# 实验一 自动识别技术——语音识别系统的设计与实现（1）

## 1. 语音识别原理

语音是一种波，声音经过话筒或麦克风采集后转换成连续变化的电信号，电信号再经过放大和滤波，被电子设备以一个固定的频率进行采样，每个采样值就是当时检测到的电信号幅值。接着，电子设备会将采样值由模拟信号量化为二进制表示的数字信号。最后对数字信号编码，存储为音频流数据。编码后为了节省存储空间，还会对音频流数据进行压缩，常见的 MP3 文件就是一种压缩后的音频流数据。

### 1.1 WAV 文件

处理音频流数据时，必须是非压缩的纯波形文件，WAV 就是最常见的无压缩声音文件格式之一。WAV 文件里存储的内容除了一个文件头以外，就是声音波形的采样点。WAV 文件还原的声音音质如何，取决于声音采样样本的多少，即采样频率的高低。采样频率越高，音质越好，但 WAV 文件也就越大。

一个 WAV 文件的参数包括采样频率、采样精度和声道数。

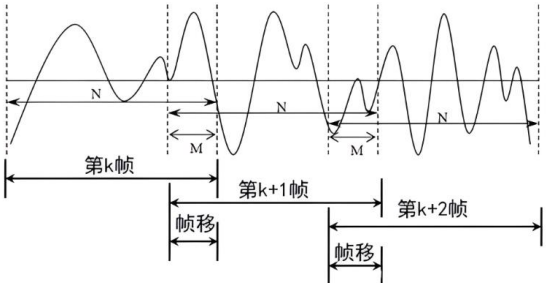
- 采样频率：每秒钟采集音频数据的次数。
- 采样精度：是指数字化后每个样本占用的位数。采样精度越高，则可以表示更大的电压变化范围，采样的信号质量也可以得到更好的保证。
- 声道数：分单声道和立体声。单声道的声音只能一个喇叭发声，立体声的声音可以使两个喇叭都发声，这样更能感受到音频信息的空间效果，但采集的数据量会增加 1 倍。

### 1.2 声学特征提取

有了数字化的音频文件之后，语音识别的第 2 步就是进行声学特征提取。语音数据往往不能像图像任务那样直接输入到模型中训练，其在长时域上没有明显的特征变化，很难学习到语音数据的特征。按 1 秒 16000 个采样点的采样频率来说，直接输入时域采样点训练数据量大且很难有训练出实际效果。因此语音任务通常是将语音数据转化为声学特征，再作为声学模型的输入。声学特征提取的过程如下：

(1) 首先，语音识别前，通常先把首尾端的静音切除，降低干扰。

(2) 其次，对声音分帧，也就是把声音切开成一小段一小段，每小段称为一帧。分帧操作一般不是简单的切开，而是使用移动窗函数来实现。帧与帧之间一般是有交叠的。




(3) 分帧后，语音就变成了很多小段。但波形在时域上几乎没有描述能力，因此必须将波形作变换。常见的一种变换方法是提取 MFCC（Mel Frequency Cepstrum Coefficient，MFCC 梅尔频率倒谱系数）特征。通过研究人耳的生理特性表明，人的听觉对频率是有选择性的。也就说，它只让某些频率的信号通过，无视它不想感知的某些频率信号。MFCC 特征保留了语义相关的一些内容，过滤掉了诸如背景杂音等无关的信息。我们把每一帧波形变成一个多维向量，可以简单地理解为这个向量包含了这帧语音的内容信息。这个过程叫做声学特征提取。实际应用中，这一步有很多细节，声学特征（语音特征）也不止有 MFCC 这一种。*（语音识别的理论基础如有兴趣请自学）*

具体提取哪些特征，要看模型要识别哪些内容，一般只是语音转文字的话，主要是提取音素；但是想要识别语音中的情绪，可能就需要提取响度、音高等参数。

### 1.3 匹配识别

提取音频的声学特征之后，语音识别的最后一步就是通过训练好的模型将这些特征进行分类，找出最佳匹配结果。

声学模型是语音识别系统中最为重要的部分之一，主流系统多采用隐马尔科夫模型(HMM)进行建模。但随着人工智能（尤其是深度学习）的发展，通过深度神经网络(DNN)来完成声学建模，模型精度有了更好的效果。

## 2. 实验内容及要求

### 1） 实验内容：录制与播放音频

编写 Python 代码实现录制和播放音频，需要用到 wave 模块和 PyAudio 模块。

(1) 录制音频：效果如下图所示，运行.py 文件，控制台显示信息“recording begins”，表示开始录音了，此时，对着计算机说话，录制时间为 5 秒，5 秒之后，录音停止，同时在控制台显示信息”recording completed “。程序运行完成后，会生成一个.wav 文件。

(2) 播放音频：利用 wave 和 PyAudio 模块编程实现 1）中的.wav 文件自动播放。

### 2） 上传要求

(1) 运行效果图

(2) 源程序文件