器的效果相对比较明显,滤波后的时域波形和频谱图与未加噪前的时域波形和频谱图都非常接近。通过 sound 函数重放滤波后的语音可以发现,噪声已经比较微弱,可以清晰地听到“笃学力行守正求新 杭州电子科技大学欢迎您”这几个字。同样,通过频率转换公式和更改性能指标参数,就可以得到高通滤波器的幅频响应以及滤波前后时域波形和频谱图的变化,如图 13—图 14 所示。

通过对频率域上的波形图的分析,对高通滤波器滤波后可以看到,在时域波形图和频率域波形图上,滤波前后的变化并不明显,只有频率域上的波形在高频段上的能量衰减,这完全符合高通滤波器的特性。我们对语音信号进行回放可以发现,声音变得不太清楚了,“笃学力行守正求新 杭州电子科技大学欢迎您”这几个字也不清楚,噪声仍然很明显。

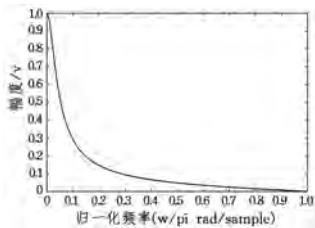


图 11 椭圆低通滤波器的幅频响应

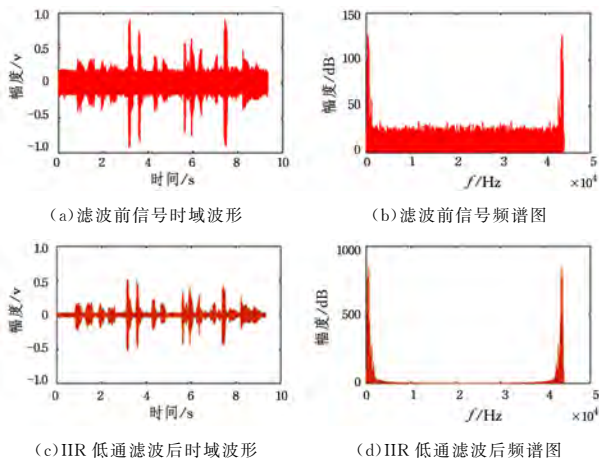


图 12 IIR 低通滤波前后波形对比

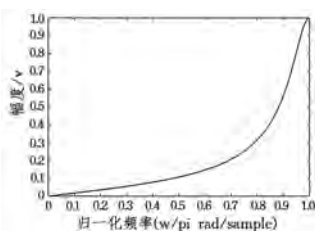


图 13 椭圆高通滤波器的幅频响应

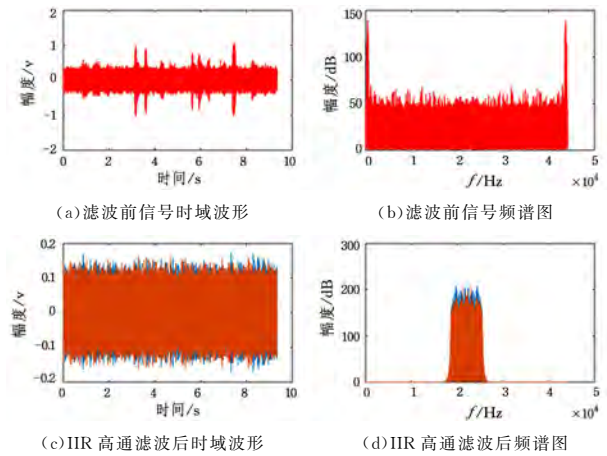


图 14 IIR 高通滤波前后波形对比

结束语 本文首先利用声卡系统完成语音的录制、读取,设置适当的采样频率对信号进行采样,并绘制出原始语音信号在时间域上的波形,利用快速傅里叶变换得到语音信号在频率域上的波形图;其次,对采集的语音信号进行加噪处理,绘制出添加噪声之后的语音信号的时域波形和频域波形图;最后,通过设计滤波器来对加噪后的语音信号进行滤波,对比其滤波前后的波形变化。在整个研究过程中, MATLAB 软件用于实现声音的回放、时域上的波形和频域上的波形显示。采样时需要设置合适的采样频率来满足采样定理,以便能恢复原始音频;再通过设置合理的性能参数,利用窗函数法和双线性变换法的设计方法,完成 FIR 数字滤波器和 IIR 数字滤波器的设计。

参考文献

- [1] TU J X, XIA Y S, ZHANG S C. A complex-valued multichannel speech enhancement learning algorithm for optimal tradeoff between noise reduction and speech distortion[J]. Neurocomputing, 2017, 267: 333-343.
- [2] 王群, 曾庆宁, 谢先明, 等. 低信噪比环境下的语音识别方法研究[J]. 声学技术, 2017, 36(1): 50-56.
- [3] 韦高梧, 冯祖勇. 基于去噪技术的 DSP 语音识别系统设计[J]. 传感器与微系统, 2017, 36(1): 108-118.
- [4] 杨臻. 语音信号准确性识别仿真研究[J]. 计算机仿真, 2017, 34(2): 285-288.
- [5] 张雪英. 数字语音处理及 MATLAB 仿真[M]. 北京: 电子工业出版社, 2016: 1-2.
- [6] 惠博. 语音识别特征提取算法的研究及实现[D]. 西安: 西北大学, 2008.
- [7] 徐靖涛, 王金根. 基于 MATLAB 的语音信号分析和处理[J]. 重庆科技学院学报(自然科学版), 2008, 10(1): 132-136.
- [8] 王彬. MATLAB 数字信号处理[M]. 北京: 机械工业出版社, 2010: 180.
- [9] 周玲. 基于 MATLAB 的语音信号数字滤波处理[J]. 安阳师范学院学报(自然科学版), 2011, 17(3): 46-49.
- [10] 张娜, 袁训峰. 含噪语音信号分析与处理[J]. 商洛学院学报, 2016, 30(6): 11-16.
- [11] 韩大伟, 熊欣. 语音信号采集和处理方法的分析[J]. 无线互联科技, 2013: 67-68.
- [12] 陈砚圃. 含噪语音信号中噪声参数的一种估计方法[J]. 西安交通大学学报, 2001, 35(10): 1096-1097.
- [13] 马慧卿, 杨风. 语音信号的滤波设计[J]. 山西电子技术, 2014(3): 7-8.
- [14] 赵培瑶, 向凤红. 基于 MATLAB 的不同数字滤波器对语音信号的去噪效果[J]. 昆明理工大学, 2015, 43(7): 717-719.