
中北大学

信息处理综合实践 说明书

学 院： 信息与通信工程学院

专 业： 电子信息工程

学 生 姓 名： 张乐平 学 号： 1905041547

课程设计题目： 语音信号的采集、加噪及滤波

起 迄 日 期： 2021 年 12 月 6 日 --- 2021 年 12 月 22 日

指 导 教 师： 郭丽娜

负 责 人： 韩建宁

2021 年 12 月 22 日

中北大学

信息处理综合实践 任务书

2021 /2022 学年 第 1 学期

学 院： 信息与通信工程学院

专 业： 电子信息工程

课程设计题目： 语音信号的采集、加噪及滤波

学 生 姓 名： 张乐平 学 号： 1905041547

学 生 姓 名： 段定泓 学 号： 1905041740

学 生 姓 名： 郭一帆 学 号： 1905040529

学 生 姓 名： 樊嘉晖 学 号： 1905040819

起 迄 日 期： 2021 年 12 月 6 日 --- 2020 年 12 月 22 日

课程设计地点： 科学楼 B 座

指 导 教 师： 郭丽娜

负 责 人： 韩建宁

下达任务书日期： 2021 年 12 月 6 日

课程 设计 任务 书

1. 设计目的:

通过课程设计,进一步巩固数字信号处理中的基本原理与方法,熟练掌握一门计算机语言,进行数字信号处理应用的开发设计,训练学生查阅设计资料和手册、程序的设计和调试等基本技能,提高学生分析、解决实际问题的能力。

2. 设计内容和要求(包括原始数据、技术参数、条件、设计要求等):

原始数据:

录制一段组员的语音文件及一段噪音信号。

技术要求:

(1)对原始语音信号和噪音信号进行频谱分析,画出两种信号的时域波形和频谱图,并对比分析其频谱特性;

(2)将语音信号和噪音信号叠加为一路信号,根据上述信号的频谱特性,自行设计一滤波器对噪声信号进行滤除,对比分析滤波前后语音信号的频谱特性,并画出滤波器的频域图。

设计要求:

课程设计的目的在于培养学生的综合素质,要求学生做到:

(1)根据题目要求查阅有关资料,确定方案,写出设计方案并进行分工;

(2)根据对选题的理解,消化查阅资料,给出相关的实现算法和理论根据;

(3)在 Matlab 环境下,实现数字语音信号的频谱分析及 FIR 滤波器的设计,给出信号频谱特性并实现滤波;

(4)组员根据分工,针对各自分工的侧重点对结果进行分析总结。

3. 设计工作任务及工作量的要求：

设计工作任务：

- (1) 了解课题背景，掌握基本技术，制定设计思路，写出设计方案；
- (2) 完成设计内容：包括算法和程序设计；实验验证及技术改进，写出设计说明书；
- (3) 设计答辩。

工作量的要求：

- (1) 通过查阅资料了解信号处理的基本理论、原理以及技术背景，理解选题意义；
- (2) 掌握课程设计要求信号处理方法的基本理论、原理和实现手段；
- (3) 完成相应的算法和实现手段；
- (4) 通过实验仿真验证算法的可行性；
- (5) 给出技术总结。

课 程 设 计 任 务 书

4. 设计成果形式及要求:

仿真结果;
课程设计说明书。

5. 工作计划及进度:

2021 年 12 月 6 日-2021 年 12 月 9 日: 明白研究任务、理解研究内容、查资料, 在教师指导下明确设计方案;

2021 年 12 月 10 日-2021 年 12 月 21 日: 仿真实现, 分析实验结果并改进算法, 完成课程设计说明书;

2020 年 12 月 22 日: 答辩。

学科部副主任审查意见:

签字: _____

年 月 日

目录

一、 总体方案设计.....	1
二、 基本原理.....	2
2.1 采样定理.....	2
2.2 采样频率.....	2
2.3 声道与声卡.....	2
2.5 时域信号的 FFT 分析.....	3
2.6 高斯白噪声.....	3
2.7 IIR 数字滤波器-巴特沃斯滤波器.....	3
2.8 FIR 数字滤波器.....	4
2.9 滤波器的性能比较.....	4
2.10 栅栏效应.....	5
三、 算法步骤.....	5
3.1 语音信号的采集与存储.....	5
3.2 语音信号.....	5
3.3 噪声信号.....	6
3.4 图形用户界面（Graphical User Interface）的设计.....	8
四、 界面设计过程及使用分析.....	10
4.1 系统各部分使用介绍.....	10
4.2 设计流程简述.....	15
五、 设计评述与收获体会.....	16
参考文献.....	17

一、总体方案设计

本课程设计语言信号输入有两种模式，第一是用麦克风采集语音信号，第二是读取本地音频，通过 FFT 进行频域变换，从而分析语音信号的时域频域。

噪声信号有高斯白噪声和高频噪声两种，通过 FFT 进行频域变换，分析噪声信号的时域频域。然后进行信号合成，对合成信号通过 FFT 进行频域变换，分析合成信号的时域频域。

接下来选用不同滤波器对合成信号进行滤波去噪，对滤波前后时域和频域的波形图进行横向与纵向对比分析。

此外该设计采用 GUI 设计用户交互界面，操作合理方便。

语音信号噪声分析与滤除的设计流程图如图 1.1 所示。

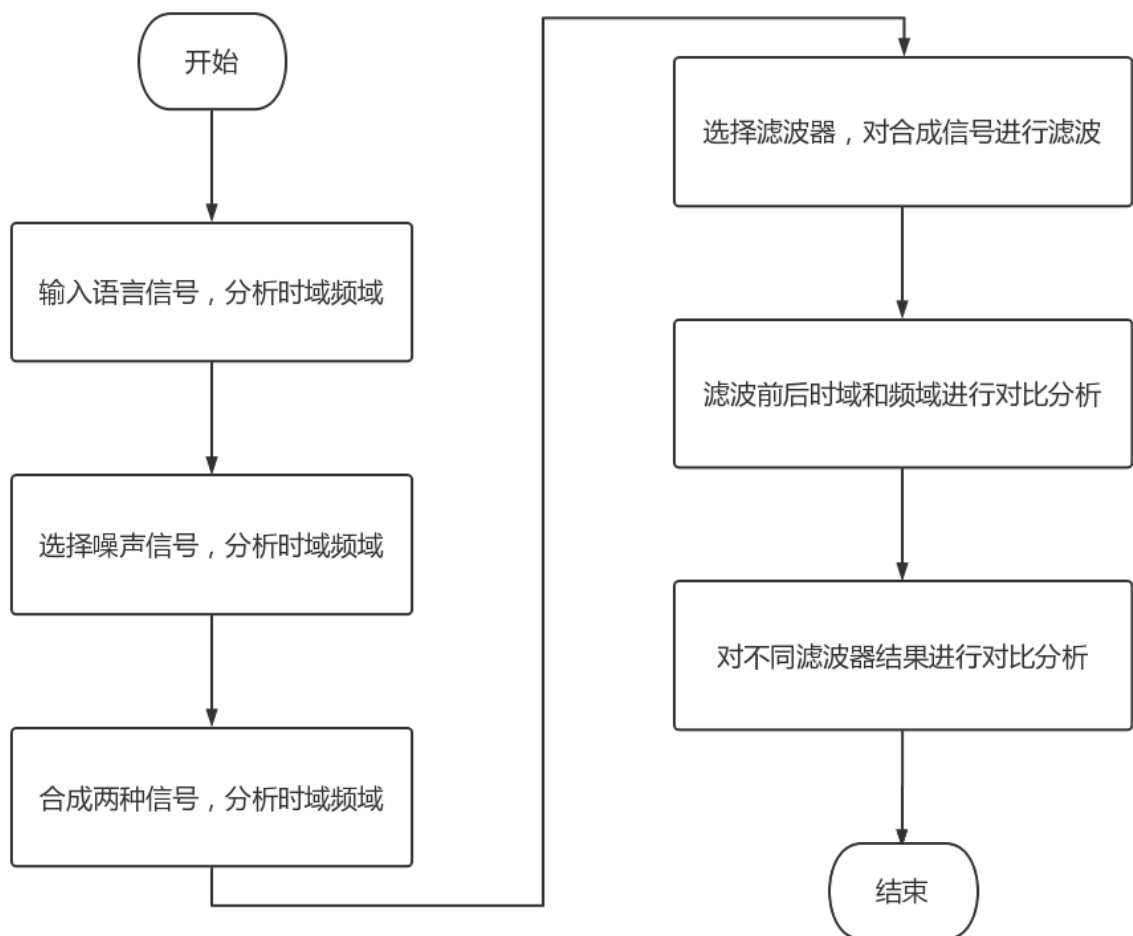


图 1.1 总体设计流程图

依据图 1.1 的具体步骤如下：

- 1.小组成员分别采用麦克风采集语音信号，作为本地处理语音信号。
- 2.准备噪声信号，分析语音信号和噪声的时域频域特点，并查阅相关资料，为设计相应的滤波器提供理论依据。
- 3.小组成员三名采用不同算法设计滤波器，一名成员通过 GUI 设计交互界面
- 4.进行噪声和语音信号的合成
- 5.对说话加噪声的语音信号进行噪声滤除，将滤波后的信号与平缓说话声的语音信号进行纵向时域频域对比分析。并对不同滤波器的滤波效果进行横向时域频域对比分析。

二、基本原理

2.1 采样定理

在进行模拟/数字信号的转换过程中,当采样频率 f_s 大于信号中最高频率 f_{\max} 的 2 倍时,则采样之后的数字信号完整的保留了原始信号中的信息,一般实际应用中保证采样频率为信号最高频率的五倍到十倍,采样定理又称奈奎斯特定理。

2.2 采样频率

采样频率,也称为采样速度或者采样率,定义了每秒从连续信号中提取并组成离散信号的采样个数,它用赫兹(Hz)来表示。

采样频率是指计算机每秒钟采集多少个声音样本,是描述声音文件的音质,音调,衡量声卡,声音文件的质量标准。采样频率越高,即采样的间隔时间越短,则在单位时间内计算机得到的声音样本数据越多,对声音波形的表示也越精确,采样频率越高声音的还原就越真实越自然。采样频率与声音频率之间有一定的关系,根据奈奎斯特理论,只有采样频率高于声音信号的最高频率两倍时,才能用数字信号表示的声音还原为原来的声音。

2.3 声道与声卡

声道是指声音在录制或播放时在不同空间位置采集或回放的相互独立的音频信号,所以声道数也就是声音录制时的音源数量或回放时相应的扬声器数量。

一般的录音信号都由两个声道,为了长度分析方便,都取了第一个声道,即取音频信号矩阵的第一行数据。

声卡也叫音频卡，是计算机多媒体系统中最基本的组成部分，也是实现声波/数字信号相互转换的一种硬件。声卡的基本功能是把来自话筒、磁带、光盘的原始声音信号加以转换，输出到耳机、扬声器、扩音机、录音机等声响设备，或通过音乐设备数字接口发出合成乐器的声音。

2.5 时域信号的 FFT 分析

计算离散傅里叶变换（DFT）的高效、快速计算方法的统称，简称 FFT，它是根据离散傅氏变换的奇、偶、虚、实等特性，对离散傅立叶变换的算法进行改进获得的。当采样序列的长度为被测信号周期的整数倍时，各次谐波的频率才会刚好与各谱线位置重合，且互不影响，如不满足这一条件，通过 FFT 得到的并非为被测信号的准确频谱。

在 MATLAB 的信号处理工具箱中函数 FFT 和 IFFT 用于快速傅立叶变换和逆变换。

2.6 高斯白噪声

高斯噪声：高斯噪声指的是它的概率密度函数服从正态分布的噪声。高斯白噪声中的高斯是指概率分布是正态函数，而白噪声是指它的二阶矩不相关，一阶矩为常数，是指先后信号在时间上的相关性。高斯白噪声是分析信道加性噪声的理想模型，通信中的主要噪声源—热噪声就属于这类噪声。

2.7 IIR 数字滤波器—巴特沃斯滤波器

IIR 数字滤波器的设计步骤分三步。

第一步按一定规则将的数字滤波器的技术指标转换为模拟低通滤波器的技术指标；

第二步根据转换后技术指标设计模拟低通滤波器的 $H_a(s)$ ；

第三步按一定的规则（抽样响应不变法和双线性变换法）实现 $H_a(s) \rightarrow H(z)$

若所设计的是高通、带通或带阻滤波器，那么需要对第一步进行改动：将高通、带通或带阻数字滤波器的技术指标先转化为低通滤波器的技术指标。

本次设计用双线性变换法设计一个巴特沃斯低通 IIR 数字滤波器，

由于给定的指标为数字滤波器的指标，则首先要转换成模拟滤波器的技术指标，这里主要是截止频率 ω_p 和 ω_s 的转换，对 α_p 和 α_s 指标不作变化。

转换关系为：

$$\Omega = \frac{2}{T} \tan\left(\frac{1}{2}\omega\right) \quad (2-7-1)$$

按照模拟低通滤波器的技术指标根据相应设计公式

$$N = \left\lceil \frac{\lg \sqrt{(10^{\delta_{st}/10} - 1)/(10^{\delta_p/10} - 1)}}{\lg(\Omega_{st}/\Omega_p)} \right\rceil \quad (2-7-2)$$

$$\Omega_c = (\Omega_{st} + \Omega_p) \div 2 \quad (2-7-3)$$

求出滤波器的阶数 N 和 $3dB$ 截止频率 Ω_c ；

根据巴特沃斯归一化低通滤波器参数表，可知归一化传输函数 $H_a(p)$ ；将 $p = \frac{s}{\Omega_c}$

代入 $H_a(p)$ 去归一，得到实际的模拟滤波器传输函数。第三步通过双线性变换法转换公

$$\text{式} \quad s = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}} \quad (2-7-4)$$

得到所要设计的 IIR 滤波器的系统函数 $H(z)$ 。

2.8 FIR 数字滤波器

使用窗函数设计 FIR 滤波器，设计先给定理想滤波器的频响 $H_d(e^{j\omega})$ ，所要求设计一个 FIR 的滤波器的频响为 $H(e^{j\omega})$ ，使 $H(e^{j\omega})$ 逼近 $H_d(e^{j\omega})$

设计过程为：设计是在时域进行的，先用傅氏反变换求出理想滤波器的单位抽样响应 $h_d(n)$ ，然后加时间窗 $w(n)$ 对 $h_d(n)$ 截断，以求得 FIR 滤波器的单位抽样响应 $h(n)$ 。

设计步骤：第一步给定频响函数 $H_d(e^{j\omega})$ 第二步求出单位抽样响应 $h_d(n)$ 第三步根据过渡带宽度和阻带最小衰减，借助窗函数基本参数表确定窗的形式及 N 的大小 第四步最后求 $h(n)$ 及 $H(e^{j\omega})$ 。

布莱克曼窗函数为二阶升余弦窗，主瓣宽，旁瓣比较低，但等效噪声带宽比汉宁窗要大一点，波动却小一点。频率识别精度最低，但幅值识别精度最高，有更好的选择性。可达到进一步抑制旁瓣的效果。

2.9 滤波器的性能比较

从性能上来说，IIR 滤波器传输函数的极点可位于单位圆内的任何地方，因此可用较低的阶数获得高的选择性，所用的存贮单元少，所以经济而效率高，但是高效率是以相位的非线性为代价的，选择性越好，则相位非线性越严重。

相反，FIR 滤波器可以得到严格的线性相位，但由于 FIR 滤波器传输函数的极点固

定在原点，所以只能用较高的阶数达到高的选择性。对于同样的滤波器设计指标，FIR 滤波器所要求的阶数可以比 IIR 滤波器高 5-10 倍，但成本较高，信号延时也较大。如果按相同的选择性和相同的线性要求，则 IIR 滤波器就必须加全通网络进行相位校正，同样要大增加滤波器的节数和复杂性。

整体来看，IIR 滤波器达到同样效果阶数少，延迟小，但是有稳定性问题，非线性相位；FIR 滤波器没有稳定性问题，线性相位，但阶数多，延迟大。

2.10 栅栏效应

序列的 N 点 DFT 是序列的频谱函数在单位圆上的 N 点等间隔抽样，也就是只限制在基频 F 的整数倍处的频谱，而不是连续的频谱函数，频谱的峰值和谷点就可能因被栅栏遮挡而观察不到，这种现象称为栅栏效应。

减小栅栏效应的一种方法是在所取时域数据的末端补一些零点，使一个周期内点数 N 增加，但是不改变原有的记录长度。

三、算法步骤

3.1 语音信号的采集与存储

3.1.1 采集

我们通过麦克风录音采集语音信号，即音频信号。麦克风其实是通过声波带动振膜一起震动来采样音频信号的。振膜在震动时会有幅度，这个幅度会随着时间的推移而变化。把这个幅度值用二维坐标图画出来就是音频模拟信号了。由于计算机不能处理模拟信号，只能处理一个一个的数据，所以我们要将这个模拟信号转换为数字信号。转换过程就是每隔一小段时间就采样一次这个时刻的幅度值。

3.1.2 存储

语音信号的存储分为三个步骤：采样、量化和编码。

采样的原理很简单，就是按照固定的频率对模拟信号的振幅进行取值。这个频率，便是采样率，单位为赫兹，表示每秒钟取得的采样的个数。我们把对声波波形幅度的数字化表示称之为“量化”。将量化后离散整数序列转化为计算机实际存储所用的二进制字节序列，即编码。

3.2 语音信号

采集语音信号，通过 MATLAB 的 `audioread` 函数，采集语音文件，可以生成语音信

号 y 与采样频率 f_s ，为了防止与之后步骤采集的语音信号 f_s 不同，采用频率变换函数 `resample` 来改变采样频率，方便之后两个不同的语音信号叠加。

程序如下：

```
[y,fs]=audioread('msqvv-e9avg.wav');
y=resample(y,48000,fs); fs=48000;    %改变采样频率为 48000
n=length(y);    %获取信号长度
y_f=fft(y,n);    %FFT 进行频域转换
f=fs*(0:n/2-1)/n;    %频率离散化点数
nt=(0:n-1)/fs;    %计算时域离散横坐标
figure(1)    plot(nt,y);    xlabel('时间 s'); ylabel('幅值 ');
title('加噪前的时域波形'); %时域波形
plot(f,abs(y_f(1:n/2)));    xlabel('频率 Hz'); ylabel('频率幅值');
title('加噪前的频谱图'); %频谱图
```

3.3 噪声信号

在本次设计中，准备了三种类型的噪声，分别是高斯白噪声、高频噪声和音频信号

3.3.1 高斯白噪声

MATLAB 提供了白噪声的函数，一个是 `Randn` 用于产生随机白噪声，通过 `sound` 回放加噪声的语音信号，可以听出来人声部分基本上被噪声覆盖。

```
zs=0.05*randn(n,1);    %函数 rand 添加随机高斯白噪声
```

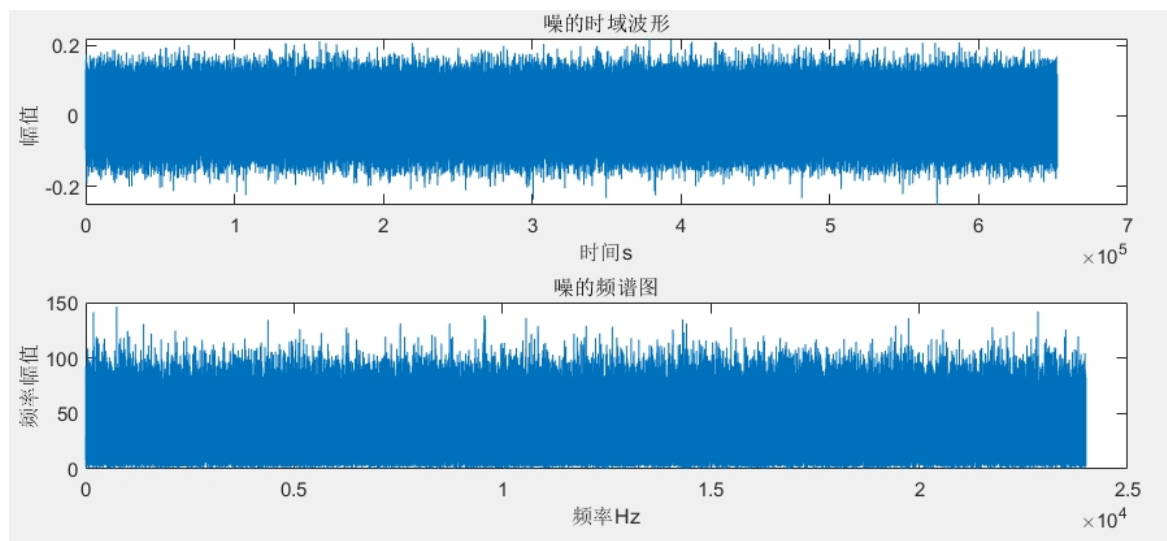


图 3.1 函数 `rand` 添加随机高斯白噪声时域频域

3.3.2 高频噪声

用 COS 函数模拟固定频率的高频噪声

```
y1=y(:,1);           %提取第一个声道
t=(0:length(y1)-1)/fs; %计算离散 T
w=5000;              %频率 w
zs=0.5*cos(2*pi*w*t); %高频噪声
```

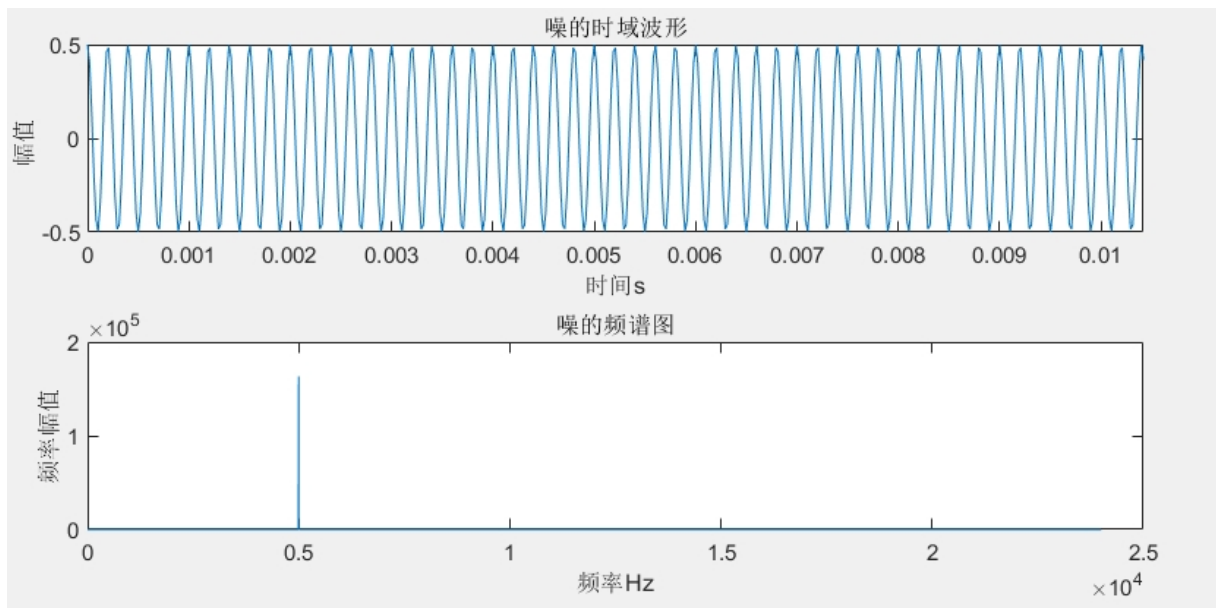


图 3.2 函数 COS 添加固定频率噪声时域频域

3.3.3 音频噪声

此外还能将两段音频信号合成，我们初步采用海豚音的音频信号。由于两段音频长度存在差别，不能直接进行合成，于是我们采用对信号进行补零的办法，来使得两段信号达到同一长度。此外补零是针对纵向序列，而音频信号采集得到的是竖向序列，我们将此进行转置补零后再转置，解决序列补零问题

```
[zs1,fs]=audioread('hai1.wav');
zs2=zs1'; %转置
zs=[zs2 zeros(2,n-length(zs2))]; %对 zs1 补零
zs=zs'; %转置
yy=y1+0.5*zs;%噪声信号的叠加
```

由于音频信号可简单看作部分白噪声与高频噪声的叠加，后续分析前两种信号的滤波效果即可。

3.4 图形用户界面（Graphical User Interface）的设计

图形用户界面（Graphical User Interface，简称 GUI，又称图形用户接口）是指采用图形方式显示的计算机操作用户界面。图形用户界面是一种人与计算机通信的界面显示格式，允许用户使用鼠标等输入设备操纵屏幕上的图标或菜单选项，以选择命令、调用文件、启动程序或执行其它一些日常任务。与通过键盘输入文本或字符命令来完成例行任务的字符界面相比，图形用户界面有许多优点。图形用户界面由窗口、下拉菜单、对话框及其相应的控制机制构成，在各种新式应用程序中都是标准化的，即相同的操作总是以同样的方式来完成，在图形用户界面，用户看到和操作的都是图形对象，应用的是计算机图形学的技术。

在本次课程设计任务中，我主要负责的部分为图形用户界面，及 GUI 的设计。之所以想到使用 GUI 作为各项信息的显示界面而非使用绘图函数所绘制的 figure 界面作为显示，是由于我们在实验过程中发现，本次课设中对语音信号进行的一系列操作，及采集加噪和滤波三个步骤可作为一套系统进行运行及显示，同时可以自行设计各种相关操作以更好地完善系统的性能，于是在讨论之后，我们一致决定使用功能更强大、性能更优越的 GUI 作为本次课设的主要实验平台。图 3.3—图 3.5 为使用脚本中 plot 函数绘制的 figure 图像，图 3.6 则为我们小组设计的 GUI 界面：

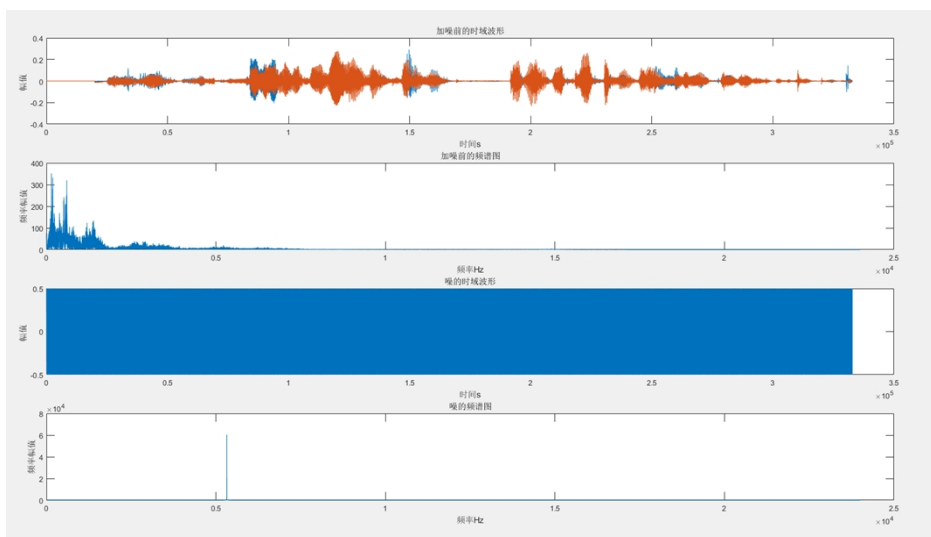


图 3.3 figure 界面 1

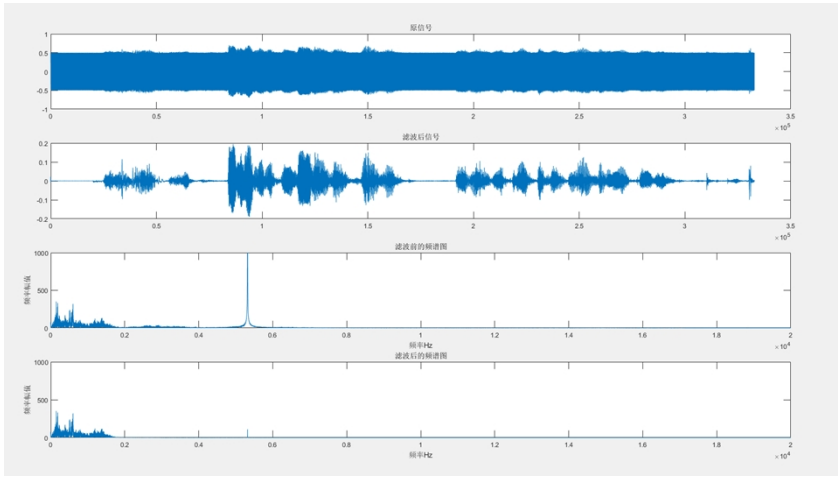


图 3.4 figure 界面 2

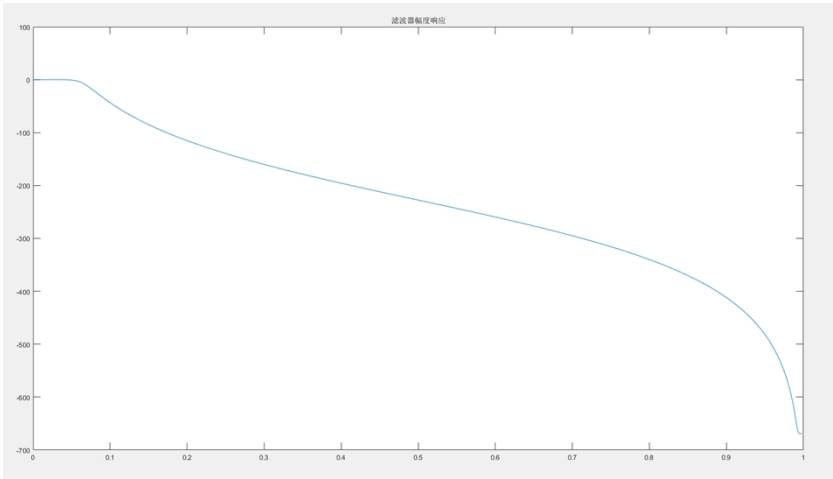


图 3.5 figure 界面 3

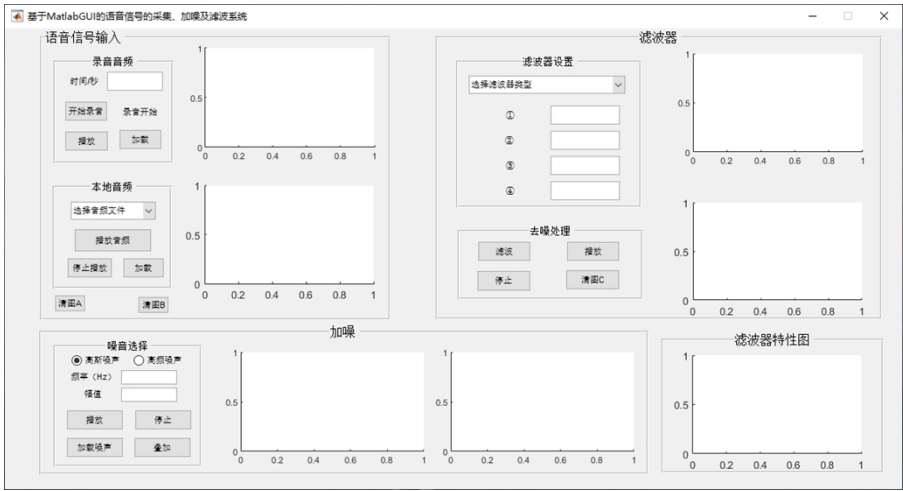


图 3.6 GUI 界面

通过两组不同界面图片的比较，可以明显看出相比于 figure 图像，GUI 界面的操作明显更加便捷，同时信息的实时显示及比较也更加合理清晰。

四、界面设计过程及使用分析

4.1 系统各部分使用介绍

语音信号输入板块是整个系统的输入部分，分为录音音频输入和本地缓存音频文件输入。在录音部分，音频信号来源为现场录音，可通过设置录音时间确定信号的长度，录音开始和录音结束均有文字提示，录音结束后即可点击“播放”按钮播放录音，同时可以点击“加载”按钮在右侧两个坐标系中分别显示时域和频域图像。本地音频部分，可通过下拉菜单选择文件，我们的系统选择了四个音频文件作为示例，可根据需要修改和选择音频文件。同样可通过点击“播放”、“停止播放”、“加载”来从听觉、视觉两方面显示音频信号。经过多次调试实验，发现此部分能较好地完成了音频信号的采集工作。

```
function pushbutton16_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton16 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global shijian R myvoice image pingpu yl fl nl fs y
fs=48000;
R = audiorecorder( 44100,16 ,1 );
set(handles.text13, 'String', '录音开始');
record(R);pause(shijian);stop(R);
myvoice = getaudiodata(R);
audiowrite('myvoice.wav',myvoice,44100);
set(handles.text13, 'String', '录音结束');
```

图 4.1 录音部分回调函数（节选）

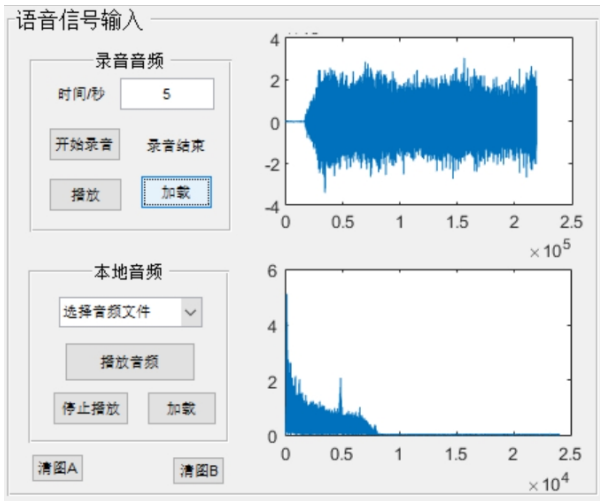


图 4.2 录音 5 秒并加载图像

图 4.2 为 GUI 界面对于 m 文件的回调函数（节选）展示，通过调用 audiorecorder

函数借助笔记本电脑自带声卡完成录音工作。图 4.3 展示了录音部分的功能，在时间框内输入 5，则录音 5 秒后显示提示字“录音结束”，即可播放和加载图像，点击左下角“清图 A”即可删除板块图像。

```
function popupmenu2_Callback(hObject, eventdata, handles)
% hObject      handle to popupmenu2 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
global y2 fs image fb pingpu n2 y
switch(get(handles.popupmenu2,'Value'))
    case 2
        [y,fs]=audioread(' 花样年华.mp3');
    case 3
        [y,fs]=audioread(' 郭CS.mp3');
    case 4
        [y,fs]=audioread(' 宁哥.wav');
    case 5
        [y,fs]=audioread(' CF.mp3');
end
```

图 4.3 本地音频文件选择部分回调函数（节选）

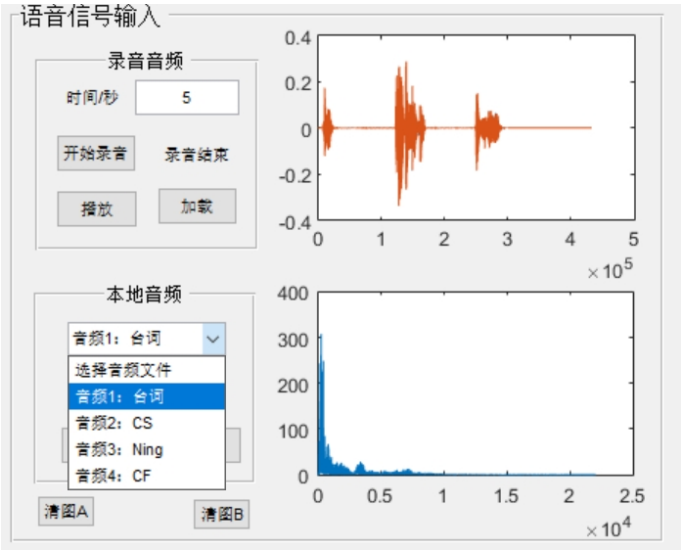


图 4.4 加载“台词”文件图像

图 4.3 为加载本地音频文件选择部分回调函数（节选），运用 switch 语句判断具体选择的是本地文件的哪一个，通过[y,fs]=audioread(' 文件名')将音频文件信息转换为数字信息存放在变量 y 中以便进行后续操作。图 4.4 展示了通过下拉菜单选择对于文件，与录音部分相同，选定文件后可通过按钮对其进行播放或绘图操作，以便直观分析。到此语音信号输入的功能基本介绍完毕，信号从此板块采集并输入进系统。

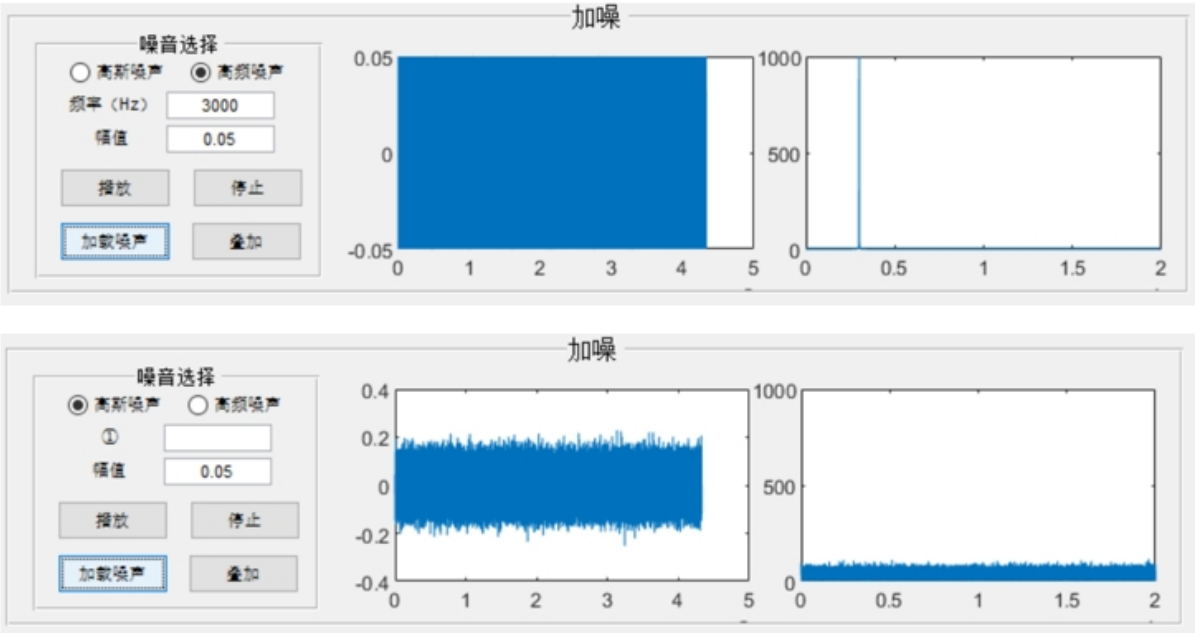


图 4.5 两种噪声源图像

加噪板块可对输入部分采集的信号进行加入噪声的处理,有两种噪声源可作为选择,一种是可调节幅值的高斯白噪声,一种是频率及幅值均可调的高频余弦信号作为高频噪声。在参数框内输入参数,选择好噪声源后,点击“加载噪声按钮后即可观察噪声信号时域、频域图像。点击“叠加”后和亮坐标系显示语音信号与噪声信号的叠加图像,“播放”和“停止”按钮用于对其进行播放以感受加噪后的信号变化。

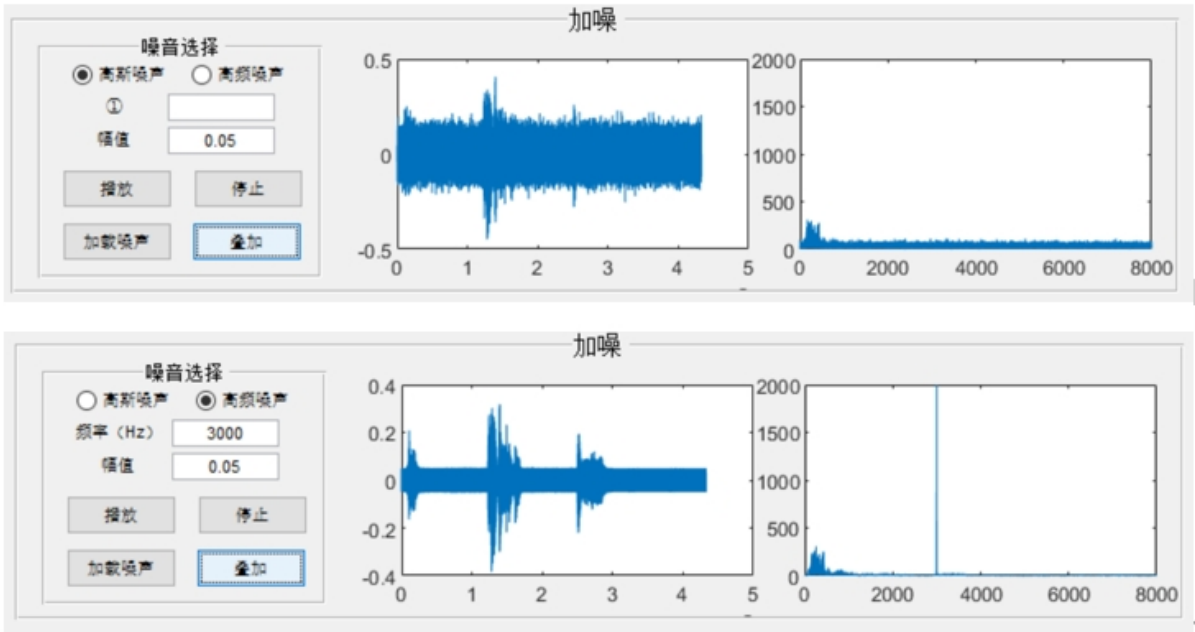


图 4.6 两种噪声源与初始语音信号叠加图像

语音信号经过采集及加噪操作后，来到滤波器滤波处理。我们针对两种不同的噪声源选择了两种不同类型的滤波器，即基于 IIR 巴特沃斯低通滤波器及基于 FIR 的布莱克曼窗带通滤波器。

可通过滤波器选择的下拉菜单选择滤波器类型，选择好类型后，下方四个参数框会显示对于滤波器的不同参数类型以调节参数构造合适的滤波器，针对巴特沃斯有通带截止频率，阻带截止频率，通带最大衰减，阻带最小衰减四个参数，布莱克曼窗带通滤波器有通带截止频率，阻带截止频率两个参数。

```
function popupmenu7_Callback(hObject, eventdata, handles)
% hObject handle to popupmenu7 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
global fs wp ws Rp Rs H W bz az lbq b
switch(get(handles.popupmenu7,'Value'))
    case 2
        Fs=fs;
        Ts=1/Fs;
        wp1=2/Ts*tan(wp*pi/2);
        ws1=2/Ts*tan(ws*pi/2); %转化为模拟滤波器截止频率
        [N,Wn]=buttord(wp1,ws1,Rp,Rs,'s'); %巴特沃斯模拟滤波器
        [Z,P,K]=buttapp(N);
        [Bap,Aap]=zp2tf(Z,P,K);
        [b,a]=impinvar(Bap,Aap,Wn);
        [bz,az]=bilinear(b,a,Fs); %用双线性变换法转换成数字滤波器
        [H,W]=freqz(bz,az);
        lbq=1;
        set(handles.text20,'String','通带截止频率');
        set(handles.text21,'String','阻带截止频率');
        set(handles.text16,'String','通带最大衰减');
        set(handles.text18,'String','阻带最小衰减');
    case 3
        M=50;
        N=M-1;
        window=blackman(M); %采用布莱克曼窗
        b=fir1(N,[wp,ws],'high',window); %带通
        [H,W]=freqz(b,1,512);
        lbq=2;
        set(handles.text20,'String','通带截止频率');
        set(handles.text21,'String','阻带截止频率');
        set(handles.text16,'String','③');
        set(handles.text18,'String','④');
end
```

图 4.7 滤波器选择部分回调函数

选择好滤波器类型及输入参数后，即可点击“滤波”进行滤波处理，滤波操作通过调用 filter 函数来实现，滤波完成后，也可通过“播放”、“停止”、“加载”等操作进行音频及图像信息展示，同时在界面右下角还会出现对应滤波器的幅频特性图，便于对滤波器更好的观测。

对高频噪声，滤波后频谱与原始信号频谱大致相同，在时域方面我们可以明显感觉滤波前后的声音有变化，声音中刺耳的声音几乎没有了。频域上也可见 3000Hz 左右基本已经没有频率分量。

对于高斯白噪声，由于噪声分布在整个频谱图内。从时域上来看还是有些轻微噪声，我们可以明显感觉滤波前后的声音有变化，声音中噪声部分大大减少了，虽然我们一定程度上恢复了原始信号的声音，但也伴随着一些噪声信号。出现噪声的可能原因我认为时我们发出的噪声信号的频率与说话频率有一部分重合造成的，这是不可避免的。

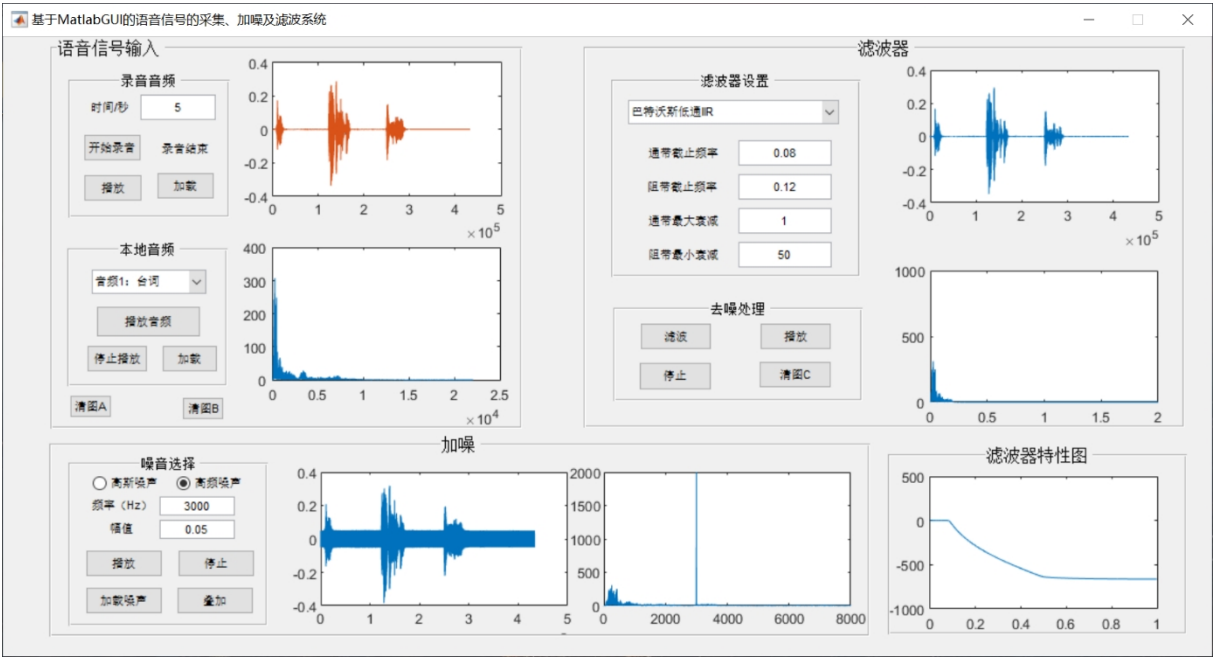


图 4.8 IIR 滤波器滤波

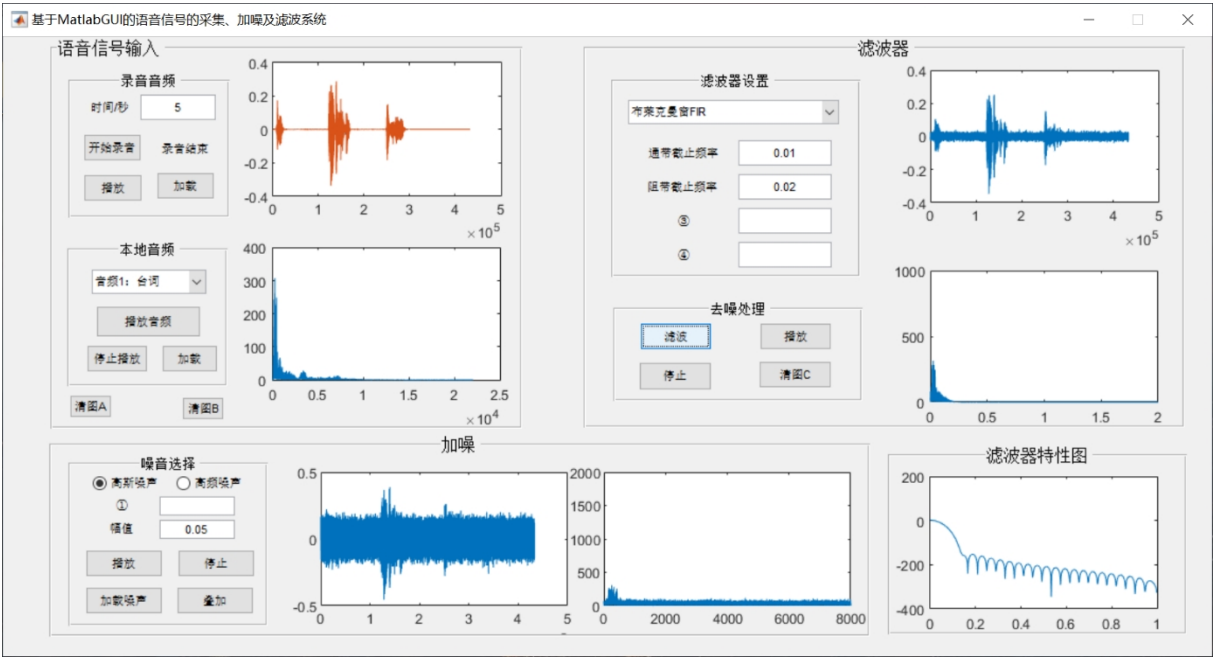


图 4.9 FIR 滤波器滤波

图 4.8 及图 4.9 展示了两两种滤波器的滤波效果，及整个系统从信号采集、加噪、滤波的横向对比，清晰直观地反映了系统的流程与功能

4.2 设计流程简述

MATLAB 的 GUI 是一个功能强大且上手容易的 UI 界面，可通过拖动右侧各种效果空间并设置对应参数即可设计出需要的界面，同时，几乎每个控制都有自己的回调函数，用于交互操作。

在设计 GUI 之前，我们先编写好了本次任务所需要的全部代码，并以 m 文件的方式运行调试实验，确定好了本系统的大致流程及所需效果控件类型和数量。多次精简代码以减少运算量之后，我开始着手设计 GUI 界面，经历了初始化、创建界面（fig）、声明全局变量，控件参数设置及控件回调函数（callback）的编写后，历经多次修改和调试，最终形成如图 4.10 和 4.11 的最终界面。

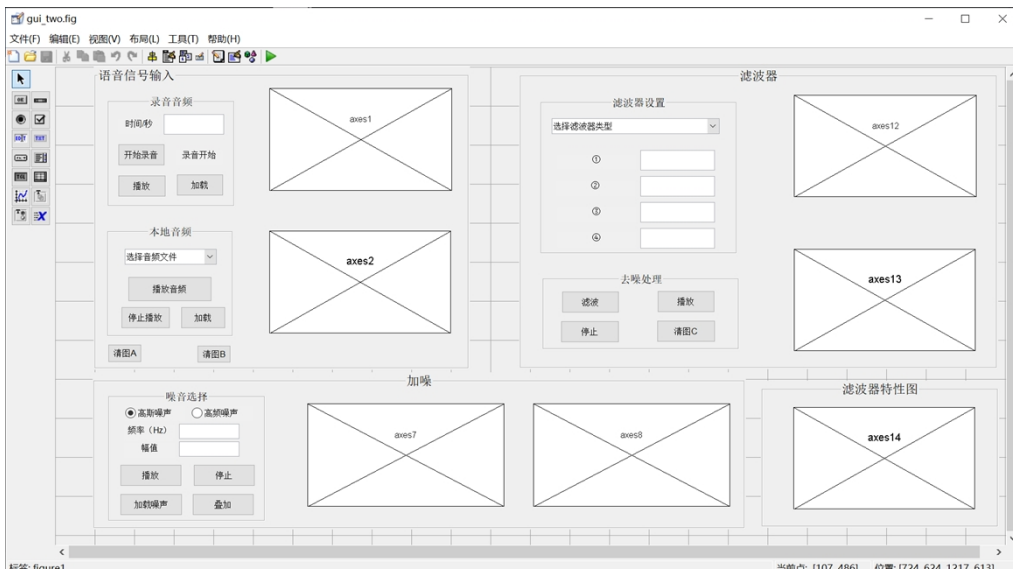


图 4.10 fig 文件界面

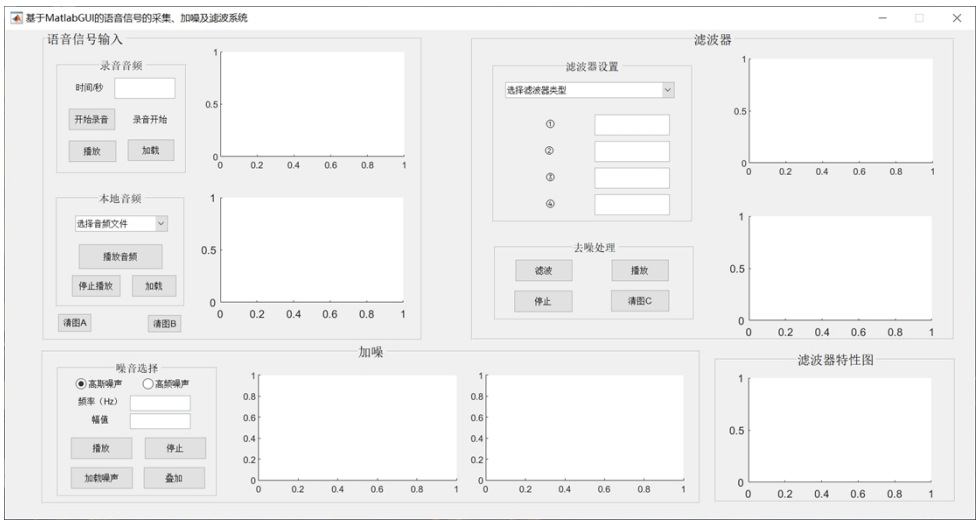


图 4.11 GUI 界面

五、设计评述与收获体会

本次课程设计的项目并不是复杂，基本都是属于信号处理、频谱分析、滤波器的基本的 MATLAB 应用。通过本次课程设计对 MATLAB 的使用有了更深入的理解。

我主要负责 GUI 交互界面的设计，设计过程中，我既要主要信号处理部分代码的编写准确无误，能完整实现课设任务要求，又要注意 GUI 界面的美观和是使用逻辑性，这其中掺杂了许多编程知识，我遇到了许多问题，最开始编译代码时总是报错，程序始终不能正确运行，后来通过查找网上的资料、浏览 HELP 文档、求助同学和助教等各种途径，最终基本上解决了所有问题，比较完美地完成了我的任务。期间，我多次修改了 fig 文件的排版，效果却始终不尽人意，我甚至一度想放弃，最后终于还是坚持下来。遇到问题、迎难而上、解决问题，我想这也是课程设计这一类实训型课程的重要意义。

将课程内的理论知识运用到实践中，我对信号进行频谱分析及滤波的设计有了更深层次的理解，同时对 MATLAB 以矩阵为基本数据模型的操作技巧更加熟练。我们没有用到非常复杂的算法，但是也能完整的完成一套滤波系统应有的功能。

任务进展期间，组员的努力与互相帮助，齐心协力让我们以比较短的时间、令人满意的完成了这次课程设计。小组成员之间的相互讨论与帮助起到了很重要的作用，我们不仅能够按时完成课程设计，也从中对本课程有了更深的认识与理解，同时也注意到自己在学习方法中的不足，并会在日后加以改正。

我们一致认为本次课程设计不足的地方是我们添加高斯白噪声进行加噪处理时，发现白噪声的频谱是布满整个频域范围内的，用低通滤波器滤出来后仍然是噪声，一般不好直接滤除。而两段语音信号的合成后，由于存在许多相同的频率分量，导致滤波效果也有限，查阅资料了解到该情况或许需要类似小波变换去噪等办法来解决，由于时间有限，未能进一步深入。

同时，但由于知识能力有限，我们所设计的信号输入、加载、滤波效果都不够完善，很多可以拓展的功能没有很好地做出来，做好的部分也有不少瑕疵。但是总体来说，不管是最后的设计出的结果还是设计过程中的一些收获感悟，都让我觉得弥足珍贵，我对信号处理有了真的认识，也从心底里萌生了深入学习的兴趣，希望我在今后的学习中，能充分吸取经验教训，在信号处理的学识和实践能力上有更大的提高。

参考文献

- [1]陈友兴等.数字信号处理原理及应用 [M] .北京：电子工业出版社，2019.
- [2]刘波，文忠，曾涯.MATLAB 信号处理 [M].电子工业出版社 2006:29-33
- [3]高西全，丁玉美.数字信号处理——原理实现及应用(第二版) [M].电子工业出版社 2010:178-192
- [4]从玉良.数字信号处理原理及 matlab 实现 [M].电子工业出版社 2005 年 7 月:153-162