目 录

[1. 设计课题 1](#_Toc114511777)

[2. audiorecorder对象及相关函数的介绍 1](#_Toc114511778)

[2.1audiorecorder对象的初始化及属性 1](#_Toc114511779)

[2.2audiorecorder对象的方法及相关函数 2](#_Toc114511780)

[3. 音频的录制与播放 3](#_Toc114511781)

[3.1设计思路 3](#_Toc114511782)

[3.2实现代码 4](#_Toc114511783)

[3.3运行结果 4](#_Toc114511784)

[3.4相关问题 4](#_Toc114511785)

[4. 音频的波形和频谱 5](#_Toc114511786)

[4.1设计思路 5](#_Toc114511787)

[4.2实现代码 5](#_Toc114511788)

[4.3运行结果 6](#_Toc114511789)

[4.4相关问题 7](#_Toc114511790)

[5. 音频的合成 7](#_Toc114511791)

[5.1设计思路 8](#_Toc114511792)

[5.2实现代码 8](#_Toc114511793)

[5.3运行结果 8](#_Toc114511794)

[5.4相关问题 9](#_Toc114511795)

[6. 音频的男女声分离 10](#_Toc114511796)

[6.1设计思路 10](#_Toc114511797)

[6.2实现代码 11](#_Toc114511798)

[6.3运行结果 11](#_Toc114511799)

[6.4相关问题 11](#_Toc114511800)

[7. GUI用户界面 12](#_Toc114511801)

[7.1设计思路 12](#_Toc114511802)

[7.2实现代码 12](#_Toc114511803)

[7.3运行结果 13](#_Toc114511804)

[7.4相关问题 14](#_Toc114511805)

[8. 工作汇总 15](#_Toc114511806)

[9. 体会 16](#_Toc114511807)

[10. 参考资源 16](#_Toc114511808)

1. 设计课题

语音信号处理

在Matlab中，我们可以使用audiorecorder命令来记录语音信号。请使用help指令来研究如何使用它。同时，我们可以使用play命令来播放语音信号。

1. 请录制持续3秒的语音信号，并播放语音信号。
2. 使用不同的采样频率记录持续3秒的语音信号，请草拟其波形和光谱。请找到可以保持语音信号不失真的最低采样频率。
3. 请分别录下男生和女生的讲话。组合两个演讲并设计一个过滤器来分隔这两个演讲。将分离的语音与原始语音进行比较，以测试分离效果。
4. audiorecorder对象及相关函数的介绍

2.1audiorecorder对象的初始化及属性

创建一个audiorecorder对象的方法如下。

recorder = audiorecorder(Fs,nBits,NumChannels,ID)

各项参数的具体含义如下。

--Fs -采样频率

--nBits -采样位数

采样位数,指定为8、16或24。

--NumChannels - 通道数

通道数,指定为1(单声道)或2(立体声)。

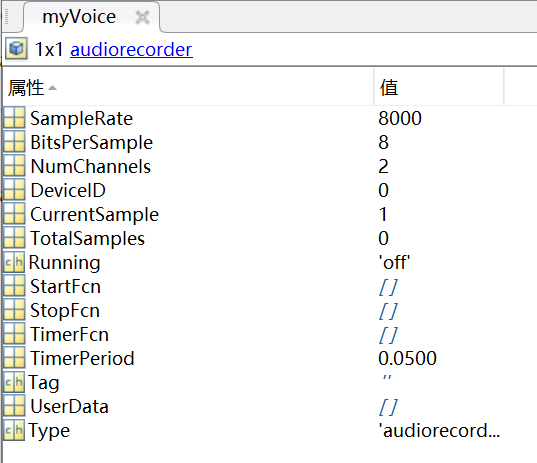
--ID -设备标识符

设备标识符，指定为整数。

其中，采样频率是较为重要的一个参数。采样频率，单位为赫兹，指定为数值标量。

采样率的有效值取决于MATLAB允许的采样率以及系统上的具体音频硬件。MATLAB的硬性限制为1000Hz <= Fs <= 384000 Hz，但还存在其他基于硬件的限制。大多数声卡支持的典型值有8000、11025、 22050、 44100、 48000 和96000。

audiorecorder对象的属性如下图所示。



2.2audiorecorder对象的方法及相关函数

audiorecorder对象的方法如下面的表格所示。

|  |  |
| --- | --- |
| get | 查询 audiorecorder 对象的属性值 |
| getaudiodata | 将录制的音频信号存储在数值数组中 |
| getplayer | 创建关联的 audioplayer 对象 |
| isrecording | 确定录制是否正在进行 |
| pause | 暂停播放或录制 |
| play | 从 audiorecorder 对象播放音频 |
| record | 将音频录制到 audiorecorder 对象中 |
| recordblocking | 将音频录制到 audiorecorder 对象中，录制完成前保持控制权 |
| resume | 从暂停状态继续播放或录制 |
| set | 设置 audiorecorder 对象的属性值 |
| stop | 停止播放或录制 |

其中getaudiodata、play、record、recordblocking将会在后面的程序中用到。使用方法如下，其中myVoice是一个audiorecorder对象，T是录制时间，单位为秒。

record(myVoice, T);

play(myVoice);

y = getaudiodata(myVoice);

其他要用到的函数如下。

保存录制内容

audiowrite(filename,y,Fs);

读取数据

[y,Fs] = audioread(filename);

收听音频

sound(y,Fs);

1. 音频的录制与播放

本部分对应设计课题中的第一小问。

3.1设计思路

首先创建一个audiorecorder对象，根据传入的参数确定采样频率Fs和录制时间T。使用recordblocking函数录制音频后，使用play方法即可播放音频。值得注意的是，为了方便后续实验的进行，应当对音频进行保存，因此还需要传入一个参数确定保存的文件名filename，filename统一以.wav为后缀。使用getaudiodata函数从audiorecorder对象中获取音频的数值矩阵，并使用audiowrite保存录制内容。

3.2实现代码

function myVoice=audio(filename,Fs,T)

myVoice = audiorecorder(Fs,8,2,0);

myVoice.StartFcn = 'disp(''Start speaking.'')';

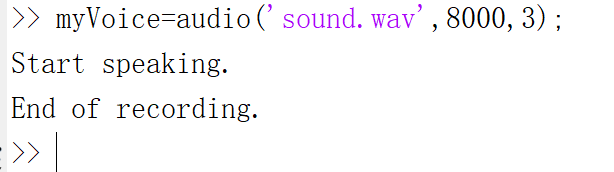
myVoice.StopFcn = 'disp(''End of recording.'')';

recordblocking(myVoice, T);

y = getaudiodata(myVoice);

audiowrite(filename,y,Fs);

play(myVoice);

3.3运行结果

3.4相关问题

1. StartFcn和StopFcn

StartFcn和StopFcn是audiorecorder对象的属性，matlab能够将这两个字符串转化为可执行语句，其中StartFcn对应的语句在录制开始时会被执行，而StopFcn对应的语句在录制结束时会被执行。显然，这两个属性对于录制本身并无作用，但是却能极大地提升人机交互的体验，方便音频的录制。

1. record和recordblocking

record和recordblocking都用于录制音频，是audiorecorder对象的方法，似乎没有太大区别。然而，record不能用于函数文件的实现代码中，原因在于record不能在录制完成前保持控制权，而实现代码的下一行代码要获取录制音频的数值矩阵，此时录制尚未结束，程序自然会报错，而recordblocking则可以解决这一问题。因此，在函数文件中应慎用record，避免在录制结束前对录制数据进行操作。当然，在命令行窗口，record可以放心使用，在录制结束前不键入新命令即可。

1. 音频的波形和频谱

本部分对应设计课题中的第二小问。

波形图：将声波转换为电信号保存起来就变成了波形图。通常我们一般讲的原始波形图都是指录制出来的波形图。波形包含了多因素信息，例如：频率、强度、音质。

光谱是复色光经过色散系统（如棱镜、光栅）分光后，被色散开的单色光按波长（或频率）大小而依次排列的图案，全称为光学频谱。这里认为音频的光谱即为其频谱，由于相频图在本实验中用处不大，仅考虑绘制幅频图。

4.1设计思路

我们考虑绘制一个音频文件的波形和频谱。首先使用audioread函数读取文件filename，获取数值矩阵x和采样频率Fs。x是一个N\*2矩阵，有两列数据的原因在于录制的音频有左右两个声道，但此次处理音频我们只需要一个声道的信息，不妨选择左声道，将x化为一个N\*1行向量。

有了向量x，我们实际上已经获得了波形的幅值，但要绘制出波形图，我们还需要知道x中每个点所对应的真实时间。已经知道抽样频率Fs，我们自然可以得到对应的真实时间t = (0:N-1)/Fs。

接着是幅频图，使用fft函数对x进行快速傅里叶变换，得到的行向量X即为幅频曲线的幅值。同样，我们需要确定X中每个点所对应的频率值。首先求出分辨率df = Fs/N，之后进行频域映射，得到对应的频率f = (0:N-1)\*df。

4.2实现代码

function timefre(filename)

[x,Fs]=audioread(filename);

x = x(:,1);

x = x';

N = length(x);

t = (0:N-1)/Fs;

X = fft(x)/N\*2;

df = Fs/N;

f = (0:N-1)\*df;

subplot(2,1,1);

plot(t,x,'g');

xlabel('Time/s');ylabel('Amplitude');

title('信号的波形');

subplot(2,1,2);

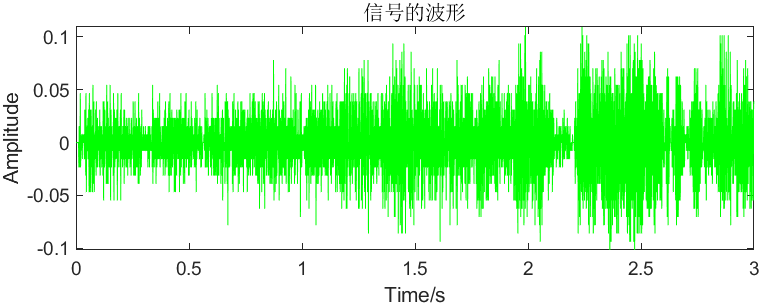
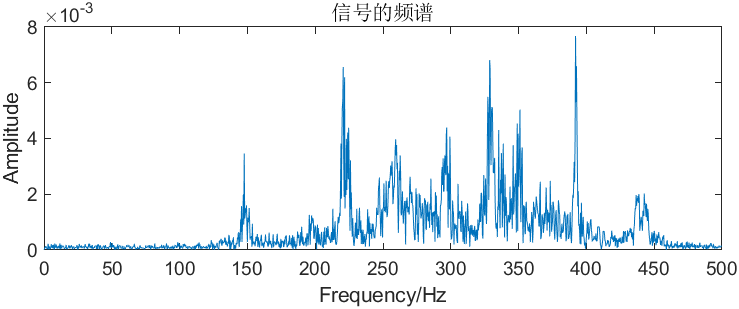
plot(f(1:round(N/2)),abs(X(1:round(N/2))));

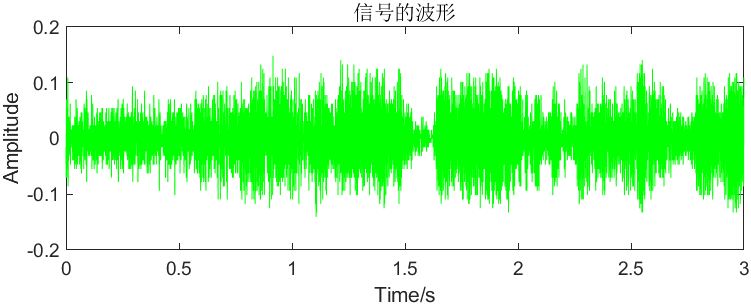
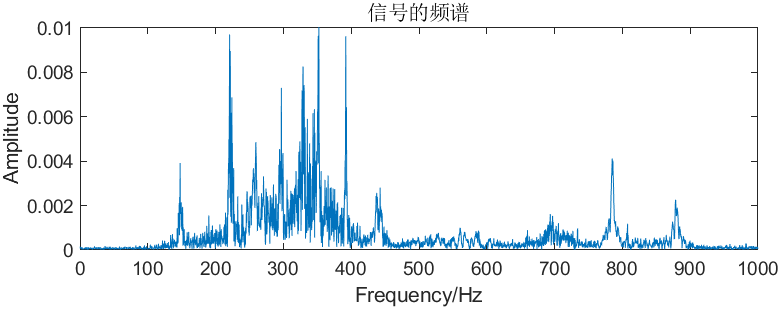
xlabel('Frequency/Hz');ylabel('Amplitude');

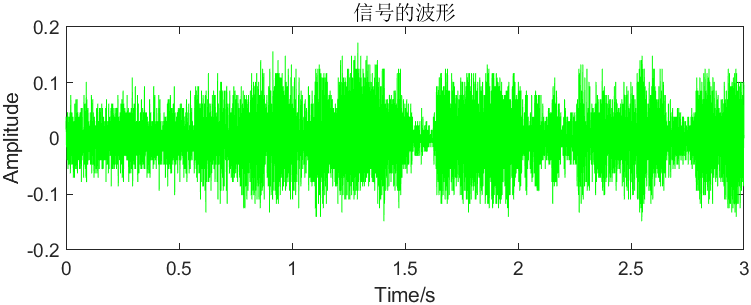
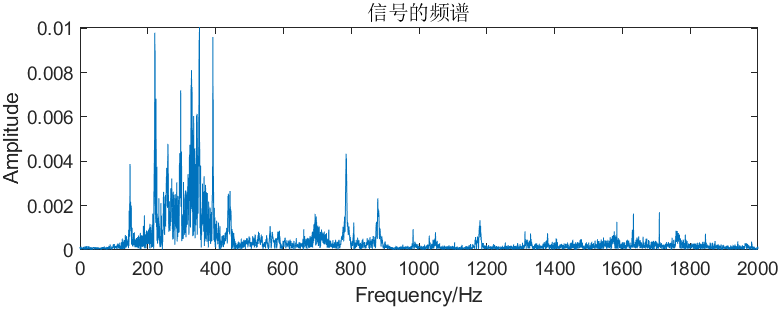
title('信号的频谱');

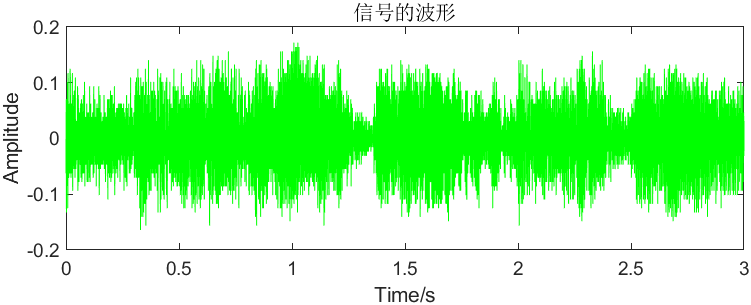
4.3运行结果

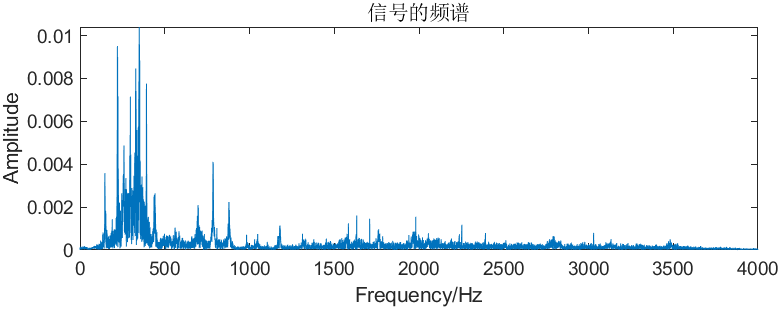
展示部分音频文件的波形图和幅频图。以下音频文件录制的音频内容相同，仅采样率不同。对应的音频文件将在资源汇总的音频资源中给出。

采样频率1000HZ（音频文件’1000fs.wav’）

采样频率2000HZ（音频文件’2000fs.wav’）

采样频率4000HZ（音频文件’4000fs.wav’）

采样频率8000HZ（音频文件’8000fs.wav’）



4.4相关问题

1. 为什么FFT后幅值要除以N/2？

DFT的频谱是用谱密度定义的，即它的幅值表示的是单位带宽的幅值。N个离散点的实数DFT将产生N/2+1个频率点。N点实数DFT以后，频谱带宽是N/2，每个频率点占的带宽是2/N，所以每个频率的实际幅值需要用DFT后的幅值乘以2/N。

1. 为什么只要半谱？

原始信号为实信号，容易证明，其幅频曲线为偶函数，记其频带宽度为到。

抽样信号的频谱可由原信号频谱与周期（频率为）脉冲信号的频谱卷积得到，将周期脉冲信号化为傅里叶级数的虚指数形式，其系数为常数，则周期脉冲信号的频谱曲线为周期脉冲。由此可知，抽样信号的幅频图可由原信号的幅频曲线经无穷次大小为的频移得到。

x经fft后得到的X对应的频域范围是。由前面的分析不难得知，当时，抽样信号的幅频图中，和的图像一致，内的图像关于左右对称。实际上，时，也不难证明图像是左右对称。因此，绘制抽样信号的幅频图只需要半谱图即可。

1. 音频的合成

本部分对应设计课题的第三小问的组合部分。

5.1设计思路

要组合两个音频文件filename1和filename2，首先要使用audioread函数分别读取对应的采样频率Fs1、Fs2和数值矩阵y1、y2。

接着要统一采样频率为Fs，这里Fs为两个采样频率中的较大值，并对采样频率较小的数值矩阵进行修改（重采样或插值等），使其采样频率变为Fs。之后将两个数值矩阵相加即可。

值得注意的是，即使统一了采样频率，由于录制时间可能不同，两个矩阵的行数可能并不相等，不能直接进行矩阵的加法。因此，应先做判断，再将两个矩阵叠加在一起。

最后将相加得到的数值矩阵y和对应的采样频率Fs通过audiowrite函数写入音频文件synfilename即可得到组合的音频。

5.2实现代码

function signalsyn(synfilename,filename1,filename2)

[y1,Fs1] = audioread(filename1);

[y2,Fs2] = audioread(filename2);

if Fs1<Fs2

y1=resample(y1,Fs2,Fs1);

elseif Fs1>Fs2

y2=resample(y2,Fs1,Fs2);

end

len1=length(y1);

len2=length(y2);

Fs=max(Fs1,Fs2);

if len1<len2

y=y2;

y(1:len1,:)=y(1:len1,:)+y1;

else

y=y1;

y(1:len2,:)=y(1:len2,:)+y2;

end

audiowrite(synfilename,y,Fs);

5.3运行结果

以音频文件'source5.wav'和'source6.wav'为例，并将其组合的音频文件命名为'syn5+6.wav'。对应的音频文件将在资源汇总的音频资源中给出。

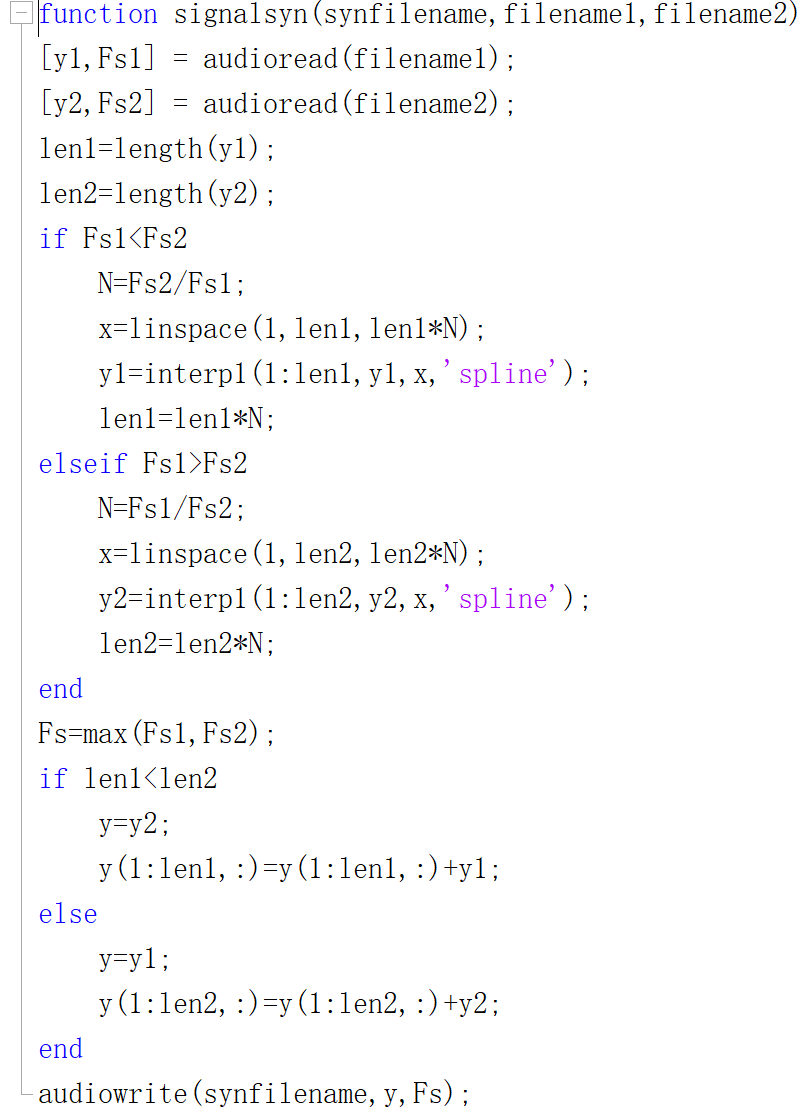
命令语句为如下。

signalsyn('syn5+6.wav','source5.wav','source6.wav')。

5.4相关问题

1. 采样率不同的音频如何组合？

音频组合可以看作是其波形的叠加，最简便的做法是将两个数值矩阵直接叠加起来。然而，不同采样率的音频，其数值矩阵同一行所对应的真实时间并不相同，可能出现的情况是A矩阵第2、3行的真实时间与B矩阵的3、5行相同。如果将两个采样率不同的数值矩阵直接相加，其效果显然会十分不理想。因此，需要在采样率较小的矩阵中添加行数（如A的2、3行之间），来统一两者的采样率。

增大一个数值矩阵对应的采样率，一种方法是插值法，我一开始采用的正是此方法。采用三次样条插值法的代码如下。

之后，我发现了重采样函数resample，它可以将数值矩阵由原始采样率修改为指定的采样率。基于此函数，我对代码进行修改，最终完成实验代码。其具体用法如下。

y=resample(x,p,q)

其中，x是待重采样的时间序列，p是重采样之后目标频率，q是待重采样的时间序列频率。

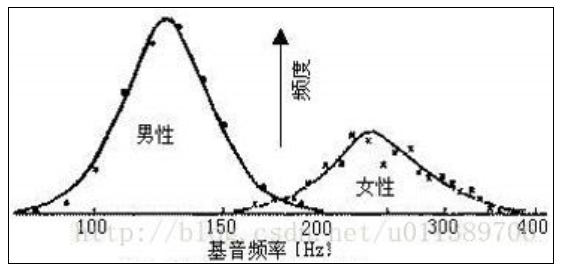
1. 响度相差太大的音频如何组合？

当两个音频的响度相差太大时，直接进行音频组合的效果不佳，很容易出现只能听到一个音频内容的情况。因此，对于响度相差较大的两个音频，在进行组合前，应当增大其中响度较小的音频，较为简单的方法是，将其对应的数值矩阵乘上一个系数。

1. 音频的男女声分离

本部分对应设计课题的第三小问的分隔部分。

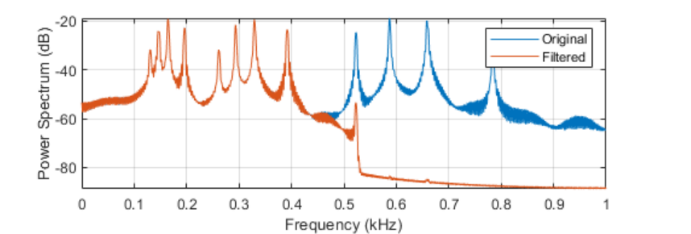
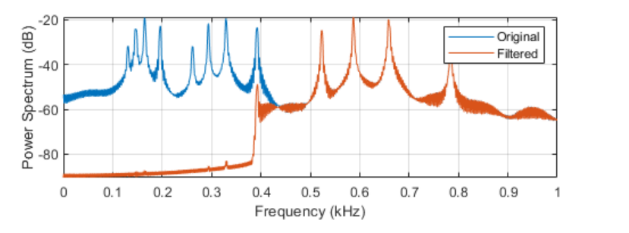
6.1设计思路

我主要从频域的区别来分隔出男女声。

一般女声的音调比男声的音调高，这种不同主要是反应在基音频率上。

人类基音的范围约为70～350Hz左右，由于生理结构的不同，男性与女性的声音呈现出不同的听觉特征，男声的基音频率大都在100—200HZ之间，而女声则在200—350HZ之间。

为从混合声音中将男生声音分离出来，我们设计低通滤波器，截止频率高于男生频率，低于女生频率，即可从混合声音信号中将男生声音提取出来。同理，要从混合声音中将女生声音分离出来，设计高通滤波器，截止频率高于男生频率，低于女生频率即可。

Matlab提供lowpass函数和highpass函数分别用于低通滤波和高通滤波。滤波前后功率谱与频率图的关系如图所示。

lowpass highpass

lowpass的使用方法如下。

y = lowpass(x,fpass,fs)

其中，x已经以fs赫兹的速率采样，fpass是滤波器的通带频率，以赫兹为单位。highpasss的使用方法与lowpass相似。

6.2实现代码

function myfilter(filename,filename1,filename2,lowf,highf)

[y,Fs] = audioread(filename);

if lowf>150

lowf=150;

end

if highf<200

highf=200;

end

y=lowpass(y,2000,Fs);

y1=lowpass(y,lowf,Fs);

y2=highpass(y,highf,Fs);

audiowrite(filename1,y1,Fs);

audiowrite(filename2,y2,Fs);

6.3运行结果

以音频文件'syn5+6.wav'为例，并将其分离的音频文件命名为'sep5.wav'和'sep6.wav'。对应的音频文件将在资源汇总的音频资源中给出。

命令语句为如下。

myfilter ('syn5+6.wav','sep5.wav','sep6.wav',80,400)。

6.4相关问题

1. 分离效果评估

分离效果一般，主要原因如下。

首先便是男生、女生声音频率范围较大，无法具体确定频率的分界线，因此设计频率时可能会存在不合理，不恰当的现象。

另外是男生、女生发声频率并非完全独立的，频率之间存在交叠，导致无法找到特定的频率完美的完成声音信号分离的任务。

想要提高分离效果，除了利用信号的频域信息外，还应考虑其他的方法。

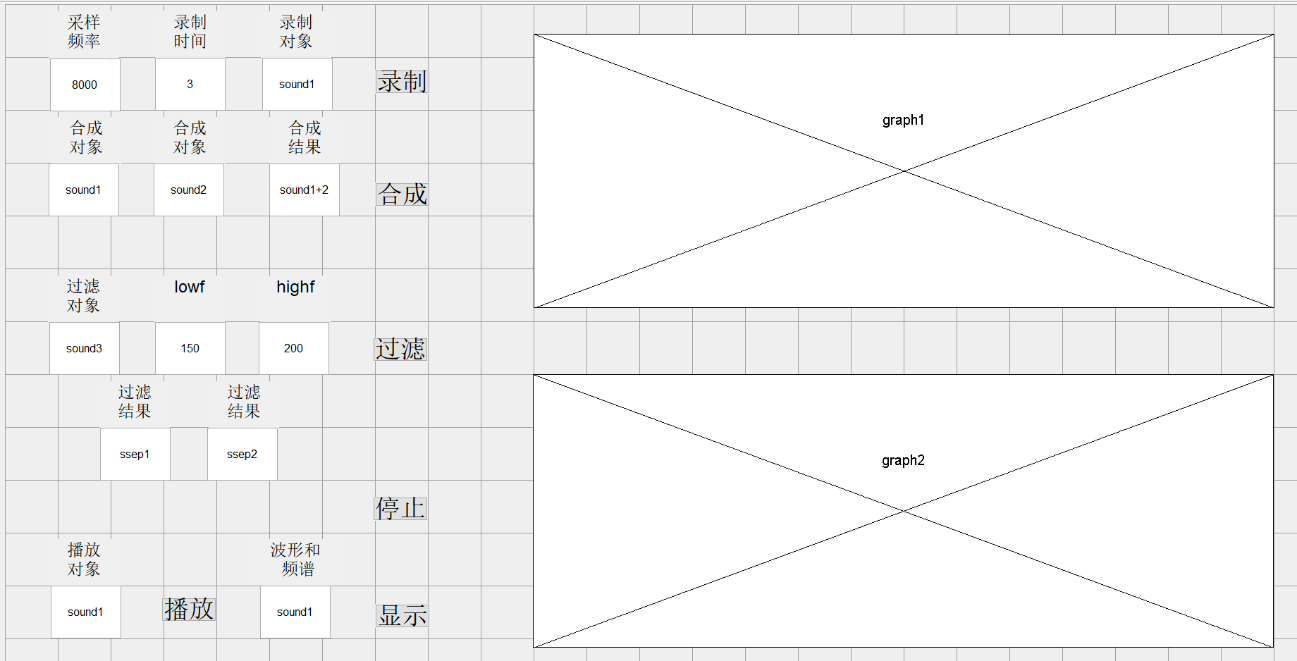
1. 为什么对lowf和highf做出限制？

设计的要求为分离男女声，对于不了解男女声基音频率区别的使用者，对lowf和highf做出限制是为了确保两者的值不会过于离谱，避免分离效果极差。此外，增加一个截止频率为2KHZ的低通滤波器用于降噪。

1. GUI用户界面

完成整个设计的任务后，我进一步想到，能否将我所做的工作及相关代码进行汇总，制作成GUI用户界面，并将其打包成应用程序。

7.1设计思路

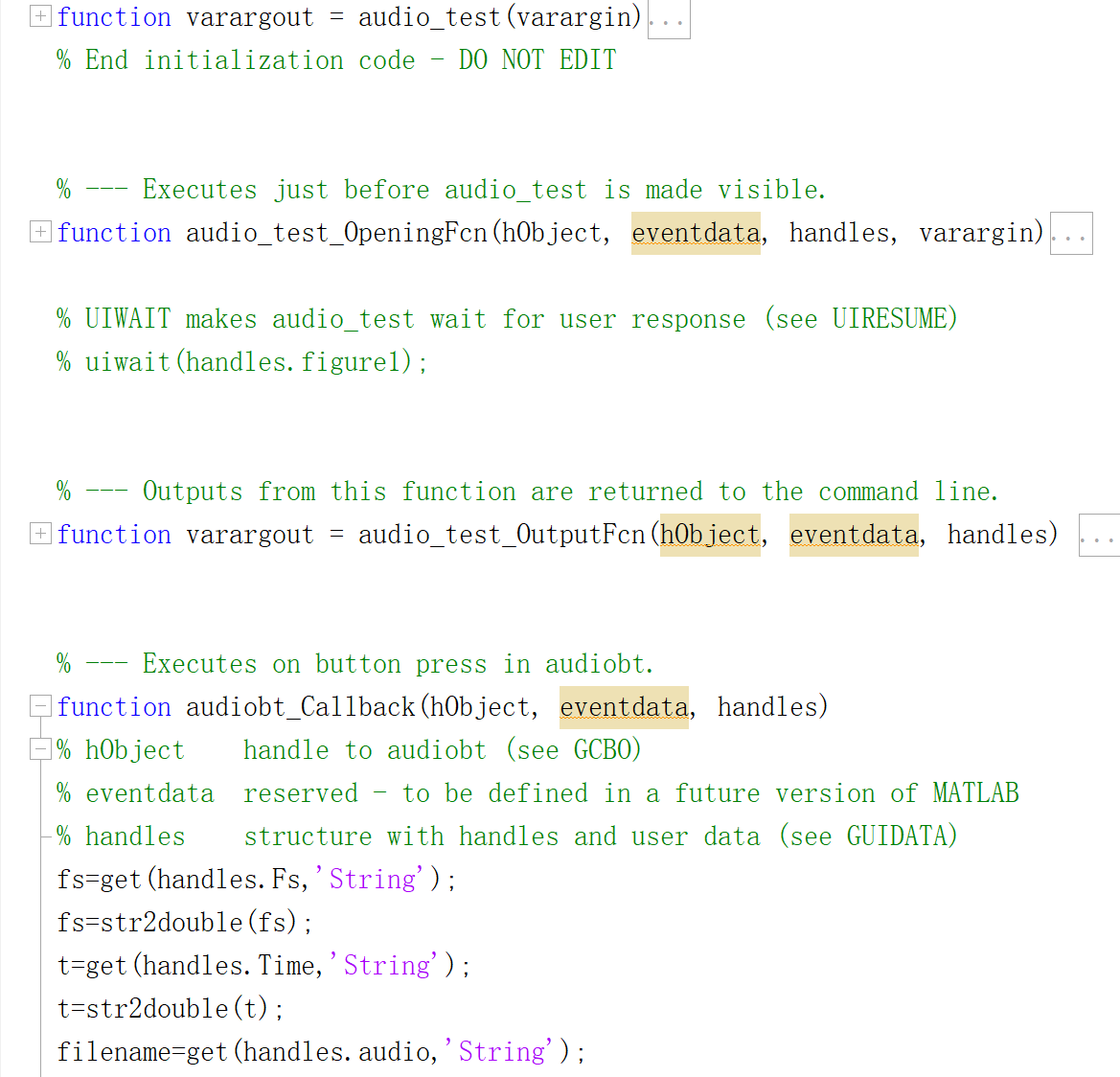
设计的重点在于用户界面中显示窗口和各种实现函数功能的按钮及其相关参数的输入文本框的布局，如下图所示。

这里所有的对象和结果皆是音频文件，后缀为’.wav’，输入时不需要键入后缀。录制、显示、合成、过滤这四个按钮的功能已经在前面提到过。此外，由于应用程序不存在命令行窗口，前面所用的StartFcn和StopFcn均失效，因此在录制结束后会有一个专门的弹出窗口告知使用者。

播放按钮用于播放播放对象，使用audioread函数读取矩阵和采样率后，再由sound函数播放即可。停止按钮用于停止正在播放的音频，使用clear sound语句即可。graph1和graph2都是坐标区，用于显示波形和幅频图。此外，为了能够放大、缩小、拖动图，还增加了对应的工具栏。

7.2实现代码

由于代码过长，此处不给出用户界面的实现代码，如有需要可在资源汇总的Matlab代码文件中的audio\_test.m文件查看。

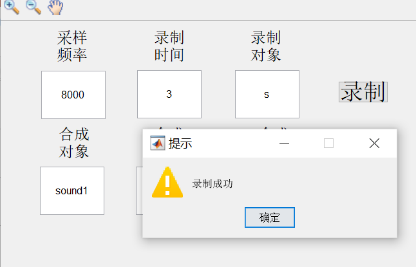
下面是部分代码的截图。

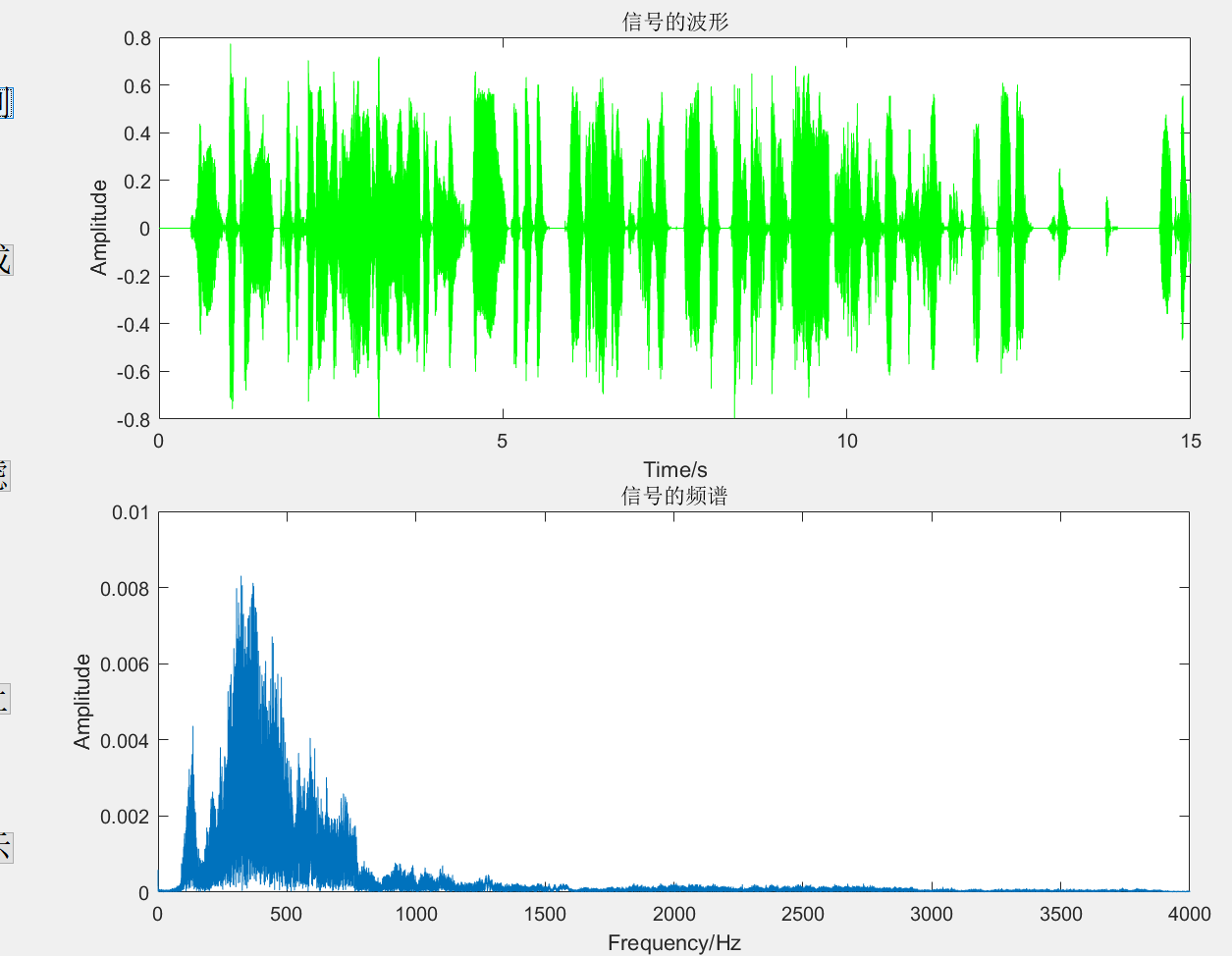
实际上，按钮的响应函数与前文中的实验代码并无多大差异，需要注意的是如何获取参数这一点。handles以结构体的形式储存了用户界面所有控件的信息，修改控件的Tag属性对控件进行命名，使用get函数对应文本输入框的字符串，并将其转化为对应的变量类型即可。

此外，在子函数的最后，必须加上以下命令来改变的handles的储存信息。

guidata(hObject,handles)

7.3运行结果

某次录音结束后的情况如下图所示。

某次显示波形和幅频图的情况如下。

7.4相关问题

一个比较重要的问题是打包成的应用程序并不是在任何环境下都能成功运行的，其主要原因在于代码中涉及到了大量矩阵相关的运算，在未安装Matlab应用程序或其版本不够高的计算机上，缺少对应的插件Matlab runtime v9.9，仅依靠底层的c代码无法运行这个运用程序。

一种解决方法是将这个插件一起打包到这个应用程序中，但这会导致应用程序过大（1GB左右），给下载带来困难，且对于已经拥有合适运行环境的人，这也是对内存的浪费。

于是我采用另一种解决办法，上传MyAppInstaller\_web应用程序用于下载这一插件，并将整个应用程序及下载插件的程序做成压缩包。使用方法是，解压后打开for\_redistribution文件夹运行MyAppInstaller\_web下载runtime，再运行audio\_test文件夹下的audio.exe。

1. 工作汇总

本次设计中，我选择的课题是语音信号处理。该课题包含以下三个要求：(1)请录制持续3秒的语音信号，并播放语音信号。(2)使用不同的采样频率记录持续3秒的语音信号，请草拟其波形和光谱。请找到可以保持语音信号不失真的最低采样频率。(3)请分别录下男生和女生的讲话。组合两个演讲并设计一个过滤器来分隔这两个演讲。将分离的语音与原始语音进行比较，以测试分离效果。我将依次解决这三个问题。

在matlabR2020b版本中，我使用audiorecorder对象进行语音信号处理。audiorecorder对象的初始化及属性，方法及相关函数在本文第二部分中有详细介绍。其中getaudiodata、play、record、recordblocking将会在我们的程序中用到。

本文第三部分介绍了我针对第一小问的设计思路。首先创建一个audiorecorder对象，根据传入的参数确定采样频率Fs和录制时间T。使用recordblocking函数录制音频后，使用play方法即可播放音频。使用getaudiodata函数从audiorecorder对象中获取音频的数值矩阵，并使用audiowrite保存录制内容。实现代码和运行结果在后续中有展示。另外我又介绍了StartFcn和StopFcn，比较record和recordblocking并说明选择后者的原因。

本文第四部分针对第二个问题，即解决获取音频的波形和频谱的问题。我们考虑绘制一个音频文件的波形和频谱。使用audioread函数读取文件filename，获取数值矩阵x和采样频率Fs。对数值矩阵x提取左声道后，得到向量x，我们得到波形的幅值，再借助抽样频率，我们便能绘制出波形图。要得到幅频图，我们考虑使用fft函数对x进行快速傅里叶变换，得到的行向量X即为幅频曲线的幅值，同样借助抽样频率确定X中每个点所对应的频率值。实现代码和运行结果在后续展示。

本文第五部分针对第三个问题，即对音频进行合成。要组合两个音频文件filename1和filename2，使用audioread函数分别读取对应的采样频率Fs1、Fs2和数值矩阵y1、y2。将相加得到的数值矩阵y和对应的采样频率Fs通过audiowrite函数写入音频文件synfilename即可得到组合的音频。实现代码和运行结果在后续展示。之后，我讨论了针对采样率不同的音频组合的改进方式，从插值法改进到使用重采样函数resample。另外又解决了响度相差太大的音频的组合问题。

本文第六部分针对第三小问的分隔部分，解决音频的男女声分离问题。我主要从频域的区别来分隔出男女声。设计低通滤波器，即可从混合声音信号中将男生声音提取出来。设计高通滤波器，要从混合声音中将女生声音分离出来。Matlab提供lowpass函数和highpass函数分别用于低通滤波和高通滤波。实际的分离效果一般。想要提高分离效果，除了利用信号的频域信息外，还应考虑其他的方法。

本文第七部分，我考虑将我所做的工作及相关代码进行汇总，制作成GUI用户界面，并将其打包成应用程序。设计的重点在于用户界面中显示窗口和各种实现函数功能的按钮及其相关参数的输入文本框的布局，在本文的第七部分中有展示。

1. 体会

本次设计，我选择了语音信号处理课题。我完成了三项基本任务，并且在此基础上还设计了一个GUI用户界面，并将其打包成了一个应用程序。可以说圆满完成了设计的任务。

语音信号处理这一课题不仅要求我们有扎实的信号与系统方面的知识，还要求我们使用Matlab作为工具处理语音信号。通过此次设计，一方面我们对本课程的一些基础知识如波形图，采样频率，滤波器等有了更深的理解。另一方面我们对Matlab中的许多函数例如getaudiodata等audiorecorder对象的相关函数进行了初步的学习和使用。不仅如此，在这次设计中我们还遇到了很多实际性的问题，例如如何组合采样率不同的音频？如何组合响度相差太大的音频？经过努力，我成功克服了困难达成了目标，并在此基础上设计了简练的用户界面使课设的完成度更高。

总体而言，此次课设激发了我学习matlab的兴趣，也让我体会到了Matlab这一软件功能的强大，更让我们充分认识到团队合作的重要性。一定会对我们今后的学习工作有很积极的影响。

1. 参考资源

MATLAB绘制音频波形及频谱<https://blog.csdn.net/weixin_43455581/article/details/110958737>

快速傅里叶变换FFT进行频谱分析（matlab）<https://blog.csdn.net/weixin_40215443/article/details/94617053>

关于对FFT分析的一些疑问及解答收集

<https://www.jianshu.com/p/78043f8f8306>

matlab：快速傅里叶（反）变换FFT&IFFT

<https://blog.csdn.net/qq_39297053/article/details/113916666>

Matlab读取写入合成音频信号audioread audiowrite 函数使用方法<https://blog.csdn.net/qq_43157190/article/details/88676838>

不同采样率的信号(出处: MATLAB中文论坛)

<https://www.ilovematlab.cn/thread-312820-1-1.html>

求MATLAB语音调整和不同采样频率的合成点拨<https://bbs.csdn.net/topics/392086971>

MATLAB中filter的理解与使用<https://blog.csdn.net/qq_38559814/article/details/86521602>

MATLAB函数——lowpass

<https://blog.csdn.net/weixin_43870101/article/details/106888716>

MATLAB GUI创建图形用户界面/交互界面<https://blog.csdn.net/wanjiac/article/details/106226006>

使用matlab设计交互界面（App Designer）<https://blog.csdn.net/tyl532667325/article/details/97639089>

matlab gui编程 获取可编辑文本框数值<https://blog.csdn.net/qq_23565735/article/details/115211595>

信号处理——MATLAB音频信号加噪、滤波<https://blog.csdn.net/zsisinterested/article/details/123386387>