****

**《机器学习与振动信号处理》**

**实验报告**

|  |  |
| --- | --- |
| 学 院： | 航海学院 |
| 学　　号： | 2020301020 |
| 姓　　名： | 邱梁城 |
| 专 业： | 信息工程 |
| 实验地点： | 教学东楼D204 |
| 指导教师： | 杨宏晖 |

**西北工业大学**

**2023年5月9日**

目录

[实验二：声音信号的采集 3](#_Toc20511)

[一、 实验目的 3](#_Toc12157)

[二、 实验原理与方法 3](#_Toc2292)

[三、 实验内容与结果 4](#_Toc24655)

[1） 实验内容 4](#_Toc4013)

[2） 实验结果 4](#_Toc32565)

[声音信号时域图 4](#_Toc7650)

[时频图 5](#_Toc1918)

[四、 实验讨论 5](#_Toc4468)

[附录 6](#_Toc6181)

# 实验二：声音信号的采集

## 实验目的

1. 掌握对声音信号的采集
2. 掌握不同采样频率下声音信号的变化

## 实验原理与方法

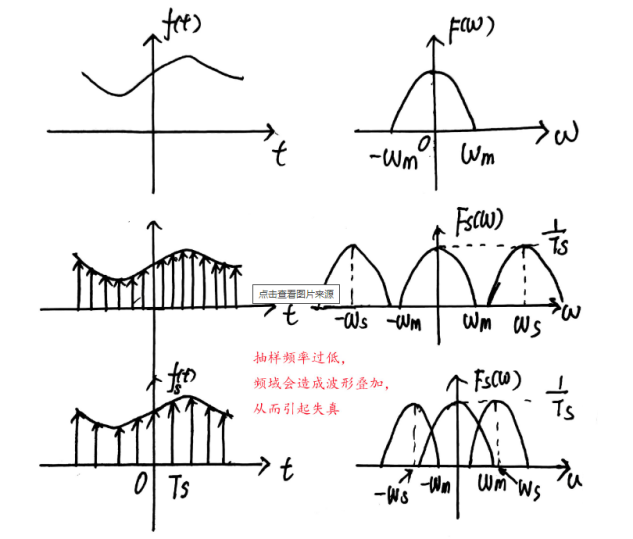
 本实验基于声音信号的模拟信号转换为数字信号的过程。具体来说，当声音信号到达麦克风或其他声音传感器时，传感器会将声音信号转换为模拟电信号。然后，这个模拟电信号经过放大、滤波等处理后，通过数据采集设备（如声卡）转换为数字信号。数据采集设备会按照一定的时间间隔（即采样周期）对模拟信号进行采样，将每个时间点上的模拟信号值转换为数字信号值，并存储在计算机内存或硬盘中。这样就得到了一组数字化的声音信号数据。

由于声音信号是非平稳信号，它们的频率和能量随时间变化。傅里叶变换可以将这些信号从时域转换到频域，对整个信号进行频率分析，无法提供信号在时间上的局部信息。短时傅里叶变换将信号分成若干个窗口，在每个窗口内对信号进行傅里叶变换，获得信号在该时间段内的频率分布情况。可以获得信号在时间和频率上的分布信息，更好地理解和分析信号的特性。

采样定理：一个信号的频率成分可以通过正弦和余弦函数的叠加来表示。当信号的频率超过采样率的一半时，采样点之间的信息就会相互干扰，导致无法恢复原始信号。因此，采样率必须高于信号频率的两倍，表示为：



才能保证采样点之间的信息不相互干扰，从而完全保留原始信号的所有信息。



## 实验内容与结果

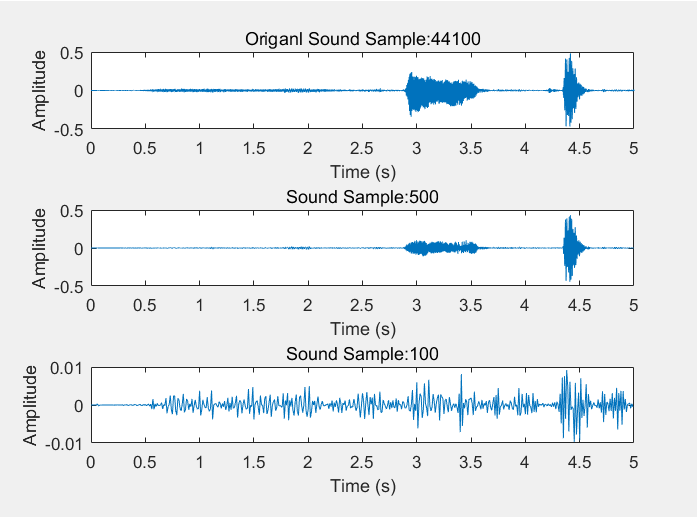
### 实验内容

①通过matlab程序对声音信号进行采样得到声音文件

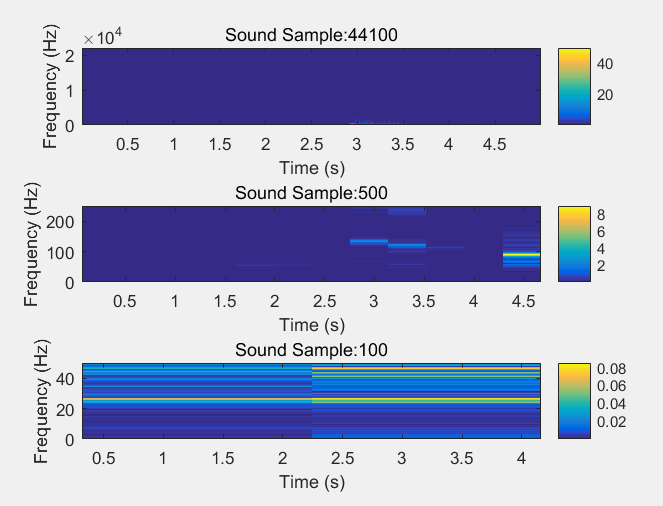
②通过对采样频率的改变观察声音信号的时域图，对采样定理进一步的了解。

### 实验结果

**声音信号时域图**



**时频图**



## 实验讨论

1. 我们可以知道由于人耳可以听到的频率为20-20khz，所以我们第一次所选取的采样频率为44100hz，基本可以将所有的信息都采集完成，我们将它作为原始信号，对其进行复采样，来验证采样定理。
2. 当采样频率改编为500hz时，其信号失真不是过于严重，仅有小部分的信号发生变化，但是当采样频率改为100hz时，我们从时域图与时频图都可以看出，信号发送了严重的失真，我们可以据此判断，我采集到的声音信号其最大频率一定大于50hz。

## 附录

clc;

clear;

% 读取声音文件

filename = 'recorded\_sound.wav';

[samples, original\_sample\_rate] = audioread(filename);

sample\_rate=original\_sample\_rate;

figure

subplot(3,1,1)

time = (0:length(samples)-1)/sample\_rate;

figure(1)

plot(time, samples);

xlabel('Time (s)');

ylabel('Amplitude');

title('Origanl Sound Sample:44100');

window = hamming(256);

noverlap = length(window)/4;

nfft = 512;

fs = original\_sample\_rate;

% 计算短时傅里叶变换

[S, f, t] = spectrogram(samples , window, noverlap, nfft, fs);

figure(2)

subplot(3,1,1)

imagesc(t, f, abs(S));

set(gca,'YDir','normal');

xlabel('Time (s)');

ylabel('Frequency (Hz)');

title('Sound Sample:44100')

colorbar;

% 设置自定义采样频率--------------------------------------

custom\_sample\_rate = 500;

% 重采样

resampled\_samples = resample(samples, custom\_sample\_rate, original\_sample\_rate);

% 播放录音数据

% soundsc(resampled\_samples, custom\_sample\_rate);

% 绘制声音波形图

time = (0:length(resampled\_samples)-1)/custom\_sample\_rate;

figure(1)

subplot(3,1,2)

plot(time, resampled\_samples);

xlabel('Time (s)');

ylabel('Amplitude');

title('Sound Sample:500');

window = hamming(256);

noverlap = length(window)/4;

nfft = 512;

fs = custom\_sample\_rate;

% 计算短时傅里叶变换

[S, f, t] = spectrogram(resampled\_samples , window, noverlap, nfft, fs);

figure(2)

subplot(3,1,2)

imagesc(t, f, abs(S));

set(gca,'YDir','normal');

xlabel('Time (s)');

ylabel('Frequency (Hz)');

title('Sound Sample:500')

colorbar;

%------------采样率100-----------------

custom\_sample\_rate = 100;

resampled\_samples = resample(samples, custom\_sample\_rate, original\_sample\_rate);

% 绘制声音波形图

time = (0:length(resampled\_samples)-1)/custom\_sample\_rate;

figure(1)

subplot(3,1,3)

plot(time, resampled\_samples);

xlabel('Time (s)');

ylabel('Amplitude');

title('Sound Sample:100');

% 计算频域信息

window = hamming(256);

noverlap = length(window)/4;

nfft = 512;

fs = custom\_sample\_rate;

% 计算短时傅里叶变换

[S, f, t] = spectrogram(resampled\_samples , window, noverlap, nfft, fs);

% 绘制STFT图像

figure(2)

subplot(3,1,3)

imagesc(t, f, abs(S));

set(gca,'YDir','normal');

xlabel('Time (s)');

ylabel('Frequency (Hz)');

title('Sound Sample:100')

colorbar;