

信号与系统大作业《语音碎片我来拼拼听》

题目1_b 实验报告

1.实现思路

解决本问题要解决两个关键问题。

1)排序与去噪的先后顺序。基于第一问，排序算法的实现还是要通过最大相关解决。但是，加入噪声的信号会对相关函数产生影响，容易产生误差，且不利于对噪声的分类处理，故而应该先去噪后拼接。

2)如何去除加性噪声。本质上高斯白噪声也是加性噪声的一种，我认为对于这两种噪声可以用同一种算法解决。思路最先想到的是小波去噪，阅读文献后发现维纳滤波和自适应滤波对此噪声解决效果也不错。

2.实现方式

2.1小波去噪

小波去噪可能有效去除加性低幅噪声，且matlab中有小波去噪的函数wden，其函数各个参数设置可以参考官方文本或各大博客。

在实际实现中，由于高斯白噪声的功率过大，导致小波效果不好，先过了一个低通滤波器，之后在过小波去噪。对于其余加性噪声，高频分量本来就少，通过低通后对真值的影响不大，仍然用此方法。

具体代码如下，被注释部分即是小波去噪。

```
clear all;close all;clc;

file_path='P1b/';
out_path = 'P1b/out/';
mkdir(out_path);
file_list = dir('P1b/*.wav');
len = length(file_list);
for counter=1:len %%%
    tp = file_list(counter).name;
    [y,fs] = audioread(strcat(file_path,tp));
    %ffs = 2200;
    %fp = 2500;
    %middle = ll_filter(y,ffs,fp);
    %index = 0.6;
    %xxd = wavelettt(middle,index);
    xxd = enhance(y);
    audiowrite(strcat(out_path,tp),xxd,8000);
    %clear sound;
end
out = slice_stick(out_path,24000);
audiowrite('P1b.wav',out,8000);
```

wavelettt.m是集成的的小波去噪函数。ll_filter是低通滤波器。

由于对内置的小波去噪不熟悉，参数调整还有疑问，最后进行了手动强制去低幅噪声。

```
function xxd = wavelettt(input,index)
xd = wden(input,'heursure','s','mln',10,'bior4.4');
ave = mean(abs(xd));
x_d = (abs(xd)>index*ave);
xxd = (xd).*x_d;
```

效果不错，但之后测试自适应滤波时发现效果更好，采用自适应。

2.2 自适应去噪(语音增强算法)

自适应滤波是在小时间区域内对语音进行拟合，用汉明窗函数等可以有效进行语音增强。此部分实现略复杂，参考百度文库代码enhance.m。

```
%% 自适应滤波
for q=1:2*numofwin-1
    frame=x(1+(q-1)* winsize/2:winsize+(q-1)* winsize/2);
    hamwin(1+(q-1)* winsize/2:winsize+(q-1)* winsize/2)=...
        hamwin(1+(q-1)* winsize/2:winsize+(q-1)* winsize/2)+ham;
    y=fft(frame.* ham);
    mag=abs(y);
    phase=angle(y);
    %幅度谱减
    for i=1:winsize
        if mag(i)- nmag(i)>0
            clean(i)=mag(i)- nmag(i);
        else
            clean(i)=0;
        end
    end
    %在频域中重新合成
    spectral=clean.* exp(1i* phase);
    %反傅里叶变换并重叠相加
    enhanced(1+(q-1)* winsize/2:winsize+(q-1)* winsize/2)=...
        enhanced(1+(q-1)* winsize/2:winsize+(q-1)* winsize/2)+real(iff(spectral));
end
%除去汉明窗引起的增益
for i=1:size
    if hamwin(i)==0
        enhanced(i)=0;
    else
        enhanced(i)=enhanced(i)/hamwin(i);
    end
end
end
```

该代码的思路是首先模拟了一个低幅随机噪声函数，用普减法估计原始信号的“初值”，以此初值处理第一段语音，之后一段段处理迭代处理，最终生成语音增强信号。

处理的代码为之前的非注释部分。

3.主函数

主程序：ex2.m

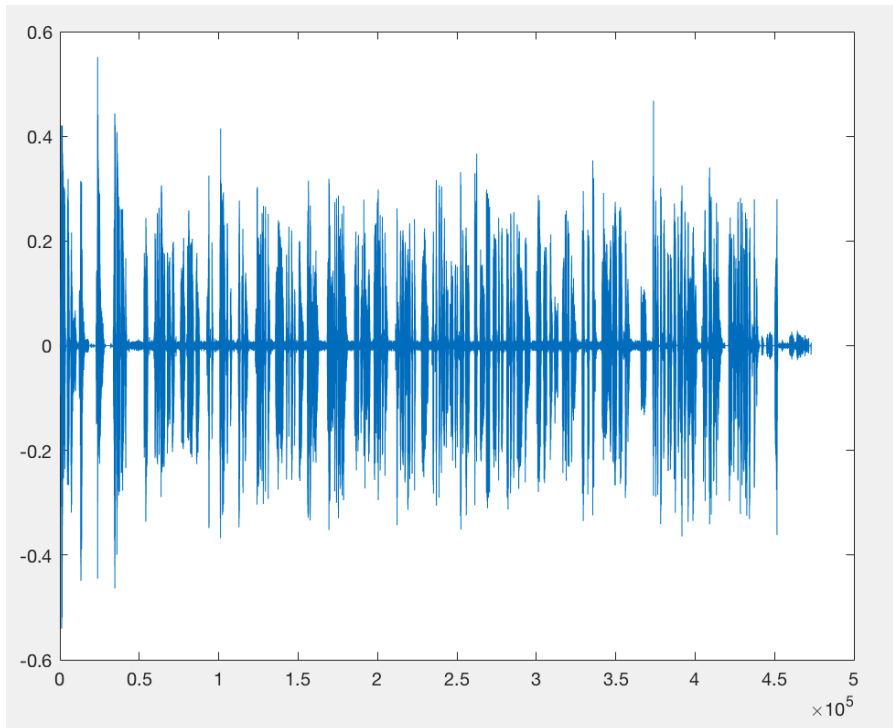
```
clear all;close all;clc;

file_path='P1b/';
out_path = 'P1b/out/';
mkdir(out_path);
file_list = dir('P1b/*.wav');
len = length(file_list);
for counter=1:len %%%
    tp = file_list(counter).name;
    [y,fs] = audioread(strcat(file_path,tp));
    %ffs = 2200;
    %fp = 2500;
    %middle = ll_filter(y,ffs,fp);
    %index = 0.6;
    %xxd = waveletttt(middle,index);
    xxd = enhance(y);
    audiowrite(strcat(out_path,tp),xxd,8000);
    %clear sound;
end
out = slice_stick(out_path,24000);
audiowrite('P1b.wav',out,8000);
```

之后思路同P1a。

4.实验结果及分析

P1b.wav文件为实验结果，下图为处理后画出的波形。



分析此波形可以发现语音信号基本被过滤出来，对于与语音信号明显分离的高斯噪声，去除良好，但是与语音信号粘合紧密的噪声去除效果较差，体现在语音上为试听的时候很“刺啦刺啦”的声音。

后续改进措施我认为小波去噪函数可以很好的解决此问题，由于matlab的各层小波系数阈值无法自己确定导致还会保留一些小声的高斯白噪声，我认为如果自己实现小波去噪函数可以解决此问题。但是由于时间有限且该函数过于复杂未能实现。