评委一评分，签名及备注	队号： 1209	评委三评分，签名及备注
评委二评分，签名及备注	选题： C	评委四评分，签名及备注
题目：语音识别技术的应用		

摘要

语音识别技术（ASR）是一项涉及多学科的综合性技术，并且语音识别系统在现代社会中的应用也越来越广泛，尤其是智能手机客服语音服务方面。因此本文就语音识别技术的各个环节展开建模分析，并通过 10 段语音样本验证语音识别模型的识别精度。

首先，本文通过 Microsoft Visio 软件对语音识别技术的基本原理绘制流程框图；然后通过 Matlab 录制一段语音样本，并对该语音信号做分帧加窗、预加重等预处理操作。

针对问题一，本文从端点检测、特征提取（MFCC 参数）、模式识别三个方面展开建模，通过 Matlab 编程与图像说明语音识别系统的各个环节。对于端点检测环节，我们采取“双门限检测”法找出语音样本的起始点和终止点；特征提取环节，在已经过端点检测的语音样本基础上，本文选取能提高识别性能的 MFCC 参数来分析；关于模式识别环节，我们首先对特征参数进行规整，然后基于神经网络算法详细阐述语音识别的过程。

针对问题二，根据问题一中的模型，本文结合软件工程中面向对象（OOD）的分析方法以及用户操作手册编写规范，为手机运营商制定了可行、简单的用户操作规则。

针对问题三，本文通过设计实验来验证语音识别模型的准确性。首先，根据用户操作规则，我们录制了不同情况下不同人的 10 段语音；然后根据问题一模型建立流程进行语音识别验证；最后结果表明，在本次试验中基于神经网络的语音识别的系统的识别准确率达 75%。

关键字：语音识别；端点检测；MFCC；神经网络；OOD

语音识别技术的应用

1. 问题的重述

语音识别技术（ASR）就是计算机通过对人类语言的认识和理解，将人类的语言信号转变成相应的文本或命令的技术，也就是让计算机听懂人说话。

语音识别是一项涉及多学科的综合性技术，其过程分为训练和识别两个阶段。 在训练阶段，语音识别系统对输入的语音信号进行学习。学习结束后，把学习内容组成语音模型库存储起来；在识别阶段，根据当前输入的待识别语音信号，在语音模型库中查找出相应的词义或语义。

随着智能机的普及，语音识别技术也更加广泛。某手机运营商想利用语音机器人作为客服，处理查询话费、查询余额、查询最新优惠活动等简单问题。试根据语音识别技术系统构建过程，完成以下问题：

1、建立模型说明语音识别技术的各个环节；

2、根据模型，为手机运营商制定一个可行的用户操作规则；

3、根据制定的规则，通过查询话费等实例验证语音识别模型。

2. 问题的分析

1.1 问题 1 的分析

题目要求建立模型说明语音识别技术的各个环节。首先我们需要充分理解语音识别技术的各个流程，通过 Microsoft visio 软件绘制语音识别基本原理框图。然后对语音识别的几个重要环节建模分析，其中包括端点检测、特征提取、模式识别三个环节。

在建模之前，采集小组成员的语音作为分析样本，然后对语音样本做加窗分帧、预加重等基本处理。

针对端点检测环节，我们充分考虑到语音信号的三种分段形式，即无声段、清音段、浊音段；然后利用预处理后的语音样本，分别计算其短时能量、短时过零率，通过“双门限法检测法”找出语音样本的起点和终点；

针对特征提取环节，本分综合分析考虑了线性预测系数(*LPC*)、线性预测倒谱系数(*LPCC*)、梅尔倒谱系数三种不同参数的优缺点后，选取了 *MFCC* 参数作为语音样本的特征参数；然后利用 Matlab 编写 mfcc 函数，进而分析得到语音样本的 *MFCC*，最后通过三维图形加以展示。

针对模式识别环节，本文选取了动态时间归整方法(DTW)，矢量量化方法(VQ)，隐马尔科夫模型方法(HMM)，人工神经网络方法(ANN)中的人工神经网络算法，然后对语音样本和待测语音样本的模式识别过程进行详细阐述，并通过语音规整算法对语音样本的特征值进行计算。通过 Matlab 编写 guizheng 函数和 bp 函数，并为问题 3 语音识别模型的验证做准备。

1.2 问题 2 的分析

根据软件工程中面向对象(OOD)的分析方法，将题目描述作为“语音识别系统”的需求分析，并结合 1.1 中语音识别各个环节的分析过程，对该系统建立软件概念中的动态模型。基于动态模型的建立过程，并参考用户操作规则编写规范制定题目要求的用户操作规则。

1.3 问题 3 的分析

该问题实质上是按照 1.2 中用户操作规则验证 1.1 语音识别模型的识别精度。我们按照操作规则设计实验。首先，分别录制 10 段语音样本，样本内容为“查询话费余额”、“1”、以及非正常语音，10 段样本中将其中 2 段作为标准语音，其他来自于不同人的 8 段语音作为测试语音。然后语音样本验证 1.1 中语音识别模型。并将验证结果通过图表以及模型识别准确率来呈现。

3. 模型假设与符号说明

3.1 模型假设

1．假设实验语音样本能表征不同人的声音特征；

2．假设实验语音样本的录制环境正常，噪声不大；

3．假设实验语音样本的语音是连续的；

4．假设语音规整后的 MFCC 参数能表征所有语音样本的特征。

3.2 符号说明

表 1 符号说明

符号	符号说明
E_{ω}	短时能量
Z_{ω}	短时过零率
$MFCC$	梅尔倒谱系数
N	帧长
fs	采样率
$H_m(k)$	三角滤波器
$H(z)$	一阶高通滤波器
M	滤波器个数

4. 模型的建立与求解

4.1 模型准备

4.1.1 语音识别系统基本原理概述

根据题目描述我们知道，语音识别系统构建过程整体上包括两大部分：训练和识别。训练阶段分别对语言训练数据库与文本训练数据库进行信号处理和挖掘所得到的“声学模型”与“语言模型”。识别阶段对用户实时语音进行自动识别，首先对用户语音进行端点检测去除噪声和静音，然后对语音信号特征提取，利用训练阶段的模型对用户语音进行模式识别、加工处理得到最终的识别结果。

利用 Microsoft Visio 软件，画出语音识别系统基本原理框图如图 1 所示。

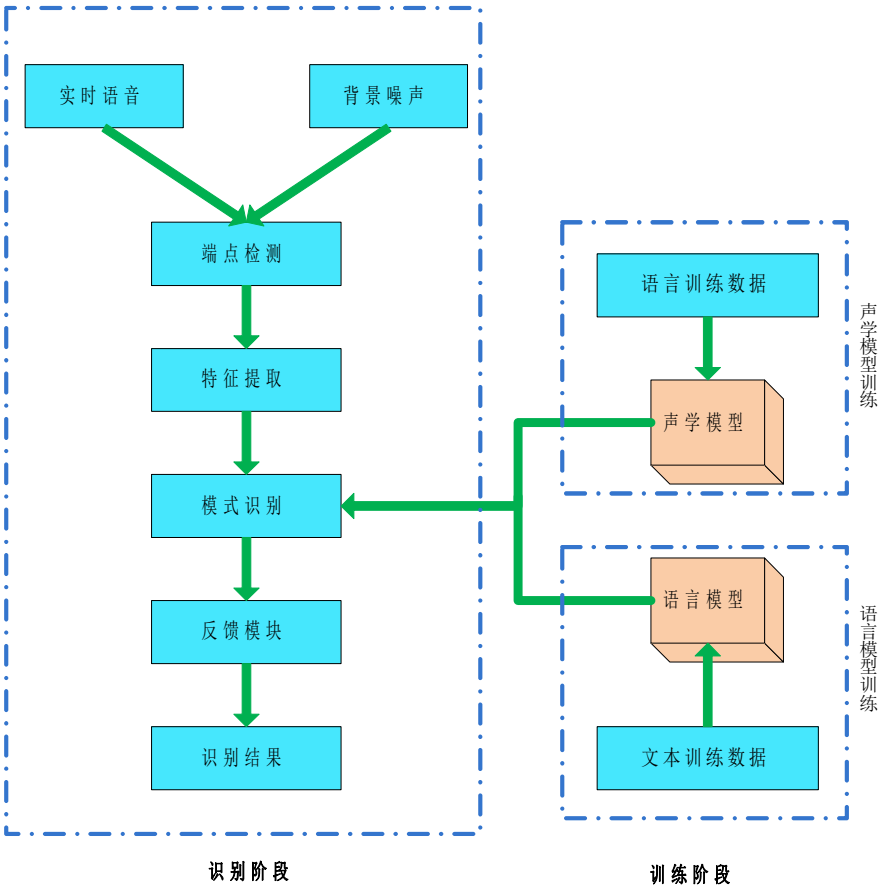


图 1 语音识别系统基本原理框图

4.1.2 语音信号的采集与预处理

（1）样本采集过程

在建模过程中，以本人的声音作为分析样本。在 Matlab 中使用 wavrecord(n,fs,ch,’dtype’)函数录取一段连续的语音“查询话费余额请按 1，查询套餐余量请按 2”的读音。因为人类语音的频谱主要集中在 4kHz 以内，而根据

采样定理，采样频率应大于信号中最高频率的两倍，所以采样频率（fs）取 16KHz；本实验将录音时间规定为 5s，因此采样的总点数（n）为 5*8000，即 40000；通道数（ch）取 1，即为单通道；采样数据的存储格式（dtype）取为“double”，即 16 位采样精度。利用 sound 函数可以较清晰地听到读音，同时发现语音开始时存在短暂杂音。用 wavwrite 函数将语音信号保存为“实验 1.wav”文件。原始语音波形图如图 2 所示（具体 Matlab 程序见附录 7.1）。

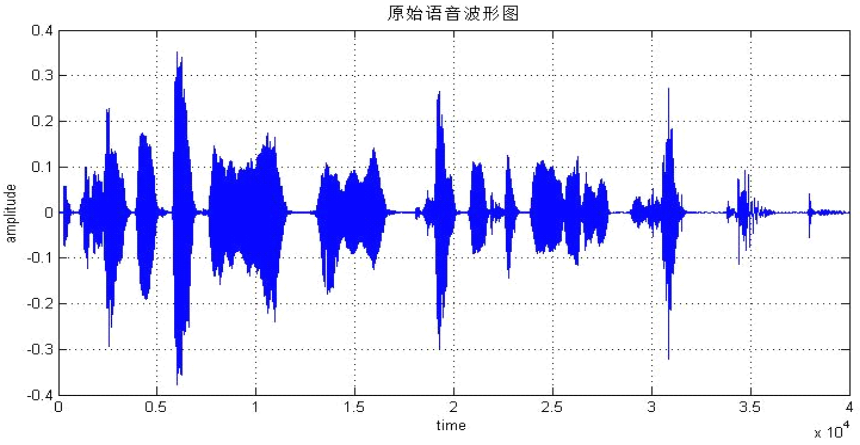


图 2 原始语音信号波形图

（2）加窗分帧处理

由于语音信号从整体上来看是一个非平稳过程，但是在一个短的时间内，其特性保持相对不变，所以语音信号具有短时平稳性，对语音信号的分析和处理必须建立在“短时”的基础上，将信号分为一段一段来分析其特征参数[1]。

分帧使用有线长度的窗函数来截取语音信号形成分析帧，窗函数将需处理区域之外的样点置零来获得当前语音帧。

因此，加窗语音信号为

$$s_{\omega}(n) = s(n) \times \omega(n)$$

在这里窗函数我们选取汉明窗窗函数，即，

$$\omega(n) = \begin{cases} 0.54 - 0.46\cos(\dfrac{2\pi n}{N-1}) & 0 \le n \le N-1 \\ 0 & \text{其它} \end{cases}$$

其中，N 为帧长。

基于上述描述，利用 Matlab 对语音信号进行预处理，对语音信号进行分帧，可以利用 voicebox 工具箱中的函数 enframe。voicebox 工具箱是基于 GNU 协议的自由软件，其中包含了很多语音信号相关的函数。首先可以得到语音信号分帧后波形如图 3 所示，其中我们令帧长 len=200，帧移 inc=100。

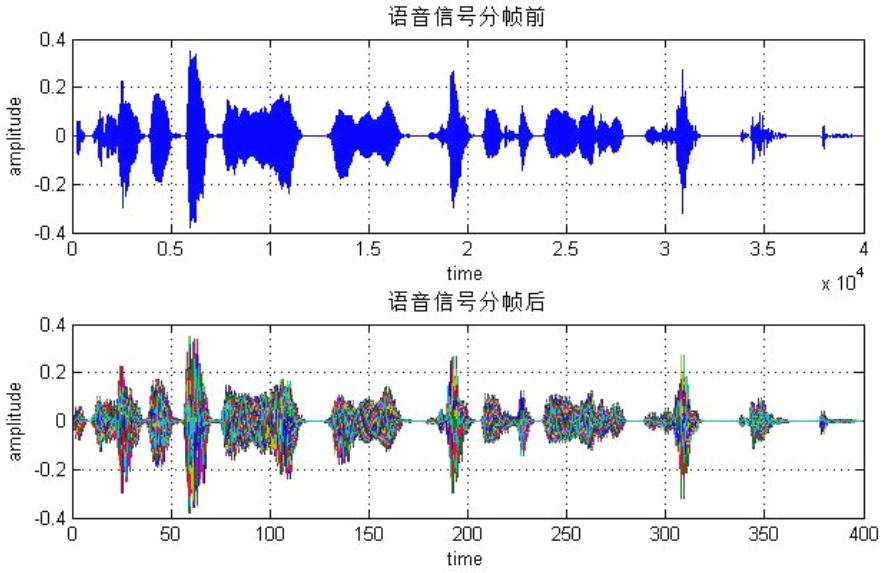


图 3 语音信号分帧后波形图

然后，我们利用 window 函数设计窗口为 120 的汉明窗，进而通过 Matlab 为分帧后的语音信号添加汉明窗，其波形图如图 4 所示(具体程序见附录 7.2)。

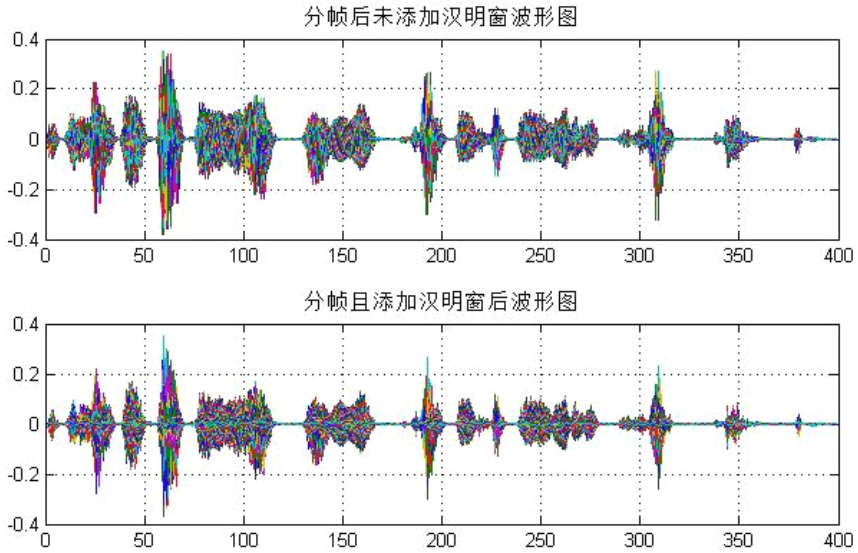


图 4 添加汉明窗后语音信号波形图

（3）预加重处理

人发音时存在口唇的辐射效应，口唇的辐射模型相当与一阶高通滤波器，所

以在对实际信号进行分析处理时，常用“预加重技术”，目的提升信号的高频部分，使信号的频谱更加平坦，方便信号的分析[1]。

即，语音信号通过一个一阶高通滤波器：

$$H(z) = 1 - \alpha z^{-1}$$

其中，$\alpha = 0.9375$。

设n时刻的语音采样值为$x(n)$，经过预加重处理后的结果为：

$$y(n) = x(n) - \alpha x(n-1)\text{，}$$

基于上述描述，利用Matlab对分帧后的语音信号做预加重处理，预加重后语音信号的波形，如图5所示（具体程序见附录7.3）。

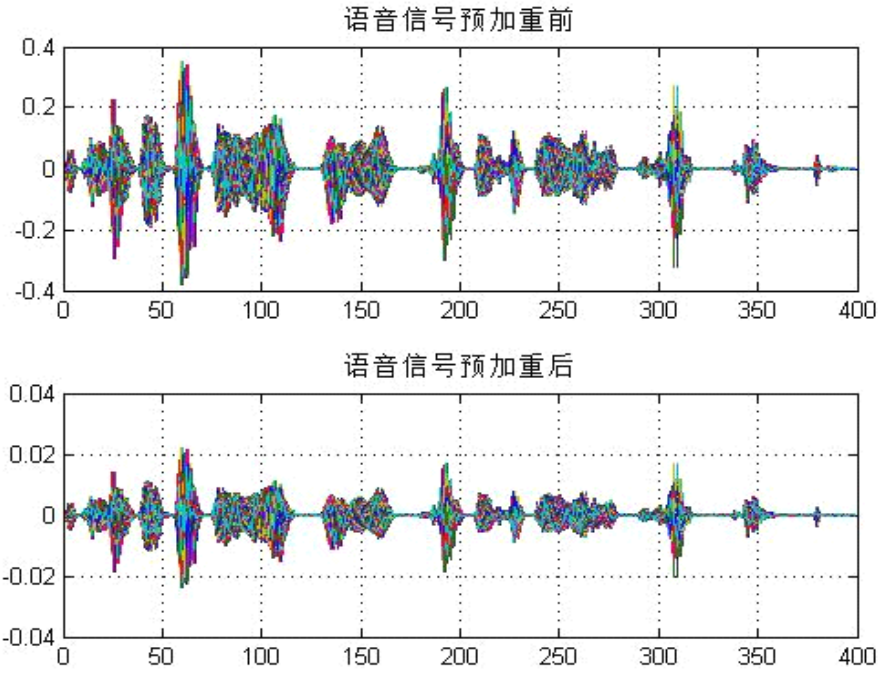


图 5 预加重后语音信号波形图

4.2 问题一的模型建立与求解

4.2.1 端点检测建模

所谓端点检测[2], 就是从一段给定的语音信号中找出语音的起始点和结束点。在语音识别系统中, 正确、有效地进行端点检测不仅可以减少计算量和缩短处理时间, 而且能排除无声段的噪声干扰、提高语音识别的正确率。

语音信号一般可分为无声段、清音段、浊音段。通常利用短时能量来检测浊音，用过零率来检测清音，两者配合实现可靠的端点检测。端点检测算法常用的是由语音能量和过零率组合的“双门限法检测法”。

基于上述描述和端点检测算法，本文从以下两个步骤进行建模分析：

Step1：利用公式分别编程计算“实验 1.wav”语音信号的短时能量、短时过零率。

为了简化模型计算，我们采用矩形窗对语音信号做加窗分帧处理。

(1) 短时能量：由 4.1（2）可得语音波形时域信号加窗分帧处理后得到第 n 帧语音信号为 $s_\omega(n)$，那么第 n 帧语音信号的短时平均能量 E_ω 为：

$$E_\omega = \sum_{m=0}^{N-1} s_\omega^2(n)$$

其中，N 为帧长。

(2) 短时过零率：一帧语音中语音信号波形穿过横轴的次数。它可以用来区分清音和浊音。语音信号 $s_\omega(n)$ 的短时过零率 Z_ω 为：

$$Z_\omega = \frac{1}{2}\sum_{m=0}^{N-1}\left|\mathrm{sgn}[s_w(n)] - \mathrm{sgn}[s_\omega(n-1)]\right|$$

其中，$\mathrm{sgn}[x] = \begin{cases} 1 & (x \geq 0) \\ -1 & (x < 0) \end{cases}$，N 为帧长。

Step2：基于 E_ω 与 Z_ω 的端点检测。

结合数字信号相关理论知识[3]，我们知道无声段的短时能量为零，浊音段的短时能量比清音段的短时能量大，而在过零率方面，无声段理想情况下过零率为零，清音段的过零率比浊音段的过零率大的多；因此，假设一段语音，如果某部分语音短时能量很大或过零率很小，那么认为该部分语音为浊音段，如果某部分语音短时能量很小或过零率很大，那么可以认为该部分语音为清音段，其他为无声段。

表 2 语音段短时端点检测

语音段	短时能量 E_ω	短时过零率 Z_ω	判断
无声段	0	0	理想情况下，两者均为零
清音段	很小	很大	短时能量很小或过零率很大
浊音段	很大	很小	短时能量很大或过零率很小

为了避免在误判以及无声段过零率过大，在利用 Matlab 软件进行分析时，我们设置短时能量最高门限 amp1=10，短时能量最低门限 amp2=2，过零率最高门限 zcr1=10，过零率最低门限 zcr2=5。

根据上述理论，本文通过 Matlab 软件对名为“实验 1”的 wav 文件进行端点检测。首先我们设置帧长 FrameLan 为 200，帧移 FrameInc 为 100 等其他参数的值，然后过零计算，最后计算短时能量。其分析结果如图 6 所示。具体 Matlab 程序见附录 7.4[8]。

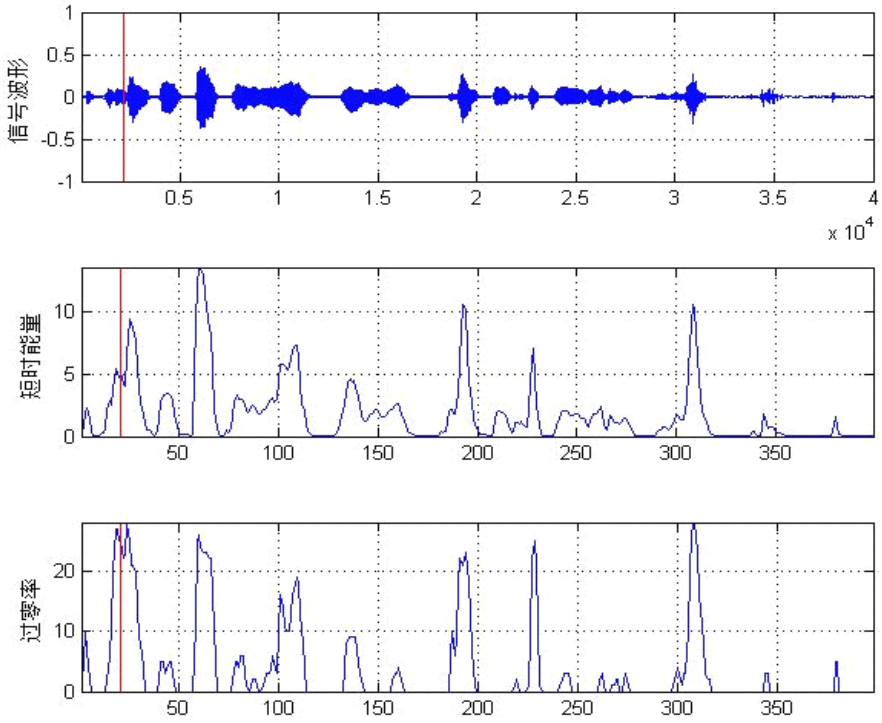


图 6　“实验 1.wav”的短时能量和过零率

4. 2. 2 特征提取建模

语音特征提取[4]就是用较少的维数来表示人语音信息的特征。常用的语音特征包括发生器的谱包络、基音、共振峰等；还有基于声道特征模型，通过线性预测分析得到的参数，比如线性预测系数(*LPC*)、线性预测倒谱系数(*LPCC*)等；还有基于人耳的听觉机理，反映听觉特性，模拟人耳对声音频率感知的特征参数，如梅尔倒谱系数(*MFCC*)等。

LPCC 是基于发音模型建立的，*LPCC* 也是一种基于合成的参数，这种参数没有充分利用人耳听觉的特性，实际上，人的听觉系统是一个特殊的非线性系统，它响应不同频率信号的灵敏度是不同的，基本上是一个对数关系。文献资料[5]表明 *MFCC* 参数能够比 *LPCC* 参数更好地提高系统的识别性能。

基于相关理论知识及上述描述，我们选取 *Mel* 倒谱系数(*MFCC*)进行特征参数提取的建模分析，其计算步骤如下：

Step1：首先确定每一帧语音采样序列的点数，本实验取 $N = 256$。对每帧序

列 $s(n)$ 进行预加重处理后再经过离散傅里叶变换(DFT)，得到离散频谱 $S(n)$ ；

$$S_\alpha(k)=\sum_{n=0}^{N-1}x(n)e^{-j\frac{2\pi nk}{N}},(0\le k\le N)$$

其中， $x(n)$ 为输入语音信号，即“实验 1.wav”语音信息。

Step2：计算 $S(n)$ 的通过 M 个 $H_m(n)$ 后所得的功率值，即计算 $S(n)$ 和 $H_m(n)$ 在离散频率点上乘积之和，得到 M 个参数 P_m ；

$$P_m(k)=\sum_{k=0}^{N-1}\left|S_\alpha(k)\right|^2H_m(k),(0\le m\le M-1)$$

为了简化模型，本文将三角滤波器简化为[6]：

$$H_m(k)=\begin{cases}0 & k<f(m-1)\\ \dfrac{k-f(m-1)}{f(m)-f(m-1)} & f(m-1)\le k\le f(m)\\ \dfrac{f(m+1)-k}{f(m+1)-f(m)} & f(m)\le k\le f(m+1)\\ 0 & k>f(m+1)\end{cases}\quad(\text{其中}\sum_{m=0}^{M-1}H_m(k)=1)$$

Step3：计算 P_m 的自然对数，得到 L_m ；

$$L_m=\ln(P_m)$$

Step4：对 L_m 计算离散余弦变换(DCT)，得到 D_m ；

$$D_m=\sum_{m=0}^{N-1}L_m\cos(\frac{\pi n(m-0.5)}{M})$$

Step5：舍弃代表直流成分的 D_0 ，取得 $D_0,D_1\cdots D_k$ 作为 $MFCC$ 参数。

为了直观的表示 $MFCC$ 参数的计算流程，我们做如图 7 所示的计算框图。

图 7 $MFCC$ 参数计算框图

首先，我们基于相关信号处理技术，做水平方向是时间轴，垂直方向是频率轴，图上的灰度条纹代表各个时刻的语音短时谱的“实验 1.wav”的语谱图。语谱图反映了语音信号的动态频率特性，在语音分析中具有重要的实用价值。被成为可视语言。 如图 8 所示。

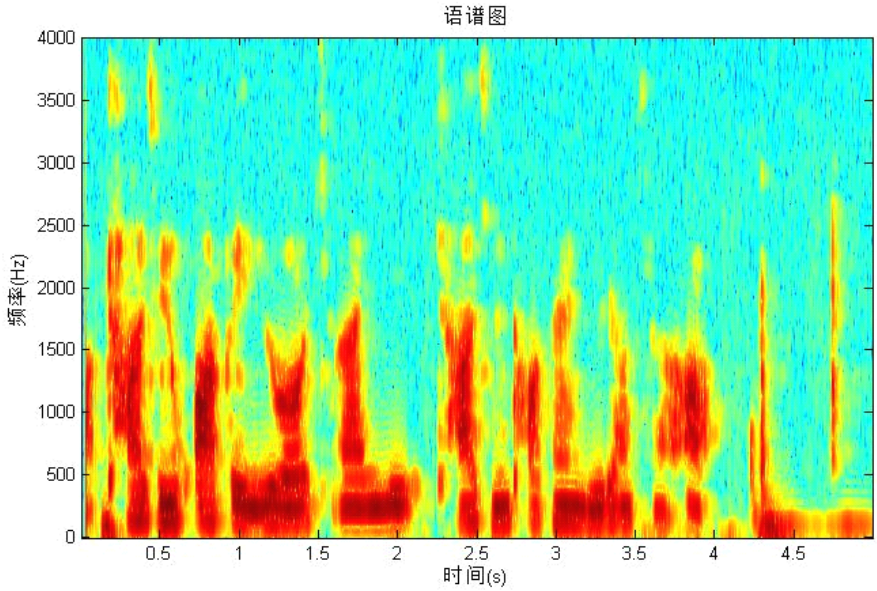


图 8 “实验 1.wav”语谱图

然后，基于图 7 所示的计算流程，本文利用 Matlab 对“实验 1.wav”语音文件特征参数 *MFCC* 提取，编程过程中，我们设采样率 fs 为 8000KHz，置滤波器的个数 M 为 24，一帧语音信号的点数 N 选取常数 256。

我们得到 *MFCC* 参数值与幅值的关系以及维数与幅值得关系，如图 9 所示。

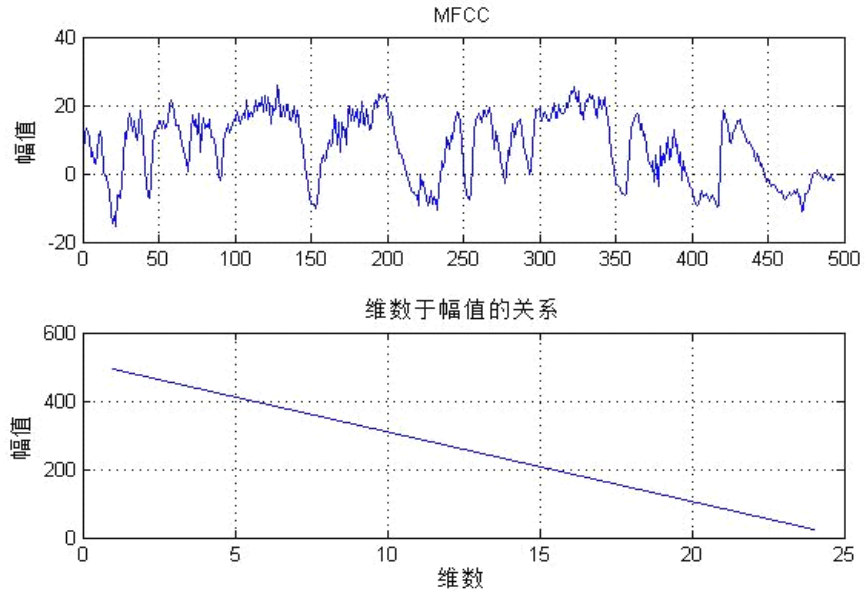


图 9 *MFCC* 参数值

进一步分析，按照本文 *MFCC* 参数提取算法，其中 x 轴为倒谱系数维数、y 轴为语音分析帧数、z 轴为倒谱值，其 *MFCC* 参数值如图 10 所示（具体程序见附录 7.5）。

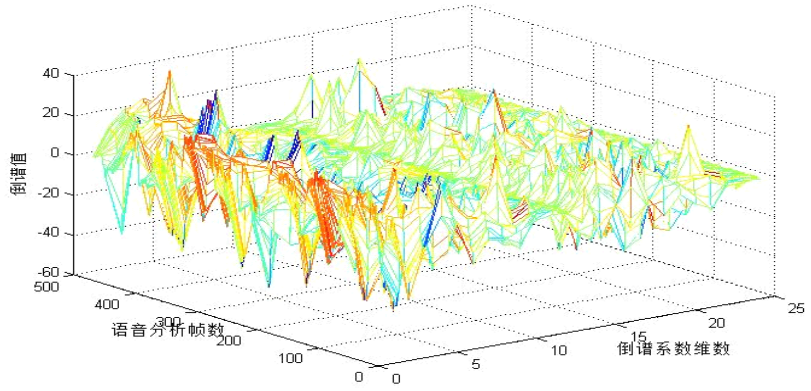


图 10 *MFCC* 参数值

4. 2. 3 模式识别模型

语音信号识别常用的方法有概率统计方法，动态时间归整方法(DTW)，矢量量化方法(VQ)，隐马尔科夫模型方法(HMM),人工神经网络方法(ANN)等[4]。

DTW 算法是较早的一种模式匹配和模型训练方法，它应用动态规划方法成功解决了语音信号特征序列比较时时长不等的难题，在孤立词语音识别中效果好；HMM 模型对动态时间有极强的建模能力，一般用于非特定人、大量词汇、连续语音的识别，但其分类决策能力弱，需要语音信号的先验统计；而人工神经网络具有比较好的分类能力。因此本文选取人工神经网络(ANN)法对语音信号进行模式识别匹配[7]。

1．人工神经网络理论知识

(1) BP 神经网络的拓扑结构

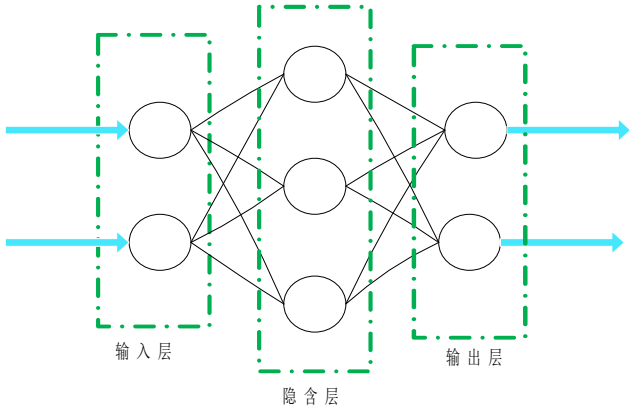


图 11 三层 BP 神经网络结构

神经网络的拓扑结构是指神经元之间的互连结构。BP 神经网络采用的是并行网格结构，包括输入层、隐含层和输出层、经作用函数后，再把隐节点的输出信号传递到输出节点，最后给出输出结果。图 11 是一个三层的 BP 网络结构。

(2) 反向传播算法

这个算法共分为两个阶段，其一正向输入信息，在输入层经隐含层逐层计算各个单元的输出值；

$$net_j = \sum w_{ij} O_i$$

$$O_j = \int (net_j)$$

第二阶段，即反向传播过程，输出误差，逐层向前算出隐含层各个单元的误差，并用此误差修正前层误差。

$$E = \frac{\sum_j (y_j - \hat{y}_j)^2}{2}$$

2．基于神经网络的语音识别

在上述神经网络理论基础知识上，本文建立神经网络模型对语音信号做模式识别。其具体模型建立流程，如图 12 所示。

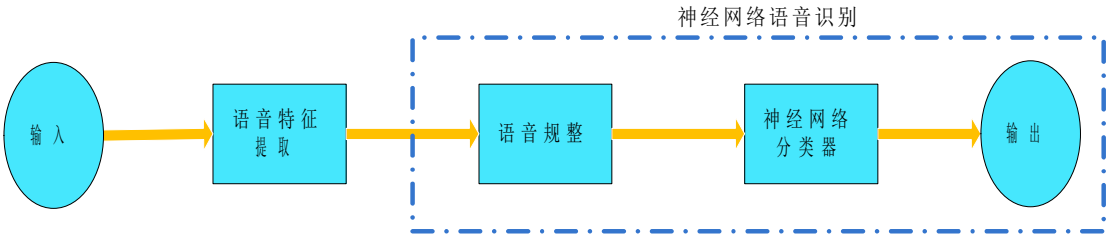


图 12 基于神经网络的语音识别

因为语音信号具有很强的随机性，为了便于神经网络分类器对不同语音的特征进行分类识别，需要对不同的语音提取相同数目特征矢量。

文献[7]中提供的语音规整的方法为是：将由语音信号所提取的特征向量，从第一帧到第 N 帧结束，计算每相邻两个帧之间的距离，查找距离最小的两个帧，将其对应的各个系数归一化成一组系数，也就是计算查找参数 $D(k)$：

$$D(k) = \sqrt{\sum_{i=1}^{M} ZH(k,i) - ZH(k+1,i)^2}$$

其中，$D(k)$ 为第 k 帧到第 $k+1$ 帧之间的距离，k 为帧号，M 为滤波器个数，

$ZH(k,i)$为特征向量矩阵的第k行第i列元素

当找到了第$k1$帧和弟$k1+1$帧之间的距离最小时，即按下式将其归整成两帧的平均：

$$ZH(k1,i)=\frac{ZH(k1,i)+ZH(k1+1,i)}{2}$$（其中，$i=1,2,3\cdots M$）.

结合语音规整的算法，首先本文利用 Matlab 编写 guizheng 函数[7]（具体程序见附录 7.6），对实验语音信号做规整处理，取规整后前 8 行特征向量作为输出结果。规整后特征参数值如下为 8 行 24 列的矩阵：

10.9287	-27.5082	-5.5957	20.7279	2.8172	-14.7039	-3.4642	-7.6500	-11.1126	0.4821	1.4543	-0.8897
6.5089	-8.0687	-1.1079	13.0823	-2.8825	-1.2349	2.9571	-3.9108	-8.5013	0.9008	0.8037	-0.4413
4.7474	-17.0782	-2.4588	13.6223	2.3440	-3.3767	3.7078	-4.6214	-7.4810	0.6488	1.2391	-0.6824
0.7194	14.5812	8.0984	-0.2886	9.4339	8.9724	2.1373	1.8200	4.0777	0.6576	1.3402	0.3353
6.3939	-4.4008	5.7664	13.2585	12.2587	4.2684	3.5594	-1.3319	-3.3443	0.5190	2.3998	-0.4326
9.0573	23.9661	15.2992	1.3478	13.4373	7.1815	-4.4588	1.6662	-0.1373	0.0004	2.5733	0.8344
11.4467	4.4267	9.7630	14.5928	13.4623	5.5464	2.8207	-3.4750	-5.8843	0.4035	3.2132	0.0227
6.4838	14.0629	18.9339	-0.1477	3.2511	1.4494	-5.3889	6.0082	-2.5120	-0.1705	1.3683	0.1580
11.0415	5.2499	18.9882	12.3514	11.3864	2.9635	-0.3405	2.5416	-7.0242	0.8938	3.3919	-0.1992
-8.1637	-4.3791	12.2235	-11.4385	-9.7034	-0.2974	3.8989	24.0206	3.0020	1.5539	-2.0613	-0.9920
0.7950	-0.6888	22.7813	6.1476	7.4672	1.2807	9.6854	19.7158	-2.5218	1.2121	1.3463	-1.4083
-12.3058	-16.6024	-1.1556	-26.0227	-21.0351	12.2561	2.3100	31.1588	6.6065	4.0805	-5.2421	-0.5402
-0.7404	-5.1341	17.9579	-8.2207	-2.7411	10.1118	3.2063	30.4697	-0.9281	3.8011	-0.7007	-0.8042
-9.2579	-12.3163	-8.8146	-26.3730	-18.1551	13.4581	3.6701	14.1206	5.4663	3.3255	-4.8454	0.4390
-0.8366	-12.0963	15.2941	-16.3124	-10.5436	17.9421	0.4746	32.5788	-0.9355	5.4776	-2.2836	-0.4781
-5.1605	-17.2410	-7.6056	-6.3285	-13.2630	11.1162	1.3383	-0.7667	4.6716	3.1528	-3.5996	0.8221

然后，对待识别语音信号做语音规整，得到相应的特征参数矩阵；

最后，通过 Matlab 编写 bp 函数（具体程序见附录 7.7），对规整后的语音信息参数值做模式识别。

4. 3 问题二的模型建立与求解

理解问题描述我们得到：手机运营商想利用语音机器人作为客服，处理查询话费余额、查询套餐余量、查询最新优惠活动等，只要用户录制一段语音发送给手机运营商的客服机器人，机器人通过语音的内容完成应答。其中不需要考虑断句以及返回给用户的形式。

本文利用软件工程中“系统”的概念，结合上述描述，提出“语音识别系统”。并将上面的描述作为“语音识别系统”的需求分析，结合 4.2 中语音识别的各个环节，我们将端点检测、特征提取、语音识别作为该系统的三个子模块。利用软件工程中面向对象(OOD)的分析方法，用户是实体，三个子模块是服务，用 UML 画出该系统的顺序图，如图 13 所示。

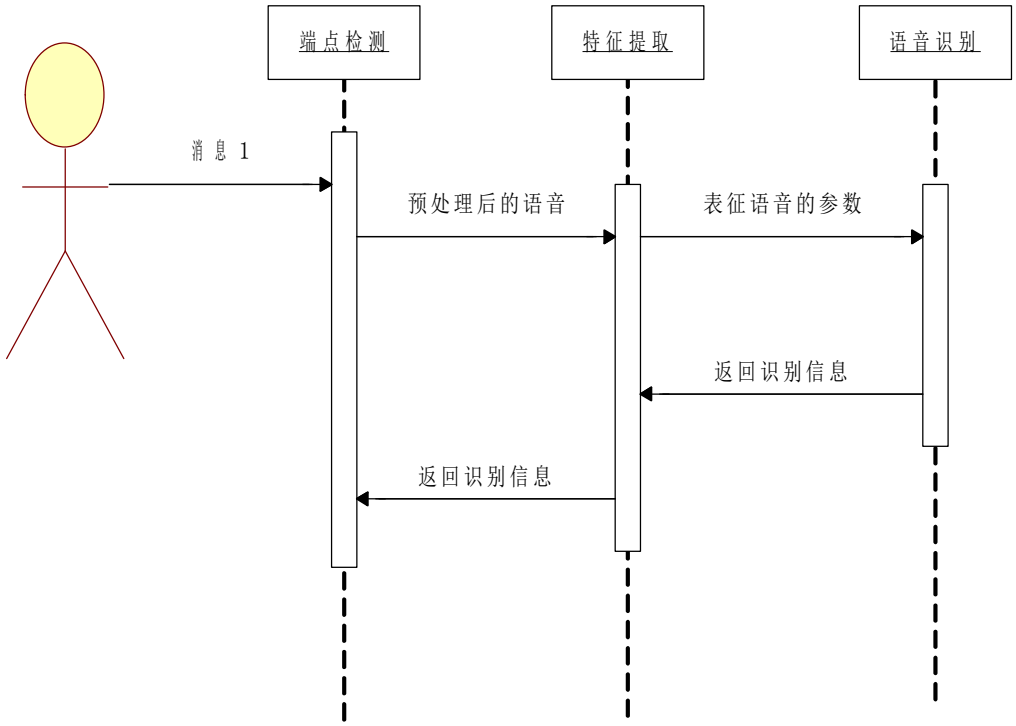


图 13 “语音识别系统”顺序图

结合软件工程中面向对象(OOD)的分析方法，本文对“语音识别系统”建立动态模型。动态模型[9]建立过程简化如下：

Step1：关系类设计。

根据前面系统的需求分析描述，我们设计系统用例如下：

表 3 “语音识别系统”系统用例

参与者	用例名称	用例说明
机器人（Robort）	UC01 端点检测	对用户语音预处理
	UC02 特征提取	提取用户语音特征参数
	UC03 语音识别	与系统内语音进行模式匹配

在解决本问题的过程中，不需要边界类和控制类。因此，在这里我们省略了系统边界类和控制类的设计过程。

Step2：用例描述

针对需求描述和 4.3 中语音识别各个环节的分析过程，本文对“语音识别系统”做以下描述。

具体用例描述，如表 4 所示。

表 4 语音识别用例描述

用例名称：语音识别	执行者：机器人
1.1 前置条件：用户语音特征参数已存入系统数据库中。	
1.2 后置条件： 如果此用例执行成功，返回用户需求信息；如果执行不成功，系统状态不变。	
1.3 主事件流	
1）当机器人接收到用户语音的特征参数时，此用例开始； 2）将用户语音特征参数与系统语音参数逐一匹配。（E-1） 3）系统返回语音识别信息。	
1.4 备选事件流	
E-1：若语音匹配不成功，返回失败信息。	

Step3：根据用例描述，绘制顺序图。

根据上面的分析方法，并参考用户操作手册编写规范[10]。本文为手机运营商指定的用户操作规则如下：

表 5 用户操作规则

使用者	拥有该手机运营商业务的手机用户
操作要求	用户录制的语音能正常分析，噪声不大
操作规则	1 用户登录微信 2 通过微信录制一段清晰连续的需求录音。 如果是查询话费余额，录音内容应为“查询话费余额”或者数字“1”的读音； 如果是套餐余量查询，录音内容应为“查询套餐余量”或者数字“2”的读音； 如果是最新优惠活动查询，录音内容应为“查询优惠”或者数字“3”的读音； 如果是其他业务，录音内容为“其他”或者数字“4”的读音； 3 选择接收方 4 点击发送给客服
备选事物	如果用户录音内容与操作规则中差别较大，系统会自动回复“操作有误”等信息。

4.4 问题三的模型建立与求解

根据问题 3 的分析，并结合 4.3 中制定的用户操作规则，本文设计如下实验以验证语音识别模型。

1．实验设计

设计实验如下：分别录制 10 段语音将其中两段作为标准样本，另外八段语音作为测试样本，语音内容为“查询话费余额”、“1”、“这是什么”、“数学建模”；

然后对测试样本的语音信号进行波形分析、端点检测、MFCC 参数提取；最后将测试样本与标准样本惊醒模式匹配，并分析实验结果。

本实验中，实验编号与实验样本对应，如表 6 所示。

表 6 测试语音样本

实验编号	实验内容
1 2 3 4	查询话费余额（标准） 查询话费余额（测试 1） 查询话费余额（测试 2） 查询话费余额（测试 3）
5 6 7 8	1（标准） 1（测试 1） 1（测试 2 1（测试 3）
9 10	这是什么（测试 1） 数学建模（测试 2）

2．实验过程及模型验证

结合 4.2 中的建模过程，本文从以下几个步骤进行实验验证。

Step1：根据语音样本的波形初步分析；

首先通过 Matlab 软件中 wavrecord 函数录制测试样本，然后绘制各个样本的波形。样本波形如图 14、图 15、图 16 所示。

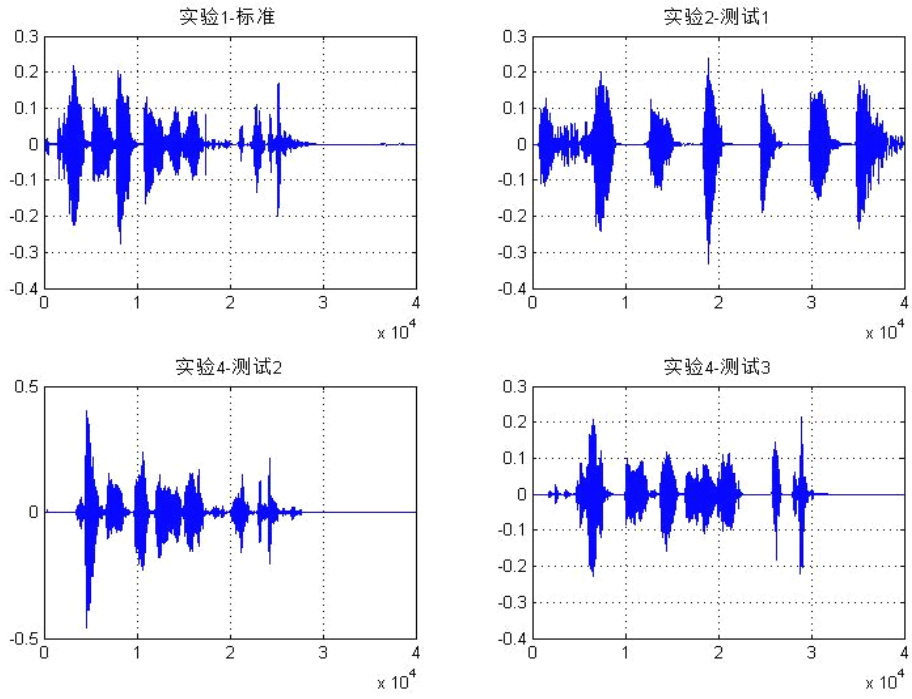


图 14 “查询话费余额”样本波形

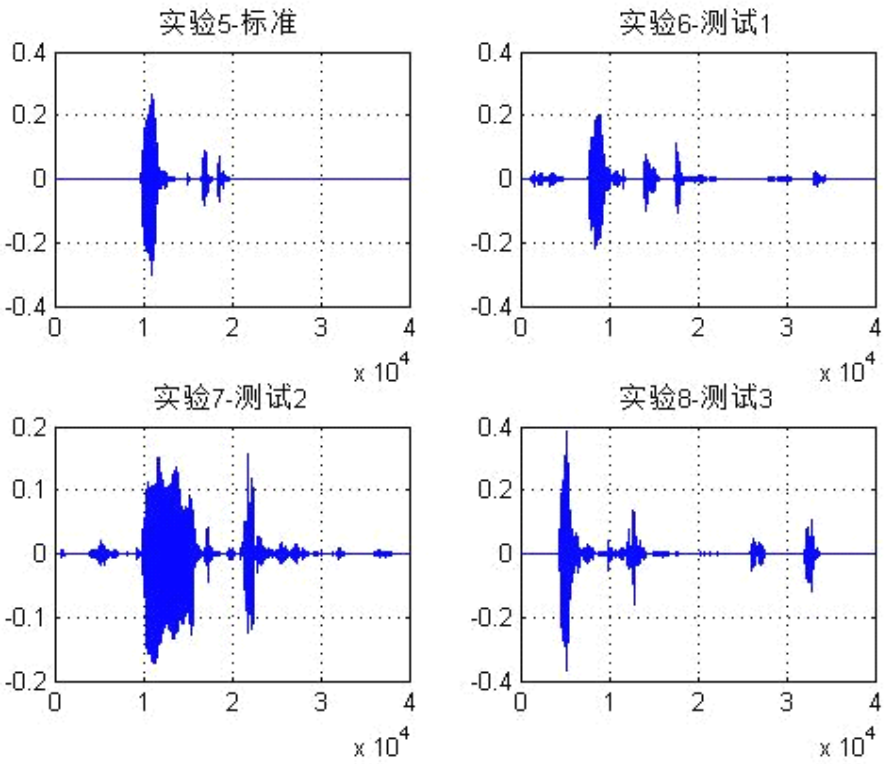


图 14 “1”样本波形

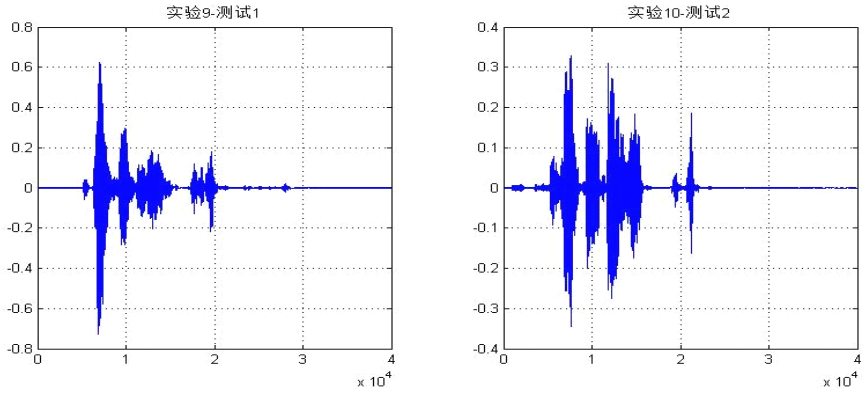
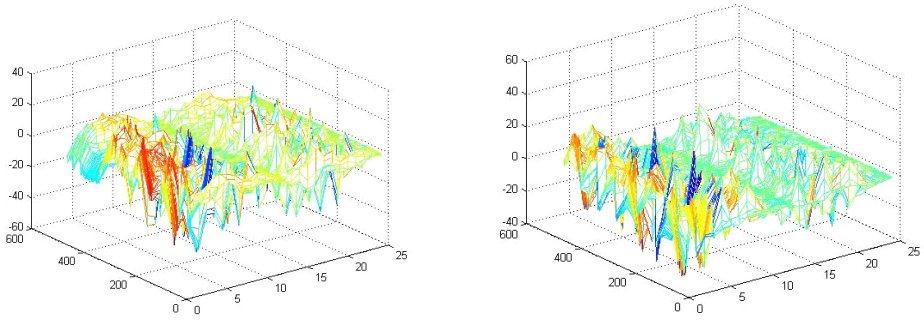


图 15 其他语音波形图

Step2：分别对语音样本进行端点检测，并提取其 MFCC 特征参数。



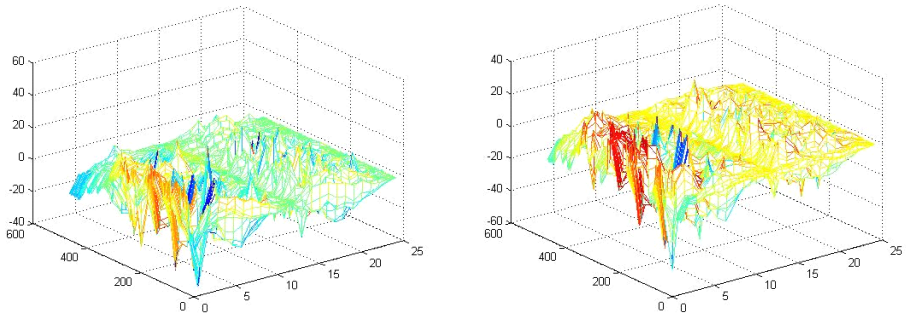


图 16 MFCC 参数

以上分别是第一部分实验中样本 1、样本 2、样本 3、样本 4 的 MFCC 参数。从图像变化态势及我们可以初步分析样本 4 与样本 1 匹配。

Step3：利用神经网络模型进行模式匹配，并分析语音识别模型正确率。设模型识别正确率用 r 表示、待测养样本总数为 n、语音样本匹配正确数为 m，那么：

$$r = \frac{m}{n} \times 100\%$$

在这里我们利用 4.3 中由 Matlab 编写的 bp 函数（具体程序见附录 7.7），对测试样本进行语音识别，最终识别结果如下表所示：

表 6 识别结果

测试样本	识别结果	理想结果
2	0	1
3	1	1
4	1	1
6	5	5
7	1	5
8	5	5
9	5	5
10	0	0

由上表可知，模型的识别准确率 $r = 75\%$。

5. 模型的评价及推广

5.1 模型的评价

模型的优点：(1)本文分别对语音识别的各个环节建模分析，对于端点检测采用“双门限检测法”、对于特征提取采取 MFCC 参数表征、对于模式识别采用神经网络算法进行建模，方法明确、思路清晰。 (2)充分利用 Matlab 编程软件，编写了大量的程序对问题进行分析，结果真实准确。(3)使用图表呈现计算结果，一目了然，真实准确。 (4)模型验证阶段设计了试验说明，论文的实践性较强。

模型的缺点：(1)语音模式匹配环节，为了更好地利用神经网络算法，本文对语音特征参数进行规整，舍弃了部分数据，对识别精度对造成影响。(2)对语音特征参数 MFCC 数据的挖掘有待加强。

5.2 模型的推广

人工神经网络(Artificial Neural Networks, ANN)是由大量结构简单的神经元相互连接，模拟人类大脑神经系统处理信息的方式，对输入信息进行并行处理和非线性映射的网络系统。

本文利用神经网络算法解决模式识别问题，所以该算法可以应用到关于识别的问题当中，比如语音特征信号的分类，人脸朝向识别等问题。

6. 参考文献

[1] 付丽辉，语音识别关键性技术的 Matlab 仿真实现，仪器仪表用户，17(3)：31 至 33，2010.
[2] 张震宇，基于 Matlab 的语音端点检测实验研究，浙江科技学院学报，19(3)：197 至 201，2007.
[3] 张仁志，基于短时能量的语音端点检测算法研究，语音技术，34(6)：52 至 54，2005.
[4] 赵力，语音信号处理，北京：机械工业出版社，2003
[5] 张晶，范明，冯文全，董金明，基于 MFCC 参数的说话人特征提取算法的改进，语音技术，32(7)：76 至 78，2001.
[6] 王让定，柴佩琪，语音倒谱特征的研究，计算机工程，29(13)：31 至 33，2003.
[7] 詹新明，杨灿，基于 Matlab 和 BP 网络的语音识别系统，微计算机信息，25(9)：176 至 178，2009.
[8] 何强，何英，Matlab 扩展编程，北京：清华大学出版社，2002.
[9] 郑人杰，马素霞，殷人民，软件工程概论，北京:机械工业出版社，2009.
[10] 用户手册编写规范，百度文库，http://wenku.baidu.com/view/81fc4d61ddccda38376bafe4.html

7. 附录

7.1 语音信息采集 Matlab 源程序

```
%语音信号采集
n=5*8000; %采样总点数 40000，采样频率 8KHz，单声道，16 位采样精度
fs=8000;
ch=1;
y=wavrecord(n,fs,ch,'double')
plot(y)；%语音信号波形
wavwrite(y,'实验 1.wav')     %把分析样本保存为"实验 1.wav"文件
```

```
%语音信号波形
[x,fs,nbit]=wavread('实验 1.wav');
plot(x)；
titile('原始语音波形图')
xlabel('time')
ylabel('amplitude')
```

7. 2 语音信息分帧加窗 Matlab 源程序

```
%语音信号加帧
[x,fs,nbit]=wavread('实验 1.wav');
len=200;     %指定帧长
inc=100;     %指定帧移
y=enframe(x,len,inc);     %分帧函数，x 为输入语音信号
figure;
subplot(2,1,1),plot(x)
title('语音信号分帧前')
xlabel('time')
ylabel('amplitude')
grid
subplot(2,1,2),plot(y)
title('语音信号分帧后')
xlabel('time')
ylabel('amplitude')
grid
%利用 window 函数设计窗口为 120 的汉明窗
N=120;     %窗口长度 120
w = window('hamming',N);
wvtool(w)          %利用 wvtool 函数观察其时域波形图及频谱特性
%语音信号分帧加窗
[x,fs,nbits]=wavread('实验 1.wav');
x1=enframe(x,200,100);
x2=enframe(x,hamming(200),100);
subplot(2,1,1),plot(x1)
title('分帧后未添加汉明窗波形图')
grid
subplot(2,1,2),plot(x2)
title('分帧且添加汉明窗后波形图')
grid
```

7. 3 语音信息预加重 Matlab 源程序

```
%语音信号预加重
[x,fs,nbit]=wavread('实验 1.wav');
len=200;
inc=100;
```

```
y=enframe(x,len,inc);
%语音信号通过一个高通滤波器 1-0.935^(-1)
z=filter([1-0.9375],1,y);
figure(2)
subplot(2,1,1),plot(y)        %加重前
title('语音信号预加重前');
grid
subplot(2,1,2),plot(z)        %加重后
title('语音信号预加重后');
grid
```

7. 4 语音信息端点检测 Matlab 源程序

```
%语音信号端点检测
%读取文件名为实验 1 的 wav 文件
[y,fs,nbits]=wavread('实验 1.wav');
%设置参数
FrameLen=200;       %帧长 200
FrameInc=100;        %帧移 100
amp1=10;     %初始短时能量高限制
amp2=2;        %初始短时能量最低限制
zcr1=10;        %初时过零率最高限制
zcr2=5;          %初始过零率最低限制
maxsilence=8;
%语音信息段中最大的静音长度；
%如果语音信息段中的静音帧数没有超过最大静音长度，那么我们认为语音还没有结束；
%如果超过了改值，那么对语音信息长度 count 进行判断，如果 count<minlen,那么认为前面的语音段为%噪声，舍弃，转向静音状态 0；若 count>minlen,那么认为语音段结束了。
minlen=15;%语音段的最短长度，若语音段长度小于此值，则认为其为一段噪音
status=0;%初始化语音段状态为 0；
count=0;%初始化语音长度为 0；
silence=0;%初始化静音段长度为 0；

%计算过零率
tmp1=enframe(y(1:end-1),FrameLen,FrameInc);     %对语音信号分帧处理
tmp2=enframe(y(2:end),FrameLen,FrameInc);
signs=(tmp1.*tmp2)<0;   %过零率公式
diffs=(tmp1 -tmp2)>0.02;
zcr=sum(signs.*diffs, 2);

%计算短时能量
amp=sum(abs(enframe(filter([1  -0.9375],1,y),FrameLen,FrameInc)),2);       %先
```

预加重处理，通过高通滤波器，然后进一步计算。

```
    %调整能量门限
    amp1 = min(amp1, max(amp)/4);
    amp2 = min(amp2, max(amp)/8);
    %进行端点检测
    y1 = 0;
    y2 = 0;
    for n=1:length(zcr)
        goto = 0;
        switch status
         case {0,1}                                  % 0=静音, =可能开始
            if amp(n) > amp1                  %  确信进入语音段
                x1 = max(n-count-1,1);
                status  = 2;
                silence = 0;
                count   = count + 1;
            elseif amp(n) > amp2 | ... %  可能处于语音段
                   zcr(n) > zcr2
                status = 1;
                count  = count + 1;
            else                                      %  静音状态
                status  = 0;
                count   = 0;
            end
         case 2,                                     % 2 =  语音段
            if amp(n) > amp2 | ...         %  保持在语音段
                zcr(n) > zcr2
                count = count + 1;
            else                                      %  语音将结束
                silence = silence+1;
                if silence < maxsilence %  静音还不够长，尚未结束
                    count  = count + 1;
                elseif count < minlen      %  语音长度太短，认为是噪声
                    status  = 0;
                    silence = 0;
                    count   = 0;
                else                                  %  语音结束
                    status  = 3;
                end
            end
         case 3,
            break;
        end
    end
```

```
count = count-silence/2;
y2 = y1 + count -1;
figure(5)
subplot(311)
plot(y)
grid

axis([1 length(y) -1 1])        %调整坐标
ylabel('信号波形');
line([y1*FrameInc y1*FrameInc], [-1 1], 'Color', 'red');
line([y2*FrameInc y2*FrameInc], [-1 1], 'Color', 'red');
subplot(312)
plot(amp);
grid

axis([1 length(amp) 0 max(amp)])        %调整坐标
ylabel('短时能量');
line([y1 y1], [min(amp),max(amp)], 'Color', 'red');
line([y2 y2], [min(amp),max(amp)], 'Color', 'red');
subplot(313)
plot(zcr);
grid

axis([1 length(zcr) 0 max(zcr)])        %调整坐标
ylabel('过零率');
line([y1 y1], [min(zcr),max(zcr)], 'Color', 'red');
line([y2 y2], [min(zcr),max(zcr)], 'Color', 'red');
```

7.5 语音信息特征（MFCC 参数）提取 Matlab 源程序

```
%特征提取 MFCC 参数
function ccc=mfcc(x)
%归一化 mel 滤波器组系数
bank=melbankm(24,256,8000);
bank=full(bank);
bank=bank/max(bank(:));

%DCT
for k=1:12
    n=0:23;
    dctcoef(k,:)=cos((2*n+1)*k*pi/(2*24));
end
%归一化倒谱提升窗口
w=1+6*sin(pi*[1:12]./12);
w=w/max(w);
```

```
%预加重滤波器
xx=double(x);
xx=filter([1 -0.9375],1,xx);
%语音信号分帧
xx=enframe(xx,256,80);
%计算每帧的 MFCC 参数
for i=1:size(xx,1)
    y=xx(i,:);
    s=y'.*hamming(256);
    t=abs(fft(s));
    t=t.^2;
    c1=dctcoef*log(bank*t(1:129));
    c2=c1.*w';
    m(i,:)=c2';
end
%差分参数
dtm=zeros(size(m));
for i=3:size(m,1)-2;
    dtm(i,:)=-2*m(i-2,:)-m(i-1,:)+m(i+1,:)+2*m(i+2,:);
end;
dtm=dtm/3;
%合并 mfcc 参数和一阶差分 mfcc 参数
ccc=[m dtm];
ccc=ccc(3:size(m,1)-2,:);
subplot(2,1,1)
ccc_1=ccc(:,1);
plot(ccc_1);title('MFCC');
grid;
ylabel('幅值');
[h,w]=size(ccc);
A=size(ccc);
subplot(2,1,2)
plot([1,w],A);
xlabel('维数');
ylabel('幅值');
title('维数于幅值的关系')
grid
mesh(ccc);
xlabel('倒谱系数维数');
ylabel('语音分析帧数');
zlabel('倒谱值')
grid on
```

7. 6 语音规整 Matlab 源程序

```
%语音规整
function cc=guizheng(x)
n=length(x(:,1));   %求 mfcc 的参数长度
m=length(x(1,:))   %列数为滤波器组的个数
M=8;     %要求规整后的帧数
for i=0:n-M-1
      for k=1:n-i-1
            for j=1:m
                  d(k)=sqrt((x(k,j)-x(k+1,j))^2);
            end
            [c,w]=min(d);
      end
      for k=1:n-i-1
          if k<w
              x(k,:)=x(k,:)
          else if     k==w
              x(k,:)=(x(k,:)+x(k+1,:))/2;
          else
              x(k,:)=x(k+1,:)
          end
       end
end
end
cc=x(1:8,:)
```

7. 7 语音识别 bp. Matlab 源程序

```
function bp(x);
p=x;
t=?;     %期望输出矩阵（根据样本情况而定）
net=newff(minmax(p),[3 2],{'tansig','purelin'},'traingd');
net.trainParam.show=50;   %设置参数
net.trainParam.lr=0.05;
net.trainParam.epochs=300;
[net,tr]=train(net,p,t);
y=x(:,1)
a=sim(net,y)
```