# 实验二 语音信号基本时频分析

## 实验目的：

1. 学习与掌握语音信号的分帧及加窗
2. 了解并分析信号加汉明窗及矩形窗后的差别
3. 分析清音与浊音的时域及频谱域波形特征（画图、对比）
4. 学习绘制语音信号的语谱图

## 实验设备

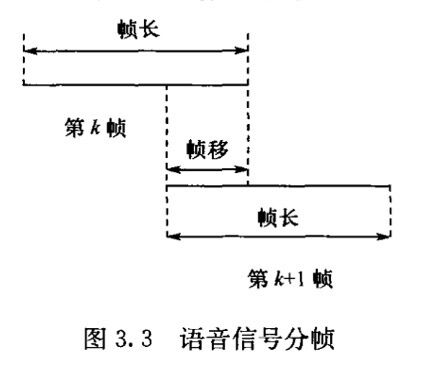
应用软件Matlab2018a，清浊音音频各一段。

## 实验原理

### （1）语音信号的分帧及加窗

语音信号是一种随时间而变化的信号，主要分为浊音和清音两大类。浊音的基音周期、清浊音信号幅度和声道参数等都随时间而缓慢变化。由于发声器官的惯性运动，可以认为在一小段时间里（一般为10~30ms）语音信号近似不变，即语音信号具有短时平稳性。这样，可以把语音信号分为一些短段（称为分析帧）来进行处理。

语音信号的分帧是采用可移动的有限长度窗口进行加权的方法来实现的。一般每秒的帧数约为33~100帧，视实际情况而定。分帧虽然可以采用连续分段的方法，但一般要采用图3.3所示的交叠分段的方法，这是为了使帧与帧之间平滑过渡，保持其连续性。前一帧和后一帧的交叠部分称为帧移，帧移与帧长的比值一般取为0~1/2，图3.3给出了帧移与帧长示意图：



常用的窗有两种，一种是矩形窗，窗函数如下:

另一种为汉明（Hamming）窗，窗函数如下：

矩形窗口给窗口中的所有L个样本赋予相同的权重（n，...，n-L + 1）

汉明窗口给出了中间样本的最大权重，并在窗口的开始和结束时强烈地逐渐减小

这两种窗的时域和频域波形可用Matlab程序实现。

在语音频谱分析时常使用汉明窗，在计算短时能量和平均幅度时通常用矩形窗。

**（2）清浊音分辨**

语音学中，将发音时声带振动的音称为浊音，声带不振动的音称为清音。

清浊音分辨通常从短时能量，短时过零率，短时自相关函数三个参数判断。

**（3）语谱图**

语谱图分析语音又称语谱分析，语谱图中显示了大量的与语音的语句特性有关的信息，它综合了频谱图和时域波形的优点，明显的显示出语音频谱随时间的变化情况。在数字技术发展以前，人们可视化研究语音数据的方法是把数据通过频率滤波器，然后各个频率的数据驱动相应的类似针式打印的设备按频率高低顺序记录在一卷纸上，信号的强弱由记录在纸上的灰度来表示。记录纸按照一定的速度旋转，即相当于在不同的时间里记录下语音数据。

语谱图的横坐标是时间，纵坐标是频率，坐标点值为语音数据能量。由于是采用二维平面表达三维信息，所以能量值的大小是通过颜色来表示的，颜色深，表示该点的语音能量越强。

语谱图实际上是一种动态的频谱。语谱图分为宽带语谱图和窄带语谱图两种。

窄带语谱图：带宽约为45Hz,具有良好的频率分辨率，但时间分辨率较差；

宽带语谱图：带宽约为300Hz，具有良好的时间分辨率，但频率分辨率较差，不同语谱图上所表现的语音特征也不同。

语谱图的时间分辨率和频率分辨率是由窗函数的特性决定的、可以按照短时傅里叶变换的第一种解释来分析频率分辨率。假设时间为固定，短时频谱相当于w（n-m）的频率响应与信号频谱的卷积。如果W（e）的通带带宽为B，则它的频域内可分辨的频谱宽度即为B这样经过卷积后，相隔频率差小于B的谱峰都会合并为一个单峰。如果需要观察语音谐波的细节，则需要提高语谱图的频率分辨率，也就是减小窗函数的带通宽度。由于带通宽度是与窗长成反比的，因此提高频率分辨率必须要增加窗长。这种情况下得到的语谱图称为窄带语谱图。

（4）Matlab语音处理基本指令语法说明

#### enframe（需安装voicebox包）

**y=enframe(x,framelength,step) %把语音信号按帧长和帧移进行分帧**

x-语音信号

framelength-帧长或窗函数，若为窗函数，帧长便取窗函数长；

step-帧移

**nf = fix((nx-framelength+step)/step);**

%帧数的计算公式为：（序列点数-帧长度+帧移）/帧移

enframe函数的输出矩阵为帧数\*帧长

#### linspace %生成线性间距向量

**y = linspace(x1,x2);** %返回包含 x1 和 x2 之间的 100 个等间距点的行向量

**y = linspace(x1,x2,n);** %生成 n 个点。这些点的间距为 (x2-x1)/(n-1)。

#### rectwin

**w = rectwin(L);** %L为窗长

#### hamming

**w = hamming(L);** %L为窗长

**w = hamming(L,sflag);**  %sflag为窗口采样

#### zeros 创建全零数组

**X = zeros %**返回标量 0。

**X = zeros(n) %**返回一个 n×n 的全零矩阵。

**X = zeros(sz1,...,szN) %**返回由零组成的 sz1×...×szN 数组，其中 sz1,...,szN 指示每个维度的大小。

**X = zeros(sz) %**返回一个由零组成的数组，其中大小向量 sz 定义 size(X)。

#### line 创建基本线条

**line(x,y) %**使用向量 x 和 y 中的数据在当前坐标区中绘制线条。如果 x 和 y 中有一个是矩阵或两者都是矩阵，则 line 将绘制多个线条。与 plot 函数不同，line 会向当前坐标区添加线条，而不删除其他图形对象或重置坐标区属性。

示例

**line(x,y,z)** %在三维坐标中绘制线条。

**line** %使用默认属性设置绘制一条从点 (0,0) 到 (1,1) 的线条。

#### axis 设置坐标轴范围和纵横比

**axis(limits) %指定当前坐标区的范围。以包含 4 个、6 个或 8 个元素的向量形式指定范围。**

**⑥ Spectrogram**

**s = spectrogram(x,window,noverlap,nfft,fs)使用nfft采样点计算离散傅里叶变换。**

**X :输入信号**

**Window:窗函数，默认为nfft长度的海明窗Hamming**

**Noverlap:每一段的重叠样本数，默认值是在各段之间产生50%的重叠**

**Nfft:做FFT变换的长度，默认为256和大于每段长度的最小2次幂之间的最大值。**

**Fs:采样频率，默认值归一化频率**

## 实验内容

### （1）对语音信号进行分帧加窗

对语音信号进行分帧，可以利用voicebox工具箱中的函数enframe。

### （2）观察窗函数及分帧后信号的时域频域图

1. 设计并对比窗函数：本步要求利用window函数设计矩形窗(rectwin)、汉明窗(hamming)，通过画出并观察其时域及频域图，得出结论。
2. 取出一帧信号，并画出分帧后的信号加汉明窗的时域及频域图画，进行分析对比，
3. 对比分析清浊音时域及频域波形特征得出结论

**（3**）**绘制一帧语音信号的语谱图**

1. 绘制一帧清音信号加汉明窗后的宽、窄带频谱及语谱图
2. 绘制一帧浊音信号加汉明窗后的宽、窄带频谱及语谱图

## 实验结果

### 对语音信号进行分帧

1. 代码

***clear all;***

***[x,fs]=audioread('D:\MYData\voice\sy2\清音f.m4a'); %清音的获取***

***[y,fs]=audioread('D:\MYData\voice\sy2\浊音b.m4a'); %浊音的获取***

***wlen=960; %设置窗长***

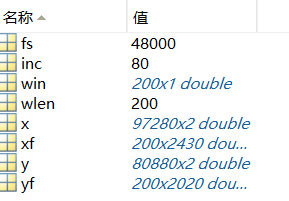
***win=hamming(wlen);***

***inc=400; %设置帧移***

***xf=enframe(x,win,inc)'; %分帧、加汉明窗***

***yf=enframe(y,win,inc)'; %分帧、加汉明窗***

1. 运行结果



### 设计窗函数并绘出其时域频域图

### **矩形窗：**

①代码

***x = linspace(0,100,10001); %在0~100的横坐标间取10001个值***

***h = zeros(10001,1); %为矩阵h赋0值***

***h(1:2001) = 0;***

***h(2002:8003) = 1; %窗内值取为1***

***h(8004:10001) = 0;***

***figure(1);***

***subplot(1,2,1);***

***plot(x,h,'b');***

***title('矩形窗时域波形');***

***xlabel('样点数');***

***ylabel('幅度');***

***axis([0,100,- 0.5,1.5]); %限定横纵坐标范围***

***line([0,100],[0,0]);***

***w1 = linspace(0,61,61);***

***w1(1:61) = 1;***

***w2 = fft(w1,1024); %对时域信号进行1024点的傅里叶变换***

***w3 = w2/w2(1); %幅度归一化***

***w4 = 20\*log10(abs(w3)); %对归一化幅度的模值取对数***

***w = 2\*[0:1023]/1024; %频率归一化***

***subplot(1,2,2);***

***plot(w,w4,'b');***

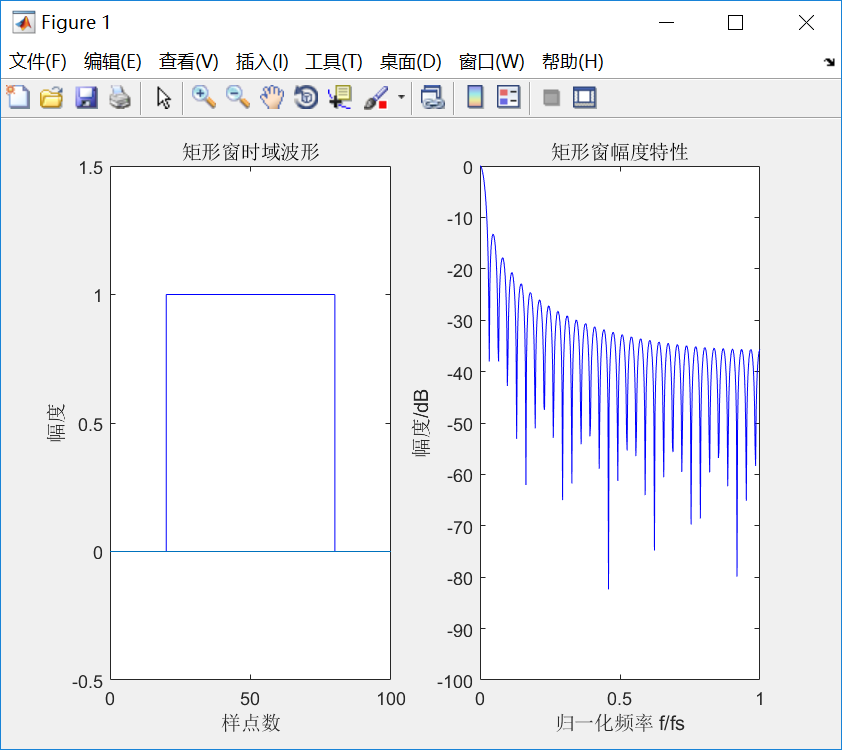
***axis([0,1,- 100,100]);***

***title('矩形窗幅度特性');***

***xlabel('归一化频率 f/fs');***

***ylabel('幅度/dB');***

②运行结果：



**汉明窗：**

①代码

***x = linspace(20,80,61); %在20~80之间取61个值***

***h = hamming(61); %取61个点的汉明窗值为纵坐标值***

***figure(1); %画图***

***subplot(1,2,1); %分配画布***

***plot(x,h,'b'); %画出波形图***

***title('汉明窗时域波形');***

***xlabel('样点数');***

***ylabel('幅度');***

***w1 = linspace(0,61,61); %取窗长内的61个点***

***w1(1:61) = hamming(61); %加汉明窗***

***w2 = fft(w1,1024); %对时域信号做1024点傅里叶变换***

***w3 = w2/w2(1); %幅度归一化***

***w4 = 20\*log10(abs(w3)); %取傅里叶变换的模值，并用分贝(dB)表示***

***w = 2\*[0:1023]/1024; %频率归一化***

***subplot(1,2,2);***

***plot(w,w4,'b');***

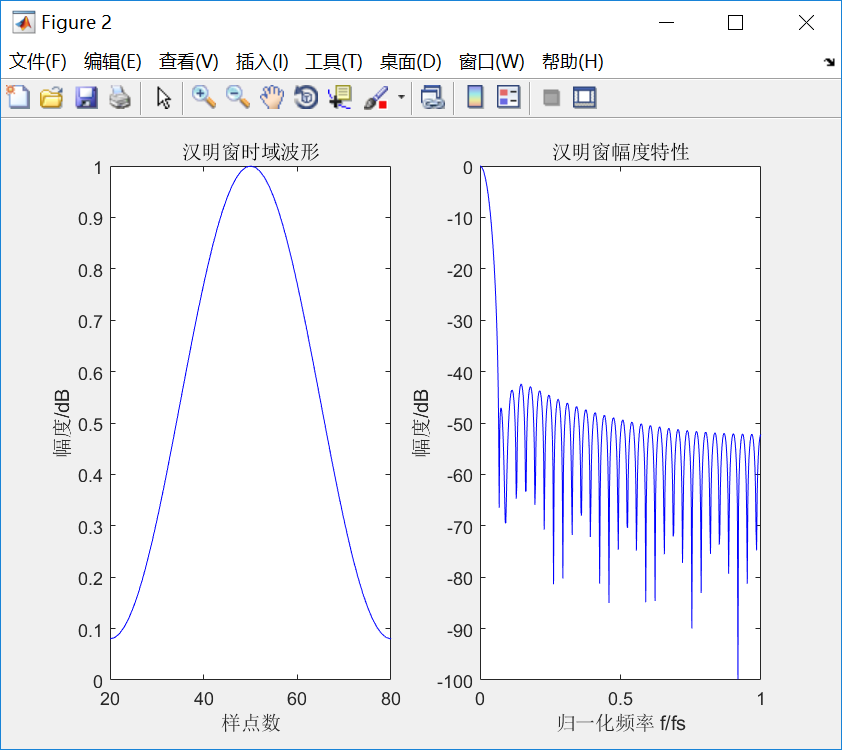
***axis([0,1,-100,0]);***

***title('汉明窗幅度特性');***

***xlabel('归一化频率');***

***ylabel('幅度/dB');***

1. 运行结果：



**对比分析：**

矩形窗的主瓣宽度小于汉明窗，具有较高的频谱分辨率，但是旁瓣峰值较大，频谱泄露比较严重。汉明窗主瓣宽度较宽，旁瓣衰减较大，低通特性较好，能够较高程度的反应信号的频率特性。

### 绘制一帧语音信号的时域频域图

1. 代码：

***%清音***

***fn1=size(xf,2); %帧数***

***xf1=xf(:,200); %取出第200帧***

***lxf1=length(xf1);***

***txf1=(0:lxf1-1)/fs;***

***figure('Name','清音','NumberTitle','off');***

***subplot(2,1,1); %绘制一帧清音信号时域图***

***plot(txf1,xf1);***

***xlabel('时间/s');***

***ylabel('幅度');***

***title('清音信号时域图');***

***grid;***

***xf2=fft(xf1);***

***xf3=20\*log10(abs(xf2+eps));***

***subplot(2,1,2);***

***plot(fs\*(0:lxf1/2-1)/lxf1,xf3(1:lxf1/2)); %绘制一帧清音信号频谱图***

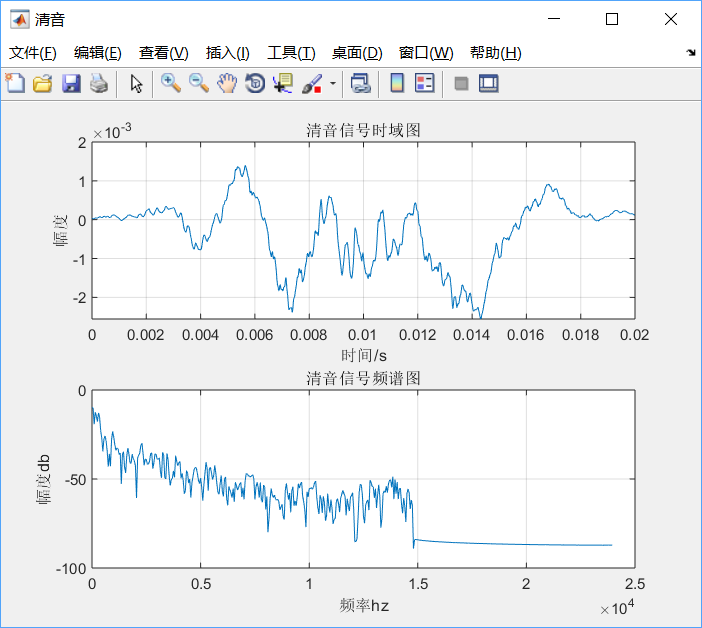
***title('清音信号频谱图');***

***xlabel('频率hz');***

***ylabel('幅度db');***

***grid;***

1. 运行结果：



**浊音：**

1. 代码：

***fn2=size(yf,2); %帧数***

***yf1=yf(:,200); %取出第200帧***

***lyf1=length(yf1);***

***tyf1=(0:lyf1-1)/fs;***

***figure('Name','浊音','NumberTitle','off');***

***subplot(2,1,1); %绘制一帧浊音信号时域图***

***plot(tyf1,yf1);***

***xlabel('时间/s');***

***ylabel('幅度');***

***title('浊音信号时域图');***

***grid;***

***yf2=fft(yf1);***

***yf3=20\*log10(abs(yf2+eps));***

***subplot(2,1,2);***

***plot(fs\*(0:lyf1/2-1)/lyf1,yf3(1:lyf1/2)) %绘制一帧浊音信号频谱图***

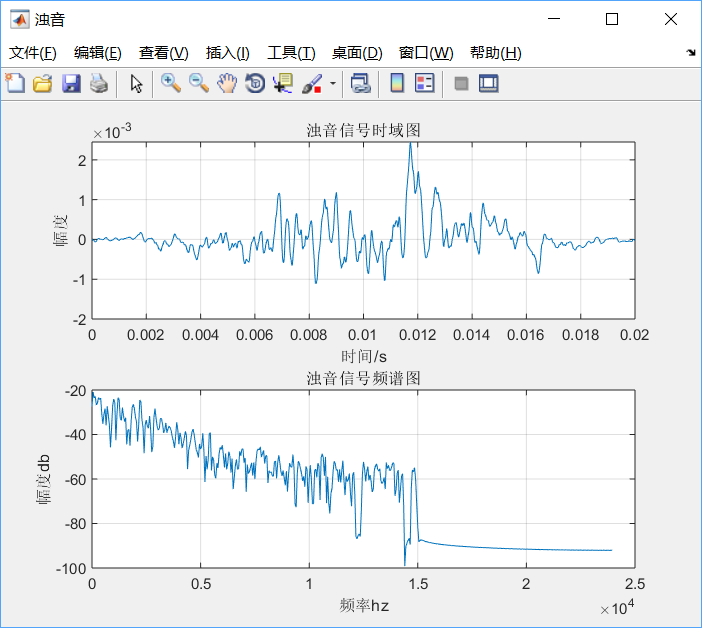
***title('浊音信号频谱图');***

***xlabel('频率hz');***

***ylabel('幅度db');***

***grid;***

1. 运行结果：



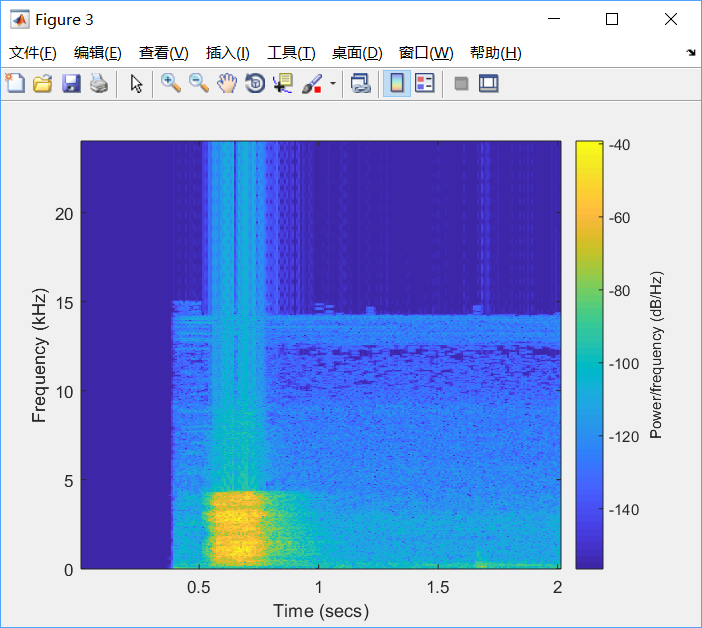
### 语谱图

1. 代码：

***Figure;***

***spectrogram(x(:,2),win,560,1024,fs,'yaxis');***

1. 运行结果：



浊音信号语谱图

①代码：

***Figure;***

***spectrogram(y(:,2),win,560,1024,fs,'yaxis');***

②运行结果：

