摘 要

语音识别与处理技术在信息技术的人机接口中得到普遍关注，它在电子产品中的应用使得人们的生活变得更加的精彩。通过语音命令，人们就能控制系统设备让其响应语音指令的相应动作。这种具备了语音识别功能的系统在互联网、通信、军事、国防等方面具有十分重要的价值。语音识别技术应用于车载平台上，它能使对小车的驾驶显得更加的灵活简单，也更加的安全与舒适。

本文研究基于特定人小词汇量的 DTW 模型与算法的语音识别技术。介绍了语音识别的基本方法，并在传统 DTW 算法的基础上对语音识别算法进行了改进与优化。本文采用可变窗长和双门限相结合的方法来进行语音端点检测。在进行最优路径选择中，本文采取了松弛起点与终点的办法来选取最优匹配路径。通过 MATLAB 的仿真结果可以看出改进后的 DTW 算法识别结果明显优化传统 DTW 算法的识别结果。全文首先是对语音识别技术的基本原理作出了介绍与分析。对于特定人孤立词小词汇量语音识别系统，本文选用 DTW 算法进行语音识别。在确定选用 DTW 算法后，本文就开始对 DTW 算法进行改进与优化。并将改进后的 DTW 算法与传统的 DTW 算法进行对比，通过仿真结果的比较我们可以看出优化后的算法优于传统算法。在进行端点检测的过程中，本文首先将分帧处理后的语音信号划分为静音段、过渡段和语音段。然后对静音段、过渡段、语音段分别取不同的窗长来进行处理。在静音段本文选用较长的窗长进行处理，对于语音过渡段我们取较小的窗长与帧移，在语音段，我们就取常规窗，这样既不会影响语音识别系统的处理速度，又能够较准确的达到端点检测的目的。在进行变窗长处理的同时本文还结合双门限端点检测的方法来进行语音信号的端点检测。在具体的 DTW 算法实现的过程中，本文利用了动态规整技术与松弛端点的方法来选取最优匹配路径。在具体的硬件实现中，本文采用了最小系统与最高性价比的方案来实现语音识别功能。语音识别模块完全采用自制的程序，而且在对小车的控制方面，本文采用了划分频段发送波形的方法来控制小车响应不同的动作。针对此语音识别系统，本文提出了需要改进的地方。最后本文对全文工作做了总结，并对语音识别的未来提出了展望。

关键字：语音识别；DTW 模型；车载语音；端点检测

Abstract

Speech recognition and processing in human-machine interfaces technology is widespread concerned. It's application makes people's lives more convenient. People can operate the device only by the command of the voice. The device which is a voice-recognition system on the Internet, communications, military, national defense and etc is of very important value. Similarly, the speech recognition technology in the platform of vehicle, it must make the driving is more flexible, more security and comfortable.

This paper is a paper based on the DTW model of speech recognition technology and it introduces the basic methods of speech recognition. That the speech recognition is improved and optimized is to be applied to vehicle simulation systems. This paper is focusing on the optimization-based endpoint detection, combining with variable window length, two-voice activity detection threshold and then making optimal path selection, taking a relaxing way to the begin and the end, so speech recognition will be more accurate. The simulation and experiment can be seen that these methods could improve the accuracy of speech recognition. And tentative on the fuzzy algorithm is applied to speech recognition model. Specifically, first, the processing of DTW model based on speech recognition including how to remove the noise, the speech feature parameter's extraction and the inter-transform of the signal between frequency domain and time domain and the basic theory of speech recognition, this paper makes the introduction and analysis to that. While determining the isolated word speech recognition applications and the DTW model, it has been improved and optimized the recognition algorithm and it realizes the system. The simulation will be improved compared the optimal algorithm speech recognition with the previous traditional method. This paper analyzed how the voice recognition algorithm improved. When at the frame processing, quiet segments, voice segments and transient segments get the different windows to process. In the quiet section of the signals of voice, we can use longer window length came to pick up the frame. In the transition section of the voice, we can use shorter window length to pick up the

frame. In the voice section, we can use regular window length to pick up the frame. At the same time we also could use a double threshold method for endpoint detection, which combines short-time average energy and the short-term zero crossing rate, we take the low threshold and high threshold to limit the value of the starting point and end point. In the specific process of DTW algorithm, using dynamic warping and relaxation methods in the endpoint, selecting the optimal path, so it can get a more accurate voice match. In specific applications, we combine the system with the highest minimum cost of the program to achieve it. In the entire application of the algorithm, this paper completes the implementation. Speech recognition process is completely self-made program, in the control of the car, the method of sending wave band division used to deal with the different car actions. This paper proposes the point which is needed to improve in speech recognition and presents the future prospect of speech recognition.

Keywords: Speech Recognition; DTW Model; Audio Of Car; Endpoint Detection

独 创 性 声 明

本人声明，所呈交的论文是本人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得武汉理工大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

签 名：黄向林 日 期：2010.5.27

学位论文使用授权书

本人完全了解武汉理工大学有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权武汉理工大学可以将本学位论文的全部内容编入有关数据库进行检索，可以采用影印、缩印或其他复制手段保存或汇编本学位论文。同时授权经武汉理工大学认可的国家有关机构或论文数据库使用或收录本学位论文，并向社会公众提供信息服务。

（保密的论文在解密后应遵守此规定）

研究生（签名）：黄向林 导师（签名）：[illegible] 日期 2010.5.27

第1章 绪 论

1.1 语音识别的研究概况

语音识别技术的研究开始于上个世纪的50年代，自AT&Bell实验室研制成功的第一个可以用来识别仅10个英文数字的语音识别系统(Audry系统)以来，语音识别技术才真正走上轨道。Audry系统主要通过测量数字元音区域的共振波谱来进行识别语音。它是一个针对特定人的离散数字识别系统。

20世纪60年代计算机开始在实际研究中得到应用，这也促使了语音识别技术得以快速地发展。这一时期出现了线性预测分析(LP Linear prediction)和动态规划(DP Dynamic programming)等在语音识别方面的几种比较重要的思想。在这两种思想之中线性预测分析技术能较好地解决语音信号产生模型的问题，而动态规划则有效解决了不等长语音信号的匹配问题。这些重要的思想给以后语音识别技术的发展奠定了基础[1]。同时Bell实验室又提出了基于模式匹配和概率统计的方法来进行语音识别的思想，这种新的思想给语音识别开辟了新的道路，给语音识别技术的发展带来了更加深远的影响。

20世纪70年代，伴随着在微电子技术方面的发展与研究，语音识别又有了新的进展。特别是在具体的应用上，语音识别技术开始成功地应用到电子产品中。这标志着语音识别技术已经能够走出实验室应用到实际的生活中。由于微电子技术与语音识别技术的完美结合以及市场对语音电子产品的需求，使得语音识别方面的成果接连不断。具体表现为：在理论上，线性预测分析技术得到了进一步的发展，而且动态时间弯曲(DTW Dynamic time warping))技术基本也已成熟，特别是提出了矢量量化(VQ Vector Quantization)和隐马尔科夫模型(HMM Hidden Markov Model)的理论，这些新的理论方法解决了当时语音识别技术所面临的困难与问题。同时在实际应用中也实现了基于线性预测倒谱等算法的识别系统。理论与实践的结合使语音识别技术取得更快的发展。

80年代，随着语音识别研究的进一步深入，HMM模型在语音识别中得到了成功的应用。而且在这一阶段人工神经网络(ANN)的提出又将语音识别技术推进到一个全新的发展阶段之中。在AT&Bell实验室研究人员的共同努力下，他们终于把原来HMM的纯数学模型进行了工程化的推广[2]。从DTW到HMM模型的改

变这标志着语音识别算法从模板匹配技术转向基于统计模型技术，而且将小词汇量转入到大词汇量的解决方案中来。语音识别技术朝向更复杂更高端的方向发展。

进入90年代，随着计算机技术的飞速发展与应用以及电信领域的快速发展，这些都加速了多媒体时代的来临。许多发达国家和一些全球知名的大企业都置身于对语音识别系统的研究中。在这一阶段，市场上出现了可以语音拨号的手机、与人对话的智能玩具等等一序列的电子产品。而且在商业服务中，出现了以语音识别、语音合成为核心技术的呼叫中心(call center)、语音门户网站等等。

我国语音识别研究工作始于二十世纪八十年代初，一直紧跟国际水平。在语音识别技术方面的研究，国家做了大量的投入。越来越多的学者都投身到语音识别技术的研究工作中来。而且在国内，基于神经网络的语音识别技术的研究工作早已开始。基于语音识别技术的电子产品也早已在国内市场出现。

综上所述，语音识别技术的研究不仅受到各个国家的重视，更是近几十年来人们一直关注的热点话题。而现在人们更多关注的是语音识别技术在车载这个平台上的应用。

1.2 语音识别系统的构成

经过飞速的发展，语音识别技术已经发展到一个实用性的阶段，已经从实验室走向市场。语音特征矢量提取单元(前端处理)、训练单元、识别单元和后处理单元共同组成了语音识别的主要系统，其系统构成如图 1-1 所示。

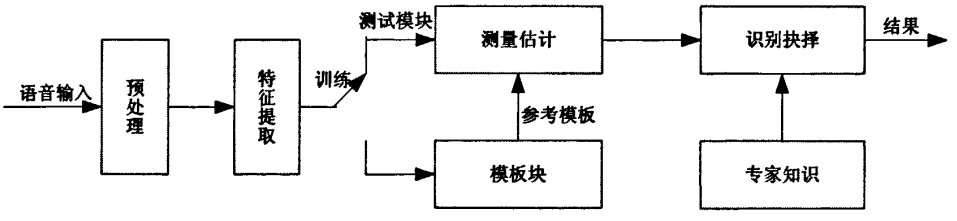


图 1-1　语音识别系统构成图

语音控制汽车是车载语音的一种发展趋势。目前，将语音识别技术应用于汽车的产品大多只有在一些玩具中才能见到，而没有应用到实际的车载平台中其主要是因为考虑到安全性以及各种车载环境因素的原因。由此可想车载语音控制这一领域蕴涵着相当大的潜在市场与挑战。语音识别理论已经可以应用到实际

阶段了，但目前语音识别技术应用到车载系统中还不够完善，存在着一些问题[3]。但最终语音识别将会成功应用于各个领域中。语音识别技术采用语音命令作为人机接口，通过说话来控制各项功能。目前比较多的是特定人语音识别，其工作原理是先需要事先进行录音，然后将录制语音作为参考模板，将待识别语音信号与参考模板语音进行匹配计算，从而找出最佳匹配结果来进行语音的识别。现在，非特定人语音识别技术的应用也正开始逐步扩大。那么具体语音识别系统是如何通过其构成部分来进行语音识别功能的呢？首先语音信号经麦克风转换成电信号，然后加在输入端，它首先经过预处理，也就包括语音信号的预加重、加窗和端点检测等。经过预处理之后，提起语音信号的特征参数，然后训练形成语音模板，然后对待识别语音同样经过预处理、特征参数提取之后与语音参考模板库进行匹配，得到识别结果。而语音识别技术应用于车载系统中能发挥其独特的优势以及在车载这个平台上能得到完美的表现。

1.3 语音识别技术在汽车上的应用

随着汽车产业的发展和汽车的普及，人们对汽车的安全性、便利性和舒适性都提出了更高的要求。汽车上所添加的功能也是越来越多，而且越来越智能化、越来越便于使用，这些都归功于汽车电子在车载这个大的平台上发挥着其独特的作用。电子产品在汽车上的应用可谓是无处不及，而这些都推动着汽车电子的发展，也为车载系统提出了更高的要求[4]。车载语音便是车载系统的重要组成部分。利用语音命令作为人机接口，通过说话即可控制车载系统的各项功能。语音识别在车载系统上的实现使得驾驶员无需双手和双眼的严密配合而只需要进行语音命令就能控制小车，这样既提高了驾驶安全性又增添了驾驶的乐趣。而全部操作只需要通过简单的几句话就可以完成，使得车载终端系统的通用性更强，也更加人性化。采用语音命令来控制汽车的相应部件来作出反应，这样既简便而且又能提高系统响应速率，增加驾驶的安全度。就目前语音识别技术在车载系统中的应用而言，语音指令不是很多，所需要训练的语音信号也就无需太多。因为只需要控制小车的相应的基本动作，而且为了提高系统响应速率以及车载语音系统对说话人语音信号响应的准确度，也应尽量使用小词汇量语音识别[5]。目前在车载语音方面应用得比较多的为特定人语音识别技术，这种识别技术需要事先进行训练录音，然后获取语音模板，这样才能响应特定人的语音指令。而它相对于非特定人语音识别技术在车载语音系统中的应用有一

定的地位和优势，成熟度也相对高些。虽然现在非特定人语音识别技术的应用正在逐渐扩大，准确性也有所提高，相信其在车载系统上的应用也会越来越多，但其理论比较复杂，实现起来比较繁琐。目前，非特定人语音识别技术应用于车载系统当中会有一定的不稳定性，不能起到较好的效果。

现阶段也出现了很多车载语音产品如免提车载GPS系统，司机可以在驾驶室内通过语音来控制这个免提车载GPS系统，通过它来对小车定位与导航，从而解决不熟悉路线的问题和提升汽车驾驶的安全性[6]。本文基于特定人的车载语音识别系统是针对小车主人先通过语音训练获取小车主人的语音特征参数，然后进行特定人识别，进行模板匹配从而来识别语音指令。本文选用特定人语音识别既能实现车主方便舒适的语音控制，又能给车主提供可靠的安全保障。

1.4 本文研究的内容与思路

课题研究的主要内容是在分析研究各种语音识别算法的基础上，根据系统设计的要求及系统所要实现的功能，选择确定了特定人小词汇量的DTW语音识别系统，利用改进与优化后的DTW算法来实现语音的识别。整个系统就是通过语音指令来控制小车的相应动作。

本论文的结构安排如下：

第一章即为绪论，简要的介绍语音识别技术的研究历程以及语音识别系统的构成。语音识别技术在车载语音系统中的应用以及发展。

第二章论述语音识别的基本原理，介绍语音识别的处理过程及原理。

第三章探讨语音识别算法的改进与实现。重点介绍语音识别中对DTW算法的改进与优化，并将改进的DTW算法进行了实现。

第四章主要是对特定人小词汇量语音识别系统的硬件系统与软件的设计与实现。

第五章是对本论文的一个总结，概括了在论文撰写的学习过程中所做的工作、收获和体会以及对以后所要开展工作的一个展望。

第 2 章　语音识别技术的分析

2.1 语音信号的预处理

对语音信号的预处理主要包括对其声音的预加重，分帧处理和窗化处理。

a) 语音信号的预加重

采用预加重方法处理语音信号能补偿语音信号的固有衰落，而且能有效地消除唇辐射的影响[7]。预加重时所需的传递函数为：

$$H(z) = 1 - 0.94z^{-1} \tag{2-1}$$

若假设$S(n)$为语音输入信号，经过预加重后得到的信号为：

$$\tilde{S}(n) = S(n) - 0.94S(n-1) \tag{2-2}$$

$\tilde{S}(n)$就是经过预加重后得到的信号。

b) 分帧处理

要将时域信号变为频率信号，而且将模拟信号进行数字化处理。那么首先就要将语音信号作分帧处理。由于在一般情况下，语音信号在 10~20ms 内是相对稳定的[8]。那么得到的分帧处理公式如下式 2-3 所示。

$$x_l(n) = \tilde{s}(Ml + n), \quad n = 0,1,\cdots\cdots,N-1, \quad l = 0,1,\cdots\cdots,L-1. \tag{2-3}$$

c) 窗化处理

因为要获取语音特征，滤出噪声信号，需要进行窗化处理。在所有的窗化处理的方法中，汉明窗的旁瓣最低，而且它具有更平滑的低通特性。为了在语音处理的过程中能有效地克服泄漏现象，我们采用汉明窗来进行窗化处理[9]。其公式即如下 2-7 所示。

$$\tilde{x}_l(n) = x_l(n)w(n), \quad 0 \le n \le N-1. \tag{2-4}$$

式中：

$$w(n) = 0.54 - 0.46\cos(\frac{2\pi n}{N-1}), \quad 0 \le n \le N-1. \tag{2-5}$$

2.2 语音信号的端点检测

对于硬件系统的语音采集口来说它需要实时的检测有没有语音指令的输

入，而语音指令又不是连续发出的，所以通过语音采集口采集到的声音数据并不全是语音指令信号，其中必定有噪音以及其它我们并不需要的信号以及数据。因此就需要系统一直判断是否有声音指令进入，何时是声音信号。而这种处理过程就是对语音信号进行端点检测。端点检测技术用来确定声音指令音头、音尾的位置。确定语音信号的起止点能更好的对语音信号进行识别，从而提高系统识别率和获取到更好的语音特征参数。端点检测常用的方法有短时过零率、短时平均能量、短时平均幅度、短时自相关函数、短时频域处理等几种[10]。但在本文中选择短时过零率和短时平均能量相结合的方法来进行端点检测。

（1）短时平均能量

短时平均能量是具有时域特征的参数。假设 S(n)为加窗后的语音信号，那么第 t 帧语音的短时平均能量可表示为如下式 2-6 所示。

$$Eng(t)=\frac{1}{N}\sum_{n=0}^{N-1}\left|S_t(n)\right| \tag{2-6}$$

其中 N 为窗的宽度，$S_t(n)$为第 t 帧语音信号中第 n 个采样点的信号样值。本文采用将获取到的语音短时平均能量取其对数值的方法，结合短时过零率来进行端点检测，能更加准确的获取到语音信号的端点值。

（2）短时过零率

短时过零率 ZCR (Zero-Crossing-Rate)用式子表示为如下 2-7 所示。

$$Z_n=\sum_{m=-\infty}^{\infty}Sgn[x(m)]-Sgn[x(m-1)]\cdot W(n-m) \tag{2-7}$$

其中：

$$\begin{cases} Sgn[x(n)]=1 & x(n)>NoiseMax \quad (NoiseMax\text{为噪声上限}) \\ Sgn[x(n)]=-1 & x(n)<NoiseMin \quad (NoiseMin\text{为噪声下限}) \\ Sgn[x(n)]=0 & otherwise \end{cases} \tag{2-8}$$

$$\begin{cases} W(n)=\dfrac{1}{2N} & 0\le n\le N-1 \quad (N\text{为一帧声音的长度}) \\ W(n)=0 & otherwise \end{cases} \tag{2-9}$$

（3）短时平均幅度

其计算公式如下式 2-10 所示。

$$M_n=\sum_{n=0}^{N-1}\left|x_n(m)\right| \tag{2-10}$$

（4）短时自相关函数

自相关函数是描述一个随机信号的重要特性。自相关函数在不同的领域，定义不完全相同。在短时处理技术中，短时自相关函数可描述为：

$$R_n(k)=\sum_{m=0}^{N-1-k}x_n(m)x_n(m+k),(0<k<K) \tag{2-11}$$

自相关函数也可以用来判断语音的清音段和浊音段[11]。

（5）短时频域处理

频域处理是语音信号和数字信号处理的一种方法，对语音信号缓慢变化的特点一般进行短时频域处理。如下式2-12所示语音信号第m帧的短时傅立叶变换为：

$$X_n(e^{jw})=\sum_{m=0}^{N-1}x_n(m)e^{-jwm} \tag{2-12}$$

短时频域的变化放映了语音信号的频谱随时间变化的性质。

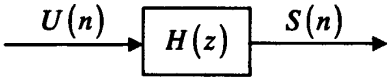
2.3 语音信号的特征参数提取

进行语音识别就是要从语音信号中提取对我们有用的信息，滤出无用的信息，从而获取特征参数来进行语音信号的匹配识别。去除对语音识别无关紧要的冗余信息，提取出对语音识别有用的重要信息这便是语音信号的特征参数提取的关键。特征提取是语音识别前端处理的主要任务，如果特征提取得好，以后的模型的设计与语音训练就会变得容易。因此语音识别所需要的特征是既具备能稳定表示语音的特性又有很强的区别性的特征。特征提取就是要获取到好的特征参数，那么如何获取到好的语音特征参数呢？它需要满足以下三方面的要求才能称为一个好的特征提取：（1）能有效的提取语音的信号特征，包括人的声道特征与听觉模型；（2）参数之间具有良好的独立性；（3）特征参数有比较高效的计算方法。就目前最常用的两种特征参数是线性预测倒谱系数LPCC和Mel倒谱系数MFCC，它们在一定程度上反映了人耳对声音的处理特性。

a) LPCC特征参数的提取

线性预测分析(LPCC)是较为常用的语音特征分析方法之一。它可以有效地解决短时平稳信号的模型化问题。LPCC的基本原理为：语音的每个样值都可以由它过去若干个样值的线性组合来近似，同样也可采用实际语音抽样信号与对它的线性预测值之间的均方差最小的方式来进行逼近，最后解出一组预测系数

[12]。可用如下图2-1的模型来表示。



2-1 信号模型图

U(n)表示模型的输入，S(n)表示模型的输出。模型的系统函数H(z)表示为：

$$H(z)=\frac{G}{1-\sum_{i=1}^{p}a_i z^{-i}} \tag{2-13}$$

式子中a_i是系数，P是预测模型的阶数。

用信号的前P个样本来预测当前样本，定义的方法如下：

$$\tilde{s}(n)=\sum_{k=1}^{p}a_k s(n-k) \tag{2-14}$$

语音信号$\tilde{s}(n)$可由过去的P个样值s(n-k)来预测。式2-14其中的a为加权系数，即LPC系数，P为LPCC预测阶数，预测误差为：

$$e(n)=s(n)-\tilde{s}(n)=s(n)-\sum_{k=1}^{p}a_k s(n-k) \tag{2-15}$$

由此可求其极值，便得到LPCC系数，LPCC系数它记录了语音信号谱的极值点的轨迹，以此LPCC系数来表示语音信号的特征。

b) Mel倒谱系数MFCC

Mel倒谱系数(Mel Frequency Cepstrum Coefficient)是基于人的听觉模型的基础上提出来的。它能形象的描述人类听觉系统对声音频率的感觉，近似计算可以表示为如下式2-16所示。

$$Mel(f)\approx 2595*\lg\left(1+{}^{f}/_{700}\right) \tag{2-16}$$

人耳对不同频率的声波有不同的听觉灵敏度，但从人的听觉灵敏度来看，人会觉得低音掩盖高音比较容易，然而高音掩盖低音就比较困难。在低频处的声音掩蔽的临界带宽比高频处的声音掩蔽的临界带宽要小。当两个频率相近的音调同时发出时，人就只能听到其中频率较低的一个音调，对于这种由于人的主观感觉突变而产生无法区分带宽边界的情况，Mel刻度是对这一临界带宽的度量方法之一。于是在语音识别过程中可以采取从低频到高频这一段频带内按临界带宽的大小由密到稀安排一组带通滤波器的方法来进行语音信号的处理[13]。对输入信号进行滤波，将每个带通滤波器输出的信号能量作为信号的基本特征。所选用的带通滤波器进行滤波的情况如下图2-2所示。

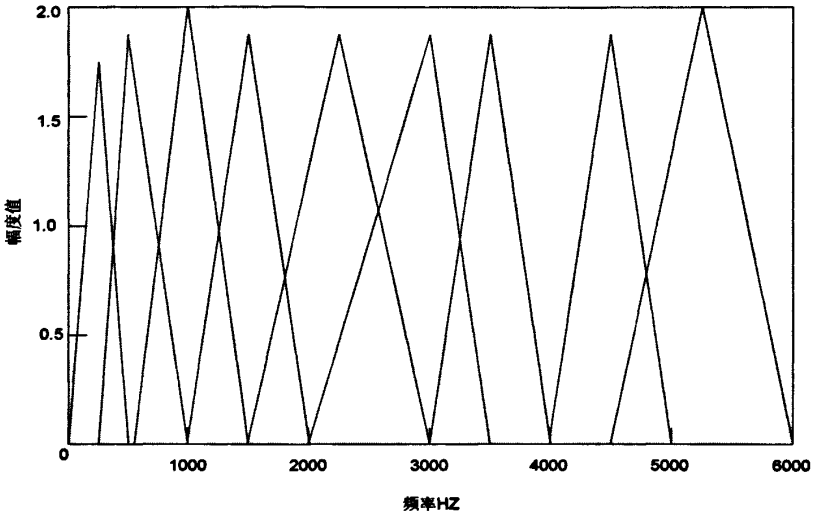


图 2-2　Mel 尺度滤波器组

MFCC特征参数提取的过程为：首先将语音信号进行FFT变换到频域，通过Mel尺度的滤波器阵列后，将经过滤波器阵列输出后的语音信号进行离散余弦变换。具体的参数计算步骤如下：首先将信号进行预加重处理然后假定取t时刻的一帧语音采样信号，帧长为N，即用式子表示为：{x(t)},t=1,2,3.....N，然后确定每一帧的采样点数和帧移。然后进行相应的变换与计算[14]。

（1）加Hamming 窗后作N点快速傅里叶变换（FFT），取到信号幅度谱|x(k)|。

（2）运用2-17此公式将实际频率尺度转换为Mel频率尺度：其中 f_m 为实际频率。

（3）然后可以设置在整个Mel轴上配置L个三角形滤波器，每个三角形滤波器的中心频率C(L) 在Mel轴频率轴上等间隔分配。假设B(L)、C(L)、A(L)分别是第L个三角型滤波器的下限，中心和上限频率，相邻滤波器之间的下限中心和上限频率有如下图2-3的关系。

$$C(L)=A(L-1)+B(L+1) \tag{2-17}$$

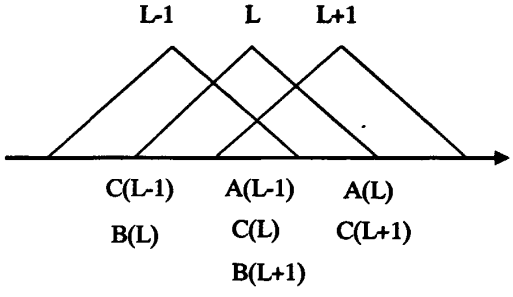


图 2-3　频率相连三角形滤波器的关系

（4）由所得到的语音信号幅度$|x(k)|$可求出每一个三角形滤波器的输出。

$$m(l)=\sum_{k=B(l)}^{A(l)}W_l(k)|X_n(k)|,l=1,2,...,L \tag{2-18}$$

$$W_l(k)=\begin{cases}\dfrac{k-B(l)}{C(l)-B(l)},B(l)\le k\le C(l),C(l)\le k\le A(L)\\ \dfrac{A(l)-k}{A(l)-C(l)}\end{cases} \tag{2-19}$$

（5）对所有三角形滤波器的输出作对数运算，然后再进行离散余弦变换，便可以得到MFCC参数。

$$c_{mfcc}(i)=\sqrt{2/N}\sum_{l=1}^{L}\lg m(l)\cos\left[(l-1/2)\frac{i\pi}{L}\right],i=1,2,...,Q \tag{2-20}$$

其中，Q为MFCC参数的阶数，c(i)为所求的MFCC的参数。

2.4 语音识别的模型与算法

随着语音识别技术的飞速发展和它越来越受人们的关注与重视，各种各样的识别方法也陆续的出现了。但主要的识别技术仍然是基于模板匹配法、HMM模型法、DTW动态时间规划模型法、ANN神经网络模型法。对于语音识别技术来说，这些方法都存在着一些共同点，基本上都有一个相同的基本原理。如下图2-4所示。语音信号经过采样预处理后，进行特征参数提取，得到一组反映该段语音特征的参数模型，然后这些特征参数送入模型库模块进行比较，声音模式匹配模块根据模型库对该段语音进行识别，最后得出识别结果。对于大词汇量，非特定人等情况的语音识别还需要通过语言模型对结果进行进一步的分析处理，最终得到正确的识别结果。

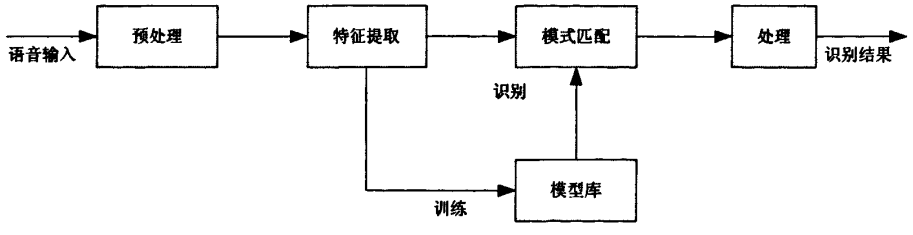


图 2-4　语音识别系统的基本结构

2.4.1 DTW(动态时间规整)

DTW动态时间规整算法其实是把一个语音段内的时变特征变为一致的过程，是一种非线性规整技术。DTW的基本思想是通过将待识别语音信号或者参考模板在时间轴上进行不均匀地拉伸或者弯曲，使其特征与模板特征对齐，并在两者之间不断的进行两个矢量距离最小的匹配路径计算，来获得两个矢量匹配时累积距离最小的规整函数。这种方法是一个将时间规整和距离测度有机结合在一起的非线性规整技术，它能保证待识别语音特征与模板特征之间最大的声学相似特性和最小的时差失真。采用这种方法能成功的解决待识别语音和模板长度不相等的问题。具体用公式来表示则为：首先得利用时间规整函数 $j=w(i)$，此函数的意义即为使测试语音矢量的时间轴i映射到模板语音矢量的时间轴j上。使其特征与模板特征对齐，并在两者之间不断的进行两个矢量距离最小的匹配路径计算，来获得两个矢量匹配时累积距离最小的规整[15]。那么具体用表达式来表示则可表示为：$D=\min_{w(i)}\sum_{i=1}^{M} d\left[T(i),R(w(i))\right]$，其中 $T(i)$ 表示测试语音矢量，$R(w(i))$ 表示测试时间轴的第i帧信号经过时间规整函数后对应于模板语音的矢量，式中 $d\left[T(i),R(w(i))\right]$ 是第i帧测试矢量 $T(i)$ 和第j帧模板矢量 $R(j)$ 之间的距离测度。D则是在最优情况下的两个矢量之间的匹配路径。DTW的搜索路径图如下2-5所示。

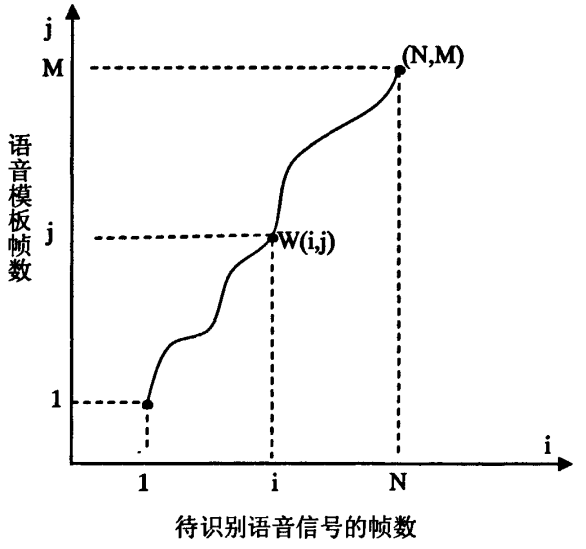


图 2-5　DTW 的搜索路径

DTW一般采用逆向思路，从过程的最后端开始一直到起点来寻找最优路径，

因为这样才能更好的确定一条更佳的路径。DTW算法它一般适用于小词汇量的特定人的孤立词识别系统，采用多模板的训练方法。由于语音的耦合性比较大、训练时又容易产生误差，所以这种方法的鲁棒性不是很好，为了克服这个问题可采用串行训练法，即重复将训练词多说几遍，也就是进行多重复训练，直到找到一致性比较好的特征矢量序列为止，这样就可以得到较好的沿DTW搜索路径的特征矢量序列，然后求这些特征序列的平均来得到模板。总之，DTW也有其优点，其优点是只包含了对要识别词的特征提取，因此训练起来比较简单，而且已经存在有效的硬件方法来实现它，不足之处是对于连续语音它却显得有些无能为力。

2.4.2 HMM(隐马尔可夫模型)

HMM是一个双重随机过程。它的一个随机模型用来表示状态的转移另一个随机模型用来表示状态和观察值之间的统计对应关系。它用概率论的方式来描述时变信号的变化过程。在该模型中，一个状态转移到另外一个状态取决于该状态的统计特性，而某一个状态的观察值也取决于该状态生成语音观察值的概率。因为在观察者的角度只可以看到观察值，而看不到状态，所以叫做隐马尔可夫模型，简写为HMM。隐马尔可夫(HMM)模型是利用概率及统计学理论来解决如何辨识具有不同参数特性的短时平稳信号段以及如何跟踪这些具有不同参数特性的短时平稳信号段它们之间的转化问题的模型[16]。它通过统计与概率论的方法来实现语音识别。就HMM模型来说，一个HMM模型可以由下列参数来决定。

T—观察符号序列的长度。其集合也可以表示为$o=\{o_1,o_2,\cdots,o_T\}$。

M—观察符号数，即每个状态可能输出的观察符号的数目。那么观察符号的集合可表示为$V=\{v_1,v_2,\cdots\cdots v_u\}$。

N—模型中的状态数目。虽然隐马尔可夫模型的状态是不能直接获得的，但这些状态它们彼此之间是相互联系着的，因为任何一个状态都可以由其它的状态来表示或者是转移而来。状态的集合可表示为$s=\{s_1,s_2\cdots\cdots s_N\}$，t时刻的状态表示为$s_t$。

π—初始状态分布。即初始时刻系统处于某个状态的概率。可表示为：

$$\pi=\{\pi_i\},\ \pi_i=p[q_i=s_i],1\le i\le N$$

A—状态转移概率分布。其中元素a_{ij}是指t时刻状态为s_i，而在t+1时刻转移到状

态s_j的概率。它是由状态转移概率构成的一个矩阵，用公式可以表示为：

$$A=\{a_{ij}\},a_{ij}=p[q_{i+1}=s_j|q_i=s_i],1\le i,j\le N$$

B—状态s_j的观测符号概率分布。即它是状态s_j的观测符号概率构成的一个矩阵，元素$b_j(k)$是指状态s_j输出观测符号v_k的概率，t时刻处于状态s_j。其公式为：

$$B=\{b_j(k)\},b_j(k)=p[v_k|q_t=s_j],1\le j\le N,1\le k\le M$$

2.4.3 ANN(人工神经网络)

人工神经网络是近些年来比较新和热门的研究方向。它的原理是由多个非常简单的处理单元彼此按某种方式相互连接而形成的计算机系统，该系统能根据外部输入信息的动态状态来做出相应的响应，它具有实时性和灵活性。人脑若要对某个模式得到正确的模式匹配，就需要进行大量的训练和纠正。训练越多，纠正越多，匹配就会越准确，人工神经网络的识别方法也是如此，它模拟人的大脑，需要通过大量的学习与训练才能投入正确使用，在使用中又不断地进行自我学习从而来更正或者调整信号值。而基于ANN的语音识别系统是由神经元、训练算法及网络结构等要素来构成的。它融合了并行处理机制、非线性信息处理机制和信息分布存贮机制等多方面的现代信息技术[17]。基于人工神经网络的语音识别系统在训练过程中能不断调整自身的参数权值和拓扑结构，以适应环境和系统性能优化的需求。而且在模式识别中也有速度快、识别率高等显著特点而且反应灵敏且能自动适应环境。人工神经网络技术一直是国内外语音识别系统研究的方向和热点。

由于人工神经网络中神经元个数众多以及整个网络存储信息容量的巨大，使得它具有很强的不确定性的信息处理能力。即使是在输入信息不完全、不准确或模糊不清的情况下，神经网络也能够通过获取到的这些不完整信息联想到存在于思维记忆中的一些相关的信息[18]。只要输入到神经网络中的信号模式接近于训练样本的信号模式，神经网络系统就能给出正确的推理结论。人工神经网络能进行自我完善从而改进训练参数，提高精确度。人工神经网络是一种非线性的处理单元，因为对于所有的输入信号，神经元对这些输入信号进行综合处理。它突破了传统的以线性处理为基础的数字电子计算机的局限，这标志着智能信息处理能力和模拟人脑智能行为能力的一大技术飞跃。

2.5 现阶段语音识别所面临的问题

就目前而言，语音识别技术仍然存在着许多有待进一步改进以及优化的地方。由于语音识别一般情况下是对自然语言的识别，那么就必须面临着连续语音的识别，然而连续语音中的因素、音节或单词之间的调音结合引起的音变，使基本模型的边界变的不明确，而且需要建立一个语法与语义的规则来理解它们，这就需要一个优化的系统来解决这些问题。不仅如此，语音识别技术同样面临着外界环境以及噪声等因素的影响，而不能精确的或者很理想的处理语音识别的问题。因为语音信息的信息量大而且变化量也很大，语音模型对于不同的说话者不可能完全一样，因此不同的讲话者所需选择的语音模型还是有差异的，即使是同一讲话者，其语音模式仍然会随时间的改变有所改变。其次语音信号有很大的模糊性，不同的语音听起来虽很相似，但实际则不同，难以区分。而且在强噪声的环境下，语音识别显得尤为困难。这些因素的影响促使语音识别技术有待进一步的改进与完善。而且端点检测的方法仍需进行优化。语音识别系统即使在安静的环境下，系统的识别错误仍然有一半以上来自端点检测。不仅如此，应用于各种环境下，不同的环境情况差别也大，语音识别技术也难以完成准确的识别。例如应用在车载方面，噪音与车载环境对语音识别来说仍然是很大的难题。语音识别技术需要融合多学科知识，如何将多学科知识更好的应用到语音识别系统之中仍然是需要解决的问题[19]。随着科技的进步发展，语音识别技术的研究也需要得到更进一步的深入。

2.6 车载语音识别系统的算法选用

由于车载系统工作环境的影响，车载语音系统就必须具有高抗噪声的功能以及性能稳定等特点。而且现在的汽车电子系统越来越庞大，每个系统构成部分都会影响到彼此的稳定性。这就要求车载语音部分也必须简单而且稳定可靠。这就需要一个比较简单可靠的语音模型和算法来达到车载语音识别系统的要求。在所有的语音模型和算法中，DTW模型以及算法是最古典以及最完善的一套方法与理论，在实际应用中也最简单和易实现。人们往往需要采用这种最简单可靠的方法来实现这样或者那样的一些功能。基于DTW模型与算法的语音识别系统训练起来比较简单，而且已经存在有效的硬件方法来实现。基于DTW模型与算法的语音识别系统应用于车载语音中能让驾驶更加的简单方便，而且语

音识别率也比较高，性能稳定，能满足车载语音识别系统的要求。首先就应用和理论的复杂度来说，DTW要比HMM和ANN简单明了，其原理易于被人们掌握和理解。其次就是DTW算法已经能通过硬件来实现[20]。最后，从车载整体性能方面来考虑，如果在汽车电子系统中加入更多更加复杂的系统或部件，难免会影响到整体性能，整个控制系统控制起来就会比较麻烦，这样各个系统部件之间就会相互的影响，产生安全以及可靠性方面的问题。于是一个简单而且独立性比较好的系统设计就显得尤为重要，因为这样的系统能让各个部分独立的工作，而不是互相产生干扰。这种系统不仅能提高整体性能，而且还能提高安全性。总之，结合语音识别技术在实际中的应用和考虑到性价比以及汽车安全方面的因素，就目前来讲DTW仍然是应用于特定人小词汇量语音识别系统上比较好的模型与算法。

2.7 本章小结

本章对语音识别的方法及基本原理进行了介绍和总结。首先介绍了语音信号的初步处理，对语音的预处理、端点检测、特征参数提取作了详细的介绍与分析。然后对语音识别的模型与算法进行了详细的介绍，DTW、HMM、ANN本文都一一作了介绍。针对语音识别所面临的问题作了分析。最后对于特定人小词汇量语音识别系统的模型与算法的选用，本文就所提到的一些模型与算法作了比较，结合实际与理论阐明了DTW作为此系统的语音模型与算法的优点。

第 3 章　语音识别算法的设计与改进

3.1 传统 DTW 算法的设计

DTW基于动态规整的思想，是语音识别中出现较早、较为经典的一种算法，它解决了发音长短不一的匹配问题。那么传统的DTW算法是如何来实现语音识别的呢？按照第2章所介绍的内容，首先对语音信号进行预处理，然后进行端点检测。在进行完端点检测之后，就需要获取语音信号的特征参数，对于Mel倒谱系数，采用如下的方式来提取。首先根据式$Mel(f)=2595\lg(1+f/700)$，将实际频率尺度转换为Mel频率尺度。然后在Mel频率轴上配置L个三角形滤波器组，而L的个数由信号的截止频率决定。其次再根据语音信号幅度谱求每一个三角形滤波器的输出。最后对所有滤波器输出作对数运算，再进行离散余弦变换就可以得到MFCC系数了。对于程序而言首先要设置好三角形滤波器系数，即归一化滤波器系数[21]。获取到特征参数以后，便可以确定模板，然后就可以对待识别语音进行特征参数提取了。本文所选取的倒谱矢量维数为12，即选用的阶数为12。在参数提取分析过程中，若选择的阶数很大，可将极零点模型用全极点模型来代替，虽然可携带大量的语音信息，但同时也增加了运算量，但阶数增加到一定程度的话又会使语音的内在特征出现很大的随机性，导致识别率降低，经过实验统计一般情况下阶数选择8到14之间的数。通过实验显示，本系统选用阶数p为12时所求出的倒谱特征参数与模板倒谱特征参数具有良好的相似度。Mel滤波器数为24个，DFT长度也为160，帧长仍然为160，帧移为40。在获取到的Mel系数后除首尾两帧，因为这两帧的一阶差分参数为0。然后将所获取到的Mel倒谱系数用MATLAB仿真出来的结果如下图3-2所示，这个语音信号是选取的数字9的发音信号[22]。由于矢量维数为12，得到的矩阵行数其实为88帧，列数为24表示阶数，从图中可以看出也就是横坐标X表示列数，纵坐标Y表示行数，Z轴表示MFCC值。而选择的是三维坐标图，所以获取到的结果如图3-1所示。

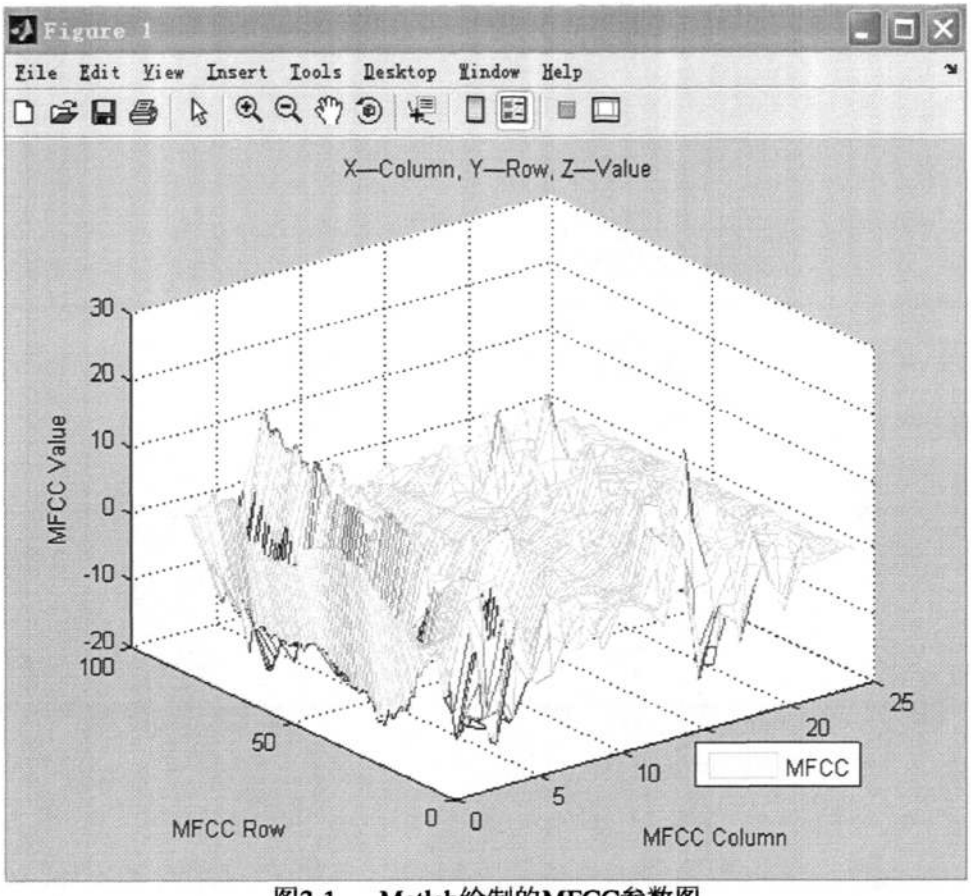


图3-1　Matlab绘制的MFCC参数图

在获得了特征参数后，就可以建立模板将模板的 MFCC 参数存入一指定数组中。在建立所有的参考模板之后，我们对语音模板要做的处理就已经完成，那么接着便要对待识别的语音进行预处理、端点检测、特征参数提取，和前面对参考模板语音所做的处理一样。在获取到的待识别语音的 MFCC 参数后同样要取一指定的数组来存储这些特征参数信息，然后用程序设置一个循环，外循环的次数为所存入的模板数，例如，如果已取了 10 个模板，那么所需要设置的外循环则为 10，然后进行内循环，内循环用来进行对待识别语音每帧分别与模板相应的每帧进行匹配计算。那么接着的问题就是如何进行模式匹配。特征序列可分为两类，对于训练阶段输入的语音进行分析，得到各组特征序列被称为参考模板，记为:

$$R_j = \left\{ r_1^j, r_2^j, \cdots, r_J^j \right\}, j = 1, 2, \cdots, V \tag{3-1}$$

式中，j 为模板对应的命令编号，J 为该命令中的所需要分析的总的帧数，V 为系统模板库中的总模板数，可以等于或大于待识别的命令条数。对识别阶段输入

的语音进行分析，得到的特征序列被称为待测试模板，记为:$T=\{t_1,t_2,\cdots,t_n\}$,n 为输入待识别语音的帧数[23]。这样模板匹配过程就是将参考模板 R 和待测试模板 T 之间进行比较，计算它们之间的相似程度。一般是通过失真度来衡量相似度的，失真越小则相似度越高，那么如何计算失真度呢？可以将模板 R 与测试模板 T 中对应的帧算起，设 n 与 j 分别为 T 和 R 中任意选取的一帧的帧号，用 $D[T(n),R(j)]$ 来表示这两帧之间的特征矢量的失真，这样就可以求出每帧的失真，然后再进行求和从而计算总失真度。用式子表示则为如下式 3-2 所示：

$$D[T,R]=\sum_{n=j=1}^{N}D[T(n),R(j)] \tag{3-2}$$

假设测试语音模板共有 N 帧矢量，而参考模板共有 J 帧矢量，且 $N\neq J$ 。那么动态时间归整就是寻找一个时间归整函数 m = w(n) ，它将测试矢量的时间轴非线性的映射到模板的时间轴上，并使函数满足：

$$D=\min_{w(n)}\sum_{n=1}^{N}d\left[T(n),R(w(n))\right] \tag{3-3}$$

式中，$d\left[T(n),R(w(n))\right]$ 是测试模板 T 的第 n 帧与参考模板的第 j 帧的距离测度。式中的 D 它表示处于最优时间规整情况下两矢量的距离。假设 T 的第 n 帧与 R 的第 j 帧对准，当 N 等于 J 且 T 和 R 完全相同时，w(n)就可以用一条斜率为 1 的线段来表示。那么当 T 和 R 不完全相同时，T 的第 n 帧与 R 的第 j 帧对准，则得到的这些点组成的线便不是一条直线而是一条曲线了，那么这条曲线对应的函数就是规整函数 w(n)，如图 2-8 所示。动态时间规整其实是将一个 n 阶段的决策过程划分为 n 个单一阶段的决策过程。那么所选取的规整函数 w(n)需要满足以下条件：1.w(n)为单调函数。2.规整函数必须从(1,1)点开始至(N,J)点结束。3.规整函数不能跳过任何点。4.最大规整量不能超过限定值，用式子表示即为：$|n-j|<Q$，Q 称为“窗宽”一般取 2。传统的 DTW 算法是把时间规整和距离测度结合起来的一种非线性规整技术。但是传统动态规整算法(DP 算法)的计算量比较大，由运算量大而影响了系统识别速率。那么如何减小计算量而且更好的进行匹配以及获取到更准确的语音识别结果就必须进行算法的改进与优化。

3.2 DTW 算法的改进与优化

应用传统DTW算法来进行识别，识别效果不是很好。那么本文对传统DTW

算法进行了改进优化。语音信号进行预处理之后便需要进行语音信号的端点检测，端点检测是语音识别技术中的关键所在。语音信号的整个端点检测的工作流程如下图3-2所示。好的端点检测方法会给以后建模以及识别过程带来很多便利，同时能实现更快更精确的识别[24]。

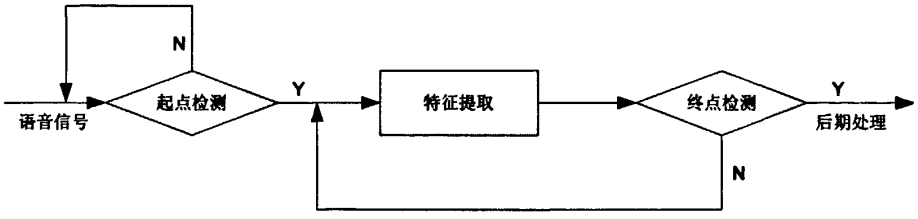


图3-2　端点检测

3.2.1 可变窗长的语音端点检测

用窗函数对语音序列进行处理后所获取的一帧语音序列的长度称为窗长也被称为帧长。它是指在窗化处理后进行的分帧处理。由于语音信号具有时域特性，它是按照时间先后顺序进行读取与存入的，那么取帧也就可以按时间顺序来取，在存储空间中也就可以按存入存储空间中的先后顺序来取。据大量实验统计，一般的语音信号的窗长取10ms至20ms之间，前一帧与后一帧的交叠部分为帧移，帧移一般是取小于10ms的。因为语音信号一般在10-20ms内是相对稳定的并且由信号的采样定理可知按上述方法来对语音信号进行取帧是合理的。而且如果对语音信号取比较小的窗长，那么就能够比较准确的检测到语音信号的端点，但是这样一来却增加了计算量，使得语音识别耗时较长，同时也会影响系统的速率。反之，如果所取的窗长很大，那么的确能减少计算量同时能提高语音识别的速度，但是端点检测的结果却是很不精确，对识别结果也会造成比较大的影响。为此，可灵活的针对不同的情况进行不同的处理，这样就可以在语音静音段时采用较长的窗进行处理，在语音段采用常规窗进行处理，在语音的过渡段采用较小的窗进行处理，这样处理既可以较为精确的判断语音的起止点也可以提高识别速率。一旦确定语音的起点，就改用常规窗长来处理语音段，同样的对于终端也采取相同的处理方式。其程序框图如下图3-3。

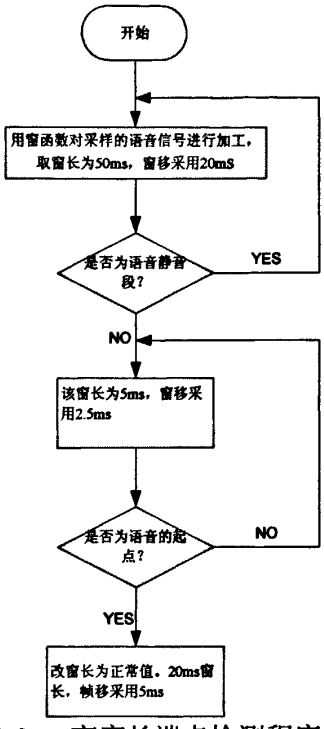


图3-3　变窗长端点检测程序框图

那么如何来判断语音是否进入到语音段，或者是静音段呢？我们用下一小节的内容来判断。

3.2.2 双门限端点检测

端点检测在语音识别过程中是比较关键的一步，好的起始点与终点能直接影响识别结果。那么先来看看，由传统端点检测方法得到的结果情况如下图3-4。图中红线即为所获取的端点，获取的端点值为：起始点42，终止点为130。

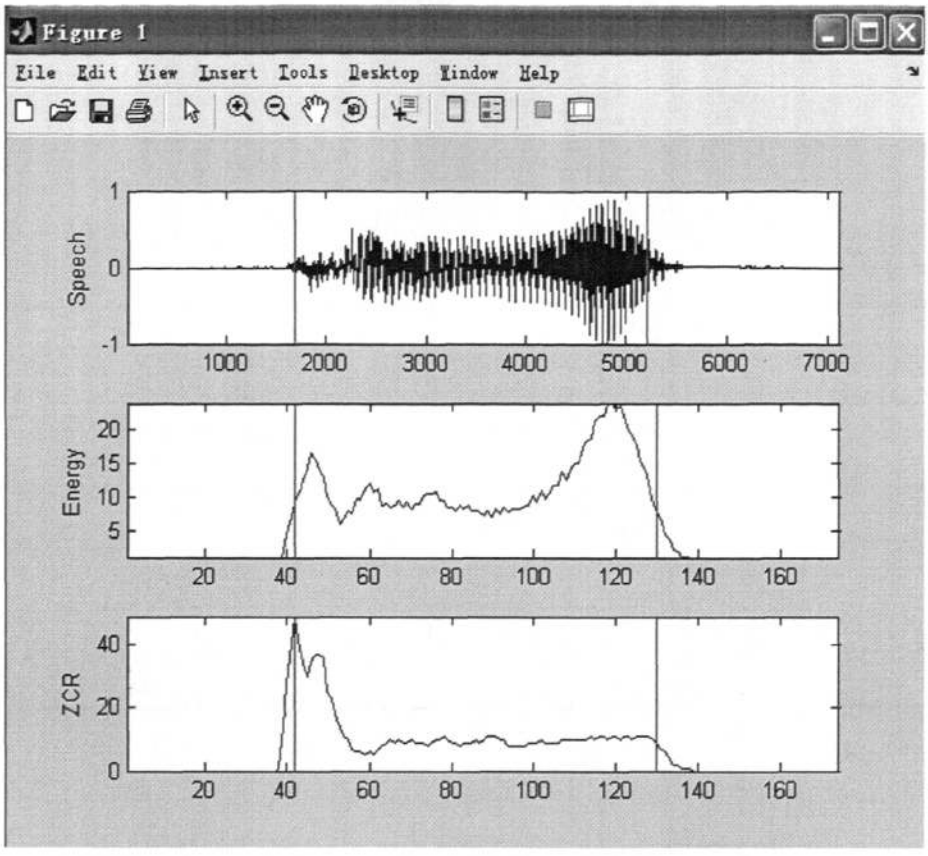


图3-4　传统端点检测的检测结果

不过上图是仅仅用到短时过零率来为端点检测做判断的。那么能否将传统的两种端点检测方法结合起来呢？基于能量过零率的双门限端点检测使用两级判断，先用短时平均能量做第一次判别，然后在此基础上用短时过零率做第二次判别。短时平均能量可由2-6式来表示， 短时过零率则可由2-7来表示。用双门限进行端点检测，高门限被用于确定语音的近似起始端点，低门限用于确定语音真正起始端点，但若只仅仅通过低门限检测到的语音信号端点未必就是语音的起始端点，也有可能是短时的噪音[25]。那么当高门限已经确定语音起始端点后，再利用低门限作为限制条件才能有效的确定语音的真正起始点。对于语音结束点，其端点检测方法与语音起始点的判别方法一样。有时噪声的能量是相当大，可能超过高门限，但是噪声一般持续时间比较短，可以用持续时间来确定信号是噪声还是语音。当然门限值是在对语音信号进行概率统计和在识别环境下获取到噪声信号的能量值之后所选定的值[26]。通过实验统计值来确定所要选取

的双门限值。具体分析图谱如下图3-5所示。

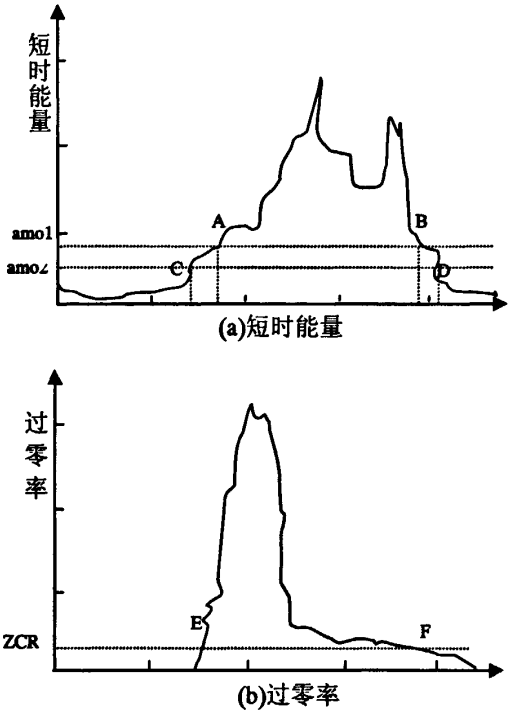


图 3-5　利用短时能量和过零率的语音端点检测

首先根据语音短时能量选取一个比较高的门限值amp1如图3-5(a)所示，语音能量在这个值之上的基本上都是语音信号，所以第一次选取的即为初判的语音起止点。它位于该门限与短时能量包络交点对应的时间间隔之外(如图AB 段之外)。然后根据噪声能量确定一个较低的门限amp2如图3-5(a)所示，这样从A点向前，B点向后进行搜索，找到低门限与短时能量包络第一次相交的位置C点和D点。像这样，那么CD段就是双门限初次判定的语音段。然后进行第二次判断，这次通过短时过零率确定的门限来判断，所谓的双门限法，即要在所采样的语音信号中找到必须能同时都满足图3-5所示的情况的采样点并将它选取为语音的起始点或者终点。那么用短时过零率确定的门限继续来判断的话就得从CD两端分别向前向后进行搜索，找到短时平均过零率第一次低于门限ZCR的两点E和F如图3-5(b)所示，这便是通过双门限法判断得到的语音的起止点[27]。所选取的门限值是经过统计以及实验来获取的，在分别计算语音信号在静音情况下语音信号的短时能量值以及短时过零率和计算语音所处环境的环境噪声的短时能量值

以及短时过零率，综合以上所得到的结果来确定所要选取的语音信号合适的双门限值。如果语音信号的短时过零率或者短时能量门限高于最低门限，那么认为进入过渡阶段，就采用较小的窗长进行处理。一旦进入语音段，就是短时能量超过较高门限，于是就从这帧以后恢复为常规窗长。本文所采用的试验相关数据如下表3-1所示。

表3-1　可变窗长试验相关数据

参数	常规窗	较大窗	较小窗
窗长	160	300	40
帧移	40	100	20
短时能量较高门限	1.6	9	0.7
短时能量较低门限	0.8	2	0.3
过零率低门限	5	8	2

在本文所述的语音识别系统中，语音的采样频率采用8kHZ，16位，20ms一帧进行采样也就是160点数据为一帧。按照语音发声特点帧移一般取10ms以内也就是80点数据以内为一帧，本系统采用可变窗长来进行处理，针对不同的语音信号部分采用不同的窗长来进行采样取帧。正常窗长取20ms一帧，5ms的帧移即40点数据的帧移。较长窗取300点为一帧，帧移取100点，而较短窗取40点为一帧，帧移取20点。经过双门限法处理后语音信号的频谱图、短时能量和短时过零率的图形仿真结果如下图3-6所示。所读取的语音信号为“9”，采用的位速为128kbps,音频的采样大小为16位，频道是单声道，音频的采样频率为8kHz，音频格式为PCM格式。

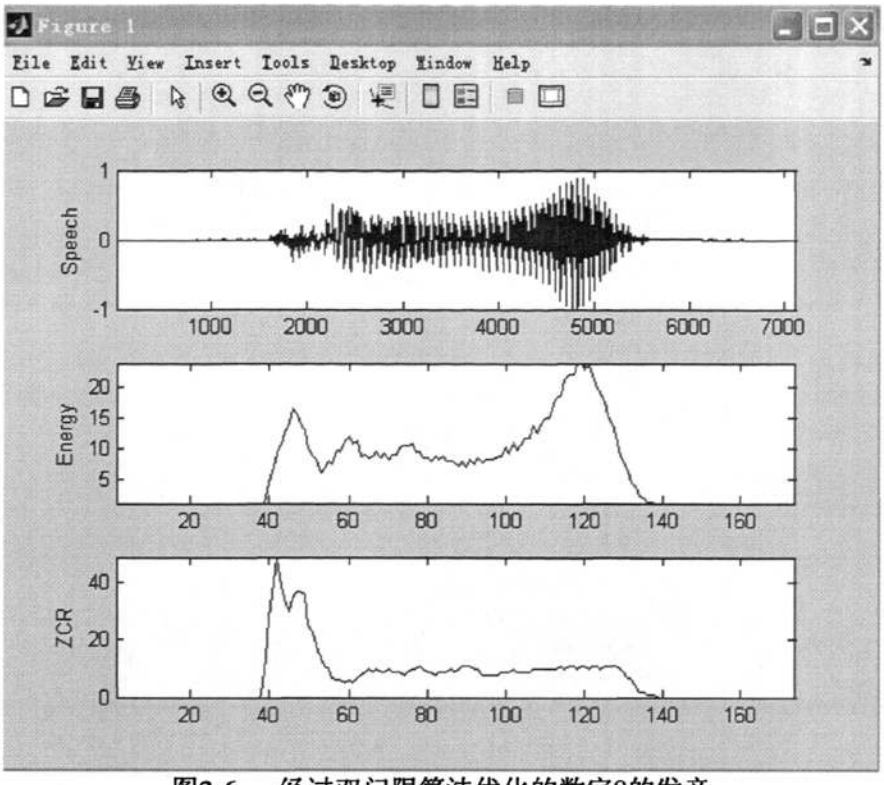


图3-6　经过双门限算法优化的数字9的发音

双门限法中的较高门限是通过在安静的环境下所获取到语音信号的能量谱然后取其能量平均值，确保语音信号的幅值都在这一值之上的信号都是所要获取的语音信号，那么较低门限则是根据所选择的环境噪声的平均能量值来选取的，所选择的低门限值要高于噪声的平均能量值，短时过零率则是通过背景噪声中的信号来确定的。根据噪声可以确定一个门限，避免虚假过零的出现，那么可以进行对过零率加一修正值，由此得到如下3-4所示的公式。

$$Z_n = \sum_{m=-\infty}^{\infty} \left\{ \left| \mathrm{sgn}\left[x(n)-T\right] - \mathrm{sgn}\left[x(n-1)-T\right] \right| + \left| \mathrm{sgn}\left[x(n)+T\right] - \mathrm{sgn}\left[x(n-1)+T\right] \right| \right\} w(n) \qquad (3\text{-}4)$$

公式中的T为修正值，可以将所选取的过零率门限设置为此值[28]。那么经过双门限端点检测和可变长窗长的端点检测相结合的方法，获取到的语音信息的频谱、短时能量以及短时过零率的图形如下图3-7所示。

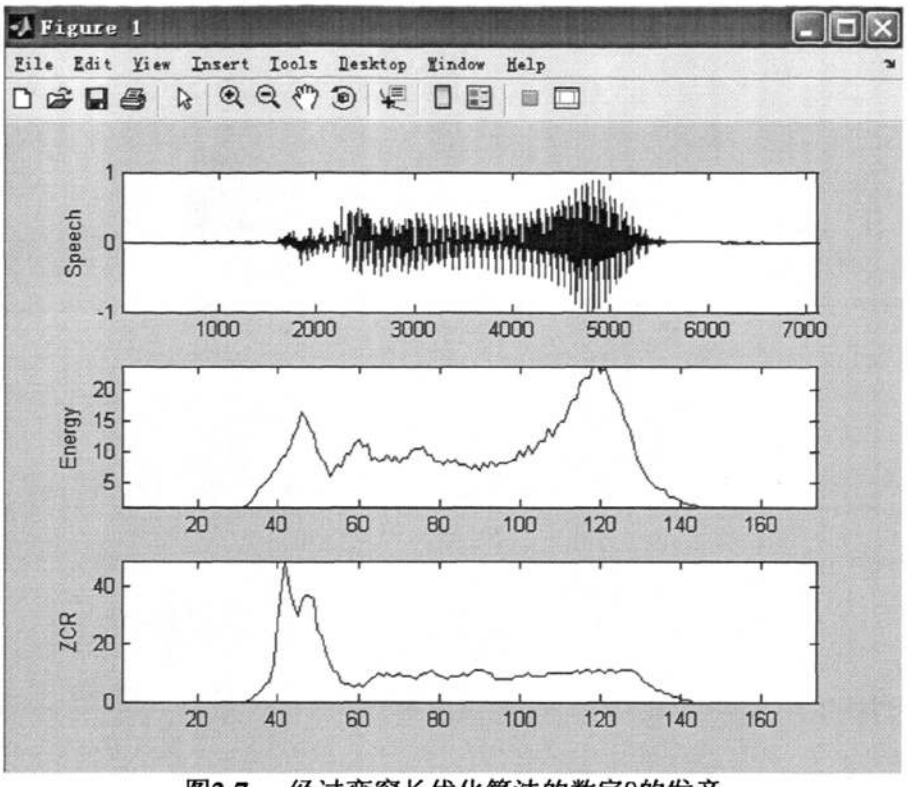


图3-7　经过变窗长优化算法的数字9的发音

可以对照3-5所示的示意图，在只用双门限进行端点检测时，所获取的静音到语音的过渡点为30（即C点或E点），通过短时能量较高门限确定的点为42（即A点），通过短时能量较高值确定的出语音点数为130（即B点），所获取的语音终止点与静音的过渡点为140（即D点或F点）。由这些值我们也能推测到语音的起始点在30至42点这两点数之间，语音的终止点在130至140这两个点数之间。我们通过MATLAB仿真所获取到的语音的起始点为38，终止点为136.5000，由于小数是在计算中所产生的，所以应该所取的点数应该为136。由上述的一些数字我们可以看出，如果整个语音信号是比较平稳的情况，而且噪声信号也是对称的，那么通过以上的数字可以看出，过度点C与D的对应。然后经过可变窗长，对不同窗长所采集的帧分别进行所选取的不同门限进行再一次的检测，所选择的门限值情况见表1所示，将不同语音段选取不同窗长和不同门限值来进行端点检测后所得到的语音起始点为38，语音终止点为136，由此可知两次语音处理所获取到的起点与终点是相对稳定的。用图形标示出来则为3-8所示情况。

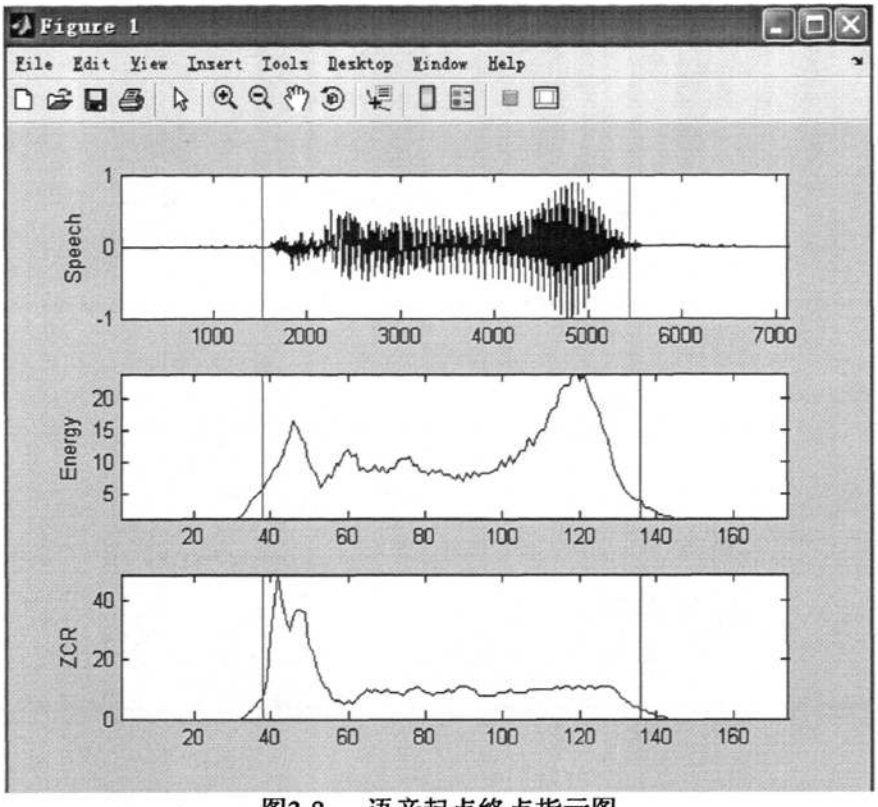


图3-8　语音起点终点指示图

3.2.3 加权端点检测

以上是通过短时能量以及短时平均能量来判断端点，区分发音区与静音区。短时平均能量是基于语音帧来进行的，短时过零率是指一帧信号中波形穿过零点的次数，短时平均能量与短时过零率的公式如下式 3-5 和 3-6 所示。

$$E(i)=\sum_{n=1}^{N}x_n^2(n) \tag{3-5}$$

$$Z(i)=\sum_{n=1}^{N-1}\left|x_i(n)-x_i(n+1)\right| \tag{3-6}$$

但是由于清音信号有较高的过零率，而浊音信号有较高的短时平均能量，所以在实际中我们便可以分别进行处理，用过零率来检测清音，用短时能量来检测浊音。现在本文将它们结合起来，分别选取一个加权系数，例如对能量较大的浊音信号，我们就对短时能量采用加权系数a，而对短时过零率采用较小的加权系数b。那么对于短时过零率较高的清音段来说，短时能量就选用较小的加权系

数，对较大的短时过零率就选用较大的加权系数，最后可用公式表示为如下3-7所示的情况。

$$D = aE(i) + bZ(i) \tag{3-7}$$

经过实验统计，当a=0.86，b=0.23时，端点检测的效果是最佳的。那么通过以上加权端点检测方法所获取到的端点检测结果如下图3-9所示。

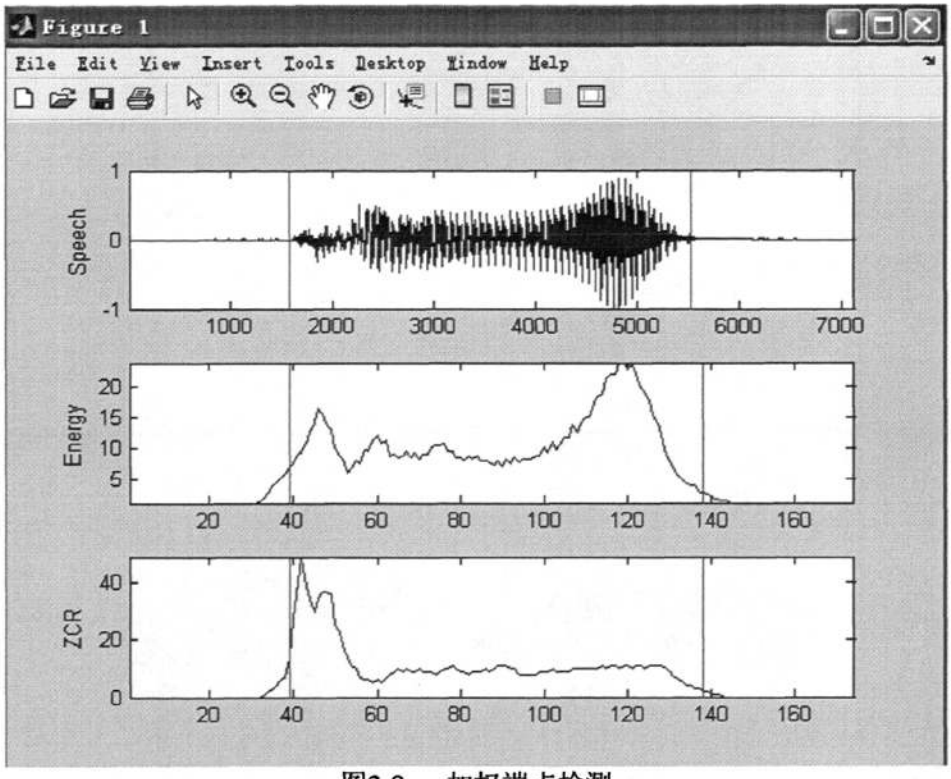


图3-9　加权端点检测

从中获取到的端点值为39与138，由于对清音与浊音进行了分别处理，所以端点更朝前。但是这种方法也存在诸多的问题，首先语音的清音与浊音不好处分，此文只是通过大概的在知晓发音的情况下所做的处理，如果在不知道语音信号特点的情况下就难以知道到底所发出的语音信号是在清音段还是在浊音段，如果需要区分则需要更加精确的方法以及更加完善的算法来实现。

3.2.4 整体路径约束的 DTW 算法识别

一般情况下，规整函数如图 2-5 所示，它会被限制在一个区域内，一般是选择一个平行四边形区域中，如图 3-10 所示。本文对 DTW 算法进行改进，采用

整体路径约束的 DTW 算法来进行规整函数的选取。经过多次实验与仿真对照，本文将所设计的此平行四边形它的一条边的斜率设置为 2，另一条边的斜率设置为 1/2。传统 DTW 算法选择的规整函数的起点是(1 ,1)，终点为(N ,J)。DTW 动态规整的目的是在此平行四边形内由起点到终点寻找一个规整函数，使其找到一个能让计算总失真最精确的函数，这样就保证了参考模板与待测试模板之间具有最大的声学相似特性。要找到一个最佳的规整函数就涉及到最佳路径选择的问题。

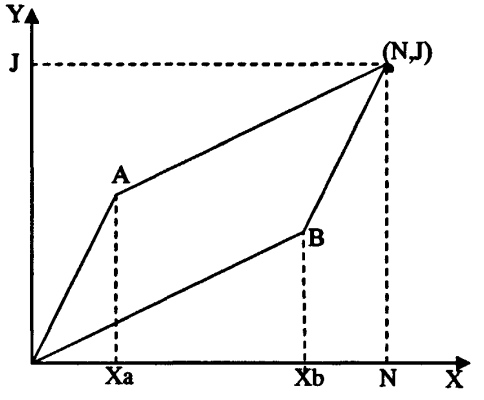


图 3-10　匹配路径约束示意图

在模板匹配过程中设定弯曲的斜率，这样就可以将那些在所设置区域外的帧匹配是没有必要计算的，同样也没有必要去保存所有的匹配距离矩阵和累积距离矩阵。

如图3-10，把实际的动态弯曲分为1—Xa，Xa—Xb，Xb—N，而且Xa与Xb需要满足通过以下两式计算所得到的最接近的整数。

$$X_a = \frac{1}{3}(2J - N) \tag{3-8}$$

$$X_b = \frac{2}{3}(2N - J) \tag{3-9}$$

根据终点（N,J）与平行四边形的边所在直线的斜率为2与1/2得到J与N的限制条件。这样当J与N不满足下面两个式子时，就认为这两帧相差太大，于是就没有必要进行动态弯曲匹配了。

$$2J - N \geq 3 \tag{3-10}$$

$$2N - J \geq 2 \tag{3-11}$$

若将匹配路径放入每格代表一个单位的网格中，可知因为每一列各格点上的匹配计算只用到了前一列的三个网格，那么对于 X 轴上每前进一帧，所要比较的 Y 轴上的帧数虽然不同，但弯折特性是一样的，累积距离是由下式 3-12 计算得到。

$$D(x,y)=d(x,y)+\min[D(x-1,y),D(x-1,y-1),D(x-1,y-2)] \qquad (3\text{-}12)$$

上式 3-12 是对 3-3 式所作的改进。其中只需要两个列矢量 D 和 d 分别保存前一列的累积距离和计算当前列的累积距离，而不用保存整个距离矩阵。这样即可以节省占用的存储空间，同样也可以提高系统处理速度[29]。

语音识别时，将待测语音与模板库中的每一个模板进行模式匹配，找到距离最小的作为输出结果。

那么现在就可以比较下通过传统 DTW 算法与经过整体路径约束后的 DTW 算法来计算匹配距离得到的失真测度情况。所选择的模板语音与测试语音为“0-9”这十个数字语音，进行分别匹配计算，得到的匹配距离也称为失真测度如下图 3-11 与下图 3-12 所示。表中的 0-9 行表示“0-9b.wav”测试语音，0-9 列表示“0-9a.wav”参考模板库语音。（注释：见下图中 0 行 0 列所代表的是待测试语音‘0’与模板语音‘0’匹配所得到的匹配距离数据 2.8118，以下各数据均相同。）

Command Window

```
dist =

  1.0e+004 *

    2.8118    3.7318    6.2172    4.0890    4.1207    4.4032    4.7963    3.7736    5.9914    5.9687
    3.0506    2.2623    8.8504    5.8146    5.3923    5.4900    6.0817    3.5474    8.3812    8.0245
    5.5317    8.5042    2.1494    2.8547    3.1299    5.3047    3.9253    6.8547    2.4506    4.8166
    3.7377    5.7870    2.4001    1.9182    2.3422    3.3254    2.7811    4.3346    2.2301    3.5479
    4.2496    6.2897    3.5666    2.8811    1.7486    4.1912    3.5952    4.9555    3.3259    4.7787
    4.6893    6.0710    5.5983    4.4619    4.2721    1.6985    4.1970    6.5658    5.0330    5.0392
    4.5260    6.3299    4.2359    3.3870    3.6871    3.9248    3.7496    5.2845    4.0339    4.9018
    3.5331    3.3769    7.3843    4.8084    4.5937    5.5099    5.4988    3.1030    7.0786    7.0247
    5.3994    7.9595    2.3944    3.1481    3.3421    5.2509    4.3012    7.3338    1.6864    4.8529
    5.3444    7.6427    4.5296    4.1464    4.1028    4.2337    4.5559    7.0067    5.3354    4.0566
```

图 3-11　传统 DTW 算法的测试语音与模板库距离

Command Window

```
dist =

  1.0e+004 *

    0.9593    2.8795    6.7212    4.4313    5.3612    4.6411    3.4612    2.9488    5.6389    3.1425
    2.4517    0.8745    9.0795    6.1980    7.3712    6.2162    5.5653    2.2115    8.9006    5.8044
    5.9705    9.2490    0.7156    2.1416    3.9558    6.1704    4.2062    6.0837    1.6155    4.9117
    3.7764    6.3916    1.6844    0.9937    3.0665    3.9832    3.0843    4.3973    1.6920    3.5027
    3.9266    6.5936    3.7107    2.7521    0.7256    4.8260    4.2220    4.8914    3.5184    3.4013
    3.9082    5.1288    5.1676    4.9359    5.0197    0.8247    3.8769    5.8527    4.7416    3.5427
    3.3360    6.3503    5.0165    3.9475    4.6770    3.8946    1.0512    5.1318    3.7982    2.3548
    3.3782    3.5605    7.3841    4.3939    5.9449    6.6161    5.9700    1.2495    7.9512    4.9623
    5.2478    8.9183    1.6819    2.6438    3.7264    5.5704    4.0630    6.9419    0.7271    4.2954
    3.7011    7.2617    4.8007    4.1399    4.4978    3.6768    2.3612    6.5204    3.6689    0.8454
```

图 3-12　规整约束路径 DTW 算法的测试语音与模板库距离

而整体路径约束后的匹配距离具体用图形来描述则为图 3-13 所示。由图可知沿 X—Y 坐标平面的对角线所得到的匹配距离最小，这也正说明所识别的结果正确。因为测试语音与参考模板语音正确匹配在坐标平面上能通过在对角线处所得到的值是否是最小来体现。

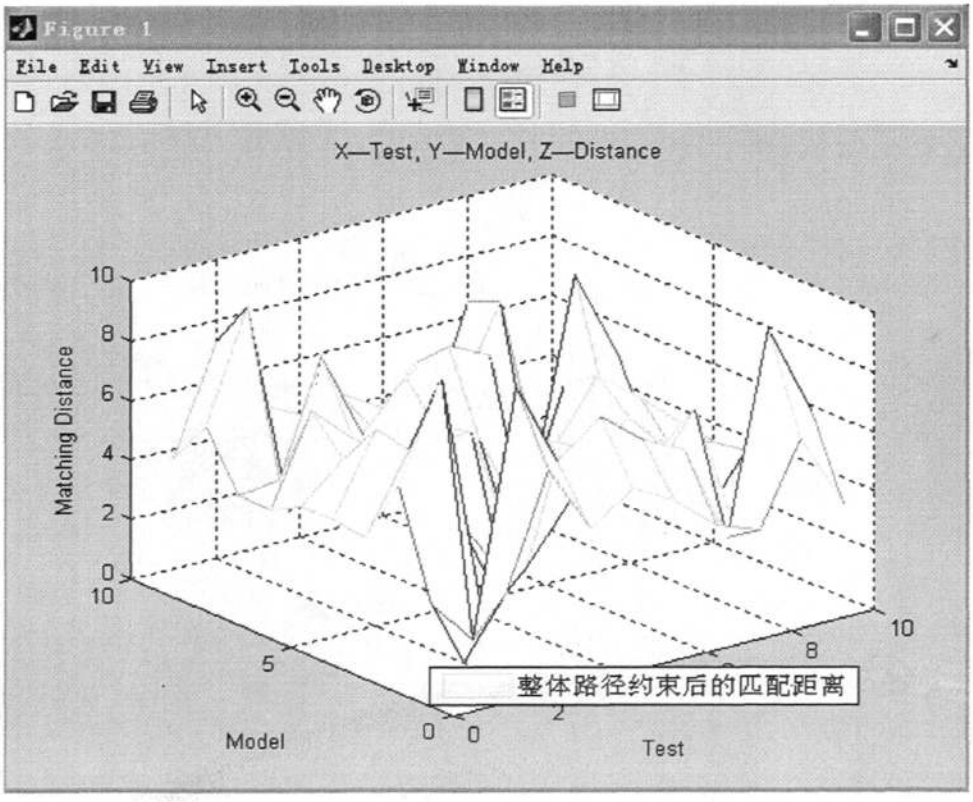


图 3-13　整体路径约束后的匹配距离图

3.2.5 松弛起终点的 DTW 算法识别

那么能不能在上几节的基础上将算法再进行优化呢？答案是肯定的。DTW 算法，一般都是将起点终点固定，即起点选择在（1，1）而终点选择在（N,J），这样起点和终点就是固定的。若不固定起点和终点结果又会如何呢？由于模板的帧数不可能完全对应，而且人发声一般要晚于计算机开始计算搜索路径数据的时间，所以可以放松一两个点使起始点从如图 3-14 所示搜索范围开始选取最佳路径的起始点。这样做的好处是使搜索路径的起点选择具有一般性，虽然在程序和算法方面与固定起终点相似，同样也是利用欧氏距离公式来计算累积误差。但这种松弛起点与终点的方法能更好的获取搜索路径，不会因为固定端点而造成误差。

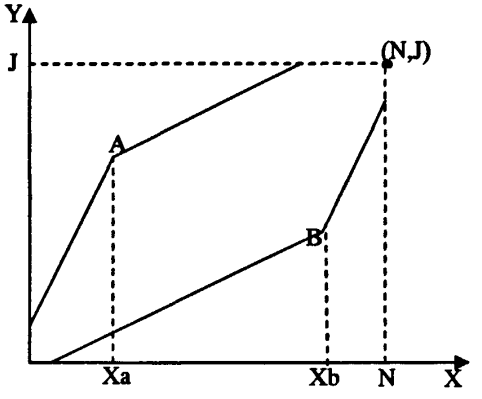


图 3-14　松弛起终点路径搜索范围示意图

那么利用改进后的 DTW 算法即松弛端点法进行语音匹配，结果又会如何呢？让我们用数据来说明问题，改进后测试语音与模板库之间的距离测度如下图 3-15 所示。

```
Command Window

dist =

  1.0e+004 *

    0.8965    2.7970    6.4551    4.1702    5.2191    4.5883    3.3832    3.0032    5.4537    2.9447
    2.3858    0.7601    8.7209    5.9461    6.8724    6.0243    5.5242    2.0773    8.4896    5.7072
    5.8477    9.0622    0.6736    2.0948    3.7871    6.0691    4.0912    5.9467    1.4698    4.8252
    3.4677    6.3916    1.5546    0.8051    2.9535    3.9321    2.9452    4.3973    1.6657    3.2869
    3.7289    6.2977    3.6345    2.7023    0.6652    4.8260    3.9288    4.7790    3.4684    3.3911
    3.8908    5.0624    4.9945    4.9341    4.8497    0.7107    3.8442    5.7790    4.7301    3.5427
    3.1964    6.0406    4.9867    3.5867    4.5403    3.8737    0.9067    4.9661    3.6975    2.3260
    3.2574    3.2982    7.2922    4.2129    5.7407    6.4193    5.8161    1.1448    7.8567    4.8883
    5.1354    8.7628    1.6397    2.7154    3.6065    5.5070    4.0201    6.8620    0.6643    4.2597
    3.4276    7.0956    4.6201    4.1142    4.3361    3.4768    2.0224    6.5315    3.5756    0.6252

开始计算匹配结果...
测试语音模板 0b.wav 与参考语音模板 0a.wav 匹配
```

图 3-15　松弛起终点 DTW 算法的测试语音与模板库距离

比较图 3-12 与图 3-15 中的数据，可看出，再经过松弛起终点的 DTW 算法处理后的结果要优于路径约束的 DTW 算法处理结果，其匹配距离也要小，而且精确度也有所提高。获取到了这些匹配距离，则只需要在软件上实现一个循环查询就可以进行识别了。针对每个测试模板，比较其与参考模板库中每个模板的距离大小，待测试模板被认为是与它测度距离最小的参考模板匹配。而经过松弛起终点的 DTW 算法优化后的匹配距离用图形描述为下图 3-16 所示的情况。

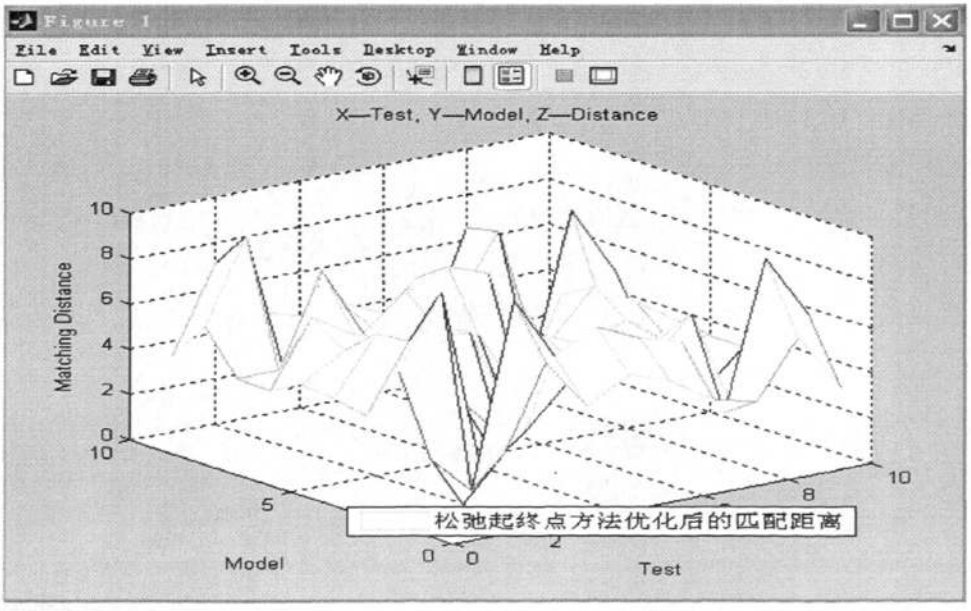


图 3-16　松弛起终点方法优化后的匹配距离

3.2.6 模糊度距离测度的 DTW 算法识别

模糊模式识别是利用模糊集的思想和方法，来解决模式识别问题。它可应用于语音信号的特征抽取与选择。用模糊算法来进行语音匹配必须首先知道模糊识别的模糊度。例如一个有 n 个支持点的模糊集 F 的模糊度$\gamma(F)$反映了它的模糊程度，可以用它与和它最接近的确定集合$\tilde{F}$之间的距离来度量[30]。用式子表示为如 3-13 所示。

$$\gamma(F)=\frac{2}{n}d\left(F,\tilde{F}\right) \tag{3-13}$$

式中$d\left(F,\tilde{F}\right)$表示 F 与$\tilde{F}$之间的距离，$\tilde{F}$只包含 F 中隶属度大于 0.5 的支持点。为了使$\gamma(F)$的值在 0 到 1 之间而将式子中设定一个 K 值，K 的取值根据采用的距离度量不同而不同。例如如果是采用欧式距离进行计算的话，那么 K 的值应该设置为 2，而此时$\gamma(F)$就称为二次模糊度，也可以将 K 称为模糊度的次数，当 K 等于 2 时，可以将$\gamma(F)$更详细的表示为如下式 3-14 所示。

$$\gamma(F)=\frac{2}{\sqrt{n}}\left[\sum_{i=1}^{n}(\mu_F(x_i)-\mu_{\tilde{F}}(x_i))\right] \tag{3-14}$$

式中隶属度函数$\mu_F(x_i)$表示x_i隶属于集合 F 的程度。那么这个隶属度函数是如何通过计算得来的呢？下面就详细介绍隶属度函数的具体表达式。如下式 3-15 所示。

$$\mu_{FS}(x_i;a,b,c)=\begin{cases}0, x_i \le a\\ 2[(x_i-a)/(c-a)]^2, a \le x_i \le b\\ 1-2[(x_i-a)/(c-a)]^2, b \le x_i \le c\\ 1, x_i \ge c\end{cases} \tag{3-15}$$

式中三个参数 a，b，c 它们是用来确定隶属度函数的具体构成的。参数 b 等于参数 a 与参数 c 的平均值。参数 a，b，c 由下式 3-16 来表示。

$$\begin{aligned}b&=(x_{qj})_{av}\\ c&=b+\max\left\{\left|(x_{qj})_{av}-(x_{qj})_{\max}\right|,\left|(x_{qj})_{av}-(x_{qj})_{\min}\right|\right\}\\ a&=2b-c\end{aligned} \tag{3-16}$$

其中x_{qj}表示第 J 类中的第 q 维特征，用$(x_{qj})_{av}$，$(x_{qj})_{\max}$，$(x_{qj})_{\min}$来分别表示该特征的均值，最大值和最小值。由这些均值、最大值、最小值来得到每个测试语音模板与每个参考模板之间的特征差距。而本文是通过如下的方式来计算得到

待测试语音与参考模板语音之间的距离测度。首先将待测试语音的 MFCC 系数与参考模板的 MFCC 系数进行 3-3 式的线性扩展变换，将待测试语音的 MFCC 系数与参考模板的 MFCC 系数具有相同的行与列。然后再按照 3-17 式将两对照模板分别取自己的隶属度参数 a，b，c。通过 3-16 式分别计算出它们各自的隶属度函数。最后将所获取到的隶属函数进行 3-15 的二次模糊度计算。将结果进行比较，找出待测试语音与参考模板库中所有的参考模板所计算出来的二次模糊度最小的那个参考模板，于是待测试语音便被认为是与此参考模板匹配。本文在对 MFCC 系数进行 3-17 的计算时是按照每列进行的。因为在前期进行三角滤波时，所选取的滤波器数为 24，所获取到的 MFCC 都为 24 列数组。将每一列进行对照计算，然后将所有列计算的二次模糊度求和。在处理过程中，本文对 3-15 式做了相应的变动。如下式 3-17 所示。

$$\gamma(F)=\frac{2}{\sqrt{m}}\frac{2}{\sqrt{n}}\left[\sum_{i=1}^{m}\sum_{j=1}^{n}\left|(\mu_F(x_i)-\mu_{\tilde{F}}(x_i))\right|\right] \tag{3-17}$$

式中 m 即为本文所获取的 MFCC 系数的列数 24，而 n 则为经过线性扩展后的行数。经过上述处理后得到的匹配距离数据如下图 3-17 所示。

```
Command Window
开始计算参考模板的参数...
开始计算待识别语音的参数...
开始进行模板匹配...

dist =

  1.0e+003 *

   3.9100   4.8022   6.7621   4.0404   3.9764   4.0148   3.9525   4.3853   4.1166   5.3076
   2.2804   2.1482   2.5984   2.4239   2.2143   2.1949   2.1812   4.0514   2.1886   5.3706
   4.7453   3.7916   2.4021   4.2787   3.7951   5.7155   3.9158   3.9330   3.8980   5.3618
   4.2021   2.3803   3.9509   2.0507   3.6676   2.6885   5.3268   3.8709   3.5253   5.3061
   4.1686   2.3538   3.9205   4.0766   2.1926   2.6573   6.4328   3.8730   2.5032   5.2511
   3.2182   4.4056   5.8744   3.0451   3.0541   2.9301   3.0840   3.9880   3.0843   5.3656
   4.1422   2.4376   3.9388   6.4754   4.5755   2.8679   2.2373   3.8942   4.9188   5.3344
   5.9212   4.2803   6.1457   3.4188   3.8435   3.5659   3.5377   3.3521   3.5495   5.3802
   4.3970   2.4249   3.9530   2.7294   2.2263   4.0478   2.3846   3.9193   1.8856   5.3090
   4.9387   4.8649   5.3167   4.8336   4.7934   4.8211   4.8431   5.3402   4.8671   4.1563
```

图 3-17　模糊度距离测度方法所获取的匹配距离

由图知所获取的结果匹配距离更加精确。

3.3 语音识别仿真结果

由以上改进后的算法所得到的最佳匹配距离作总结，可以发现算法在改进的过程中是一步一步的在优化。本文是采用将下一个改进是在上一个改进的基础上进行的。通过仿真结果的比较可以直观的看出算法优化处理后的结果情况。将采用几种优化方法处理后的结果作比较，具体情况如下图 3-18 所示。

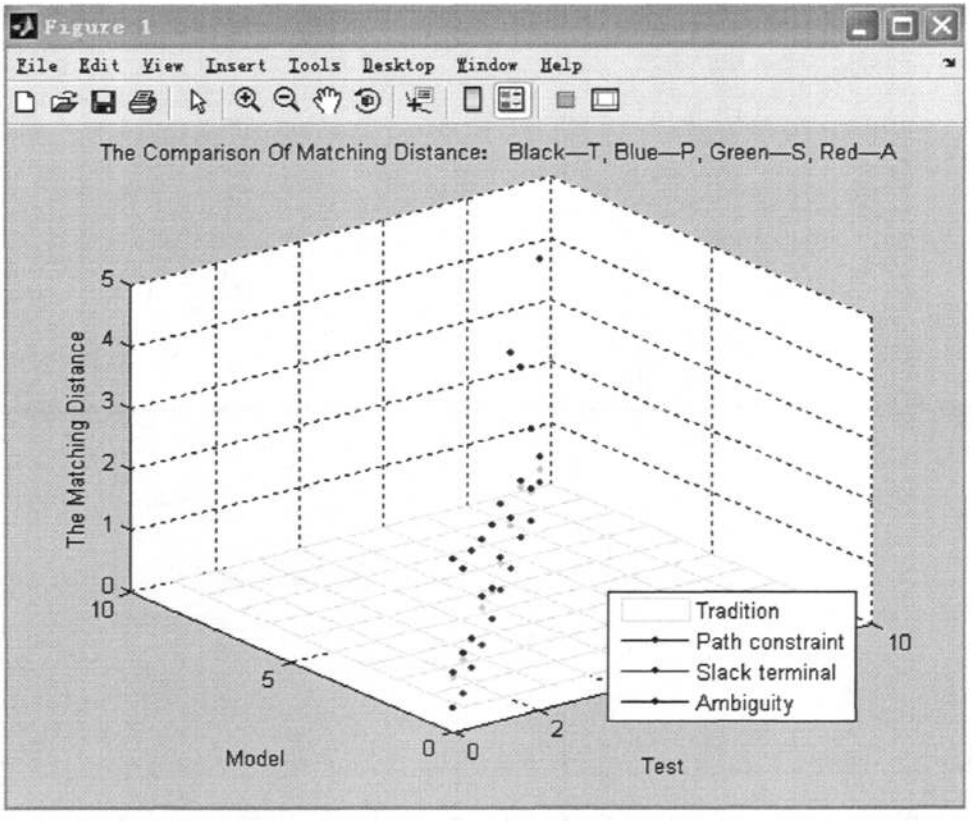


图 3-18　几种优化方法处理后的最佳匹配距离仿真结果比较

其中 X 坐标代表“0-9b.wav”10 个待测试语音，Y 坐标代表“0-9a.wav”10 个参考模板语音，每个点所代表的值是该点的 X 坐标所代表的测试语音与该点的 Y 坐标所代表的测试语音所得到的匹配值。例如：X 坐标 1 与 Y 坐标 1 的交点处所对应的 Z 值就是测试语音“1b.wav”与参考模板语音“1a.wav”所匹配的距离。其中黑点代表的是经过传统 DTW 算法得到的匹配距离，绿点代表的是经过整体路径约束后的匹配距离，蓝点代表的是经过松弛起终点后的匹配距离，红点代表的是经过模糊测度后的匹配距离。由图可以看出在同一匹配语音处红色代表的

匹配距离最小最精确。我们将黑点代表的传统方法所得到的匹配距离值与红点代表的经过几种算法优化后得到的匹配距离值进行比较，发现红点代表的值明显要小于黑点所代表的值。这也正体现了经过算法优化处理后的匹配距离更加的精确了。

具体仿真所获取的情况见下图 3-19 所示。其中作为参考模板的语音为 0a-9a.wav 这十个语音，而作为待测试的语音模板为 0b-9b.wav 这十个语音。由仿真结果知语音识别结果正确。

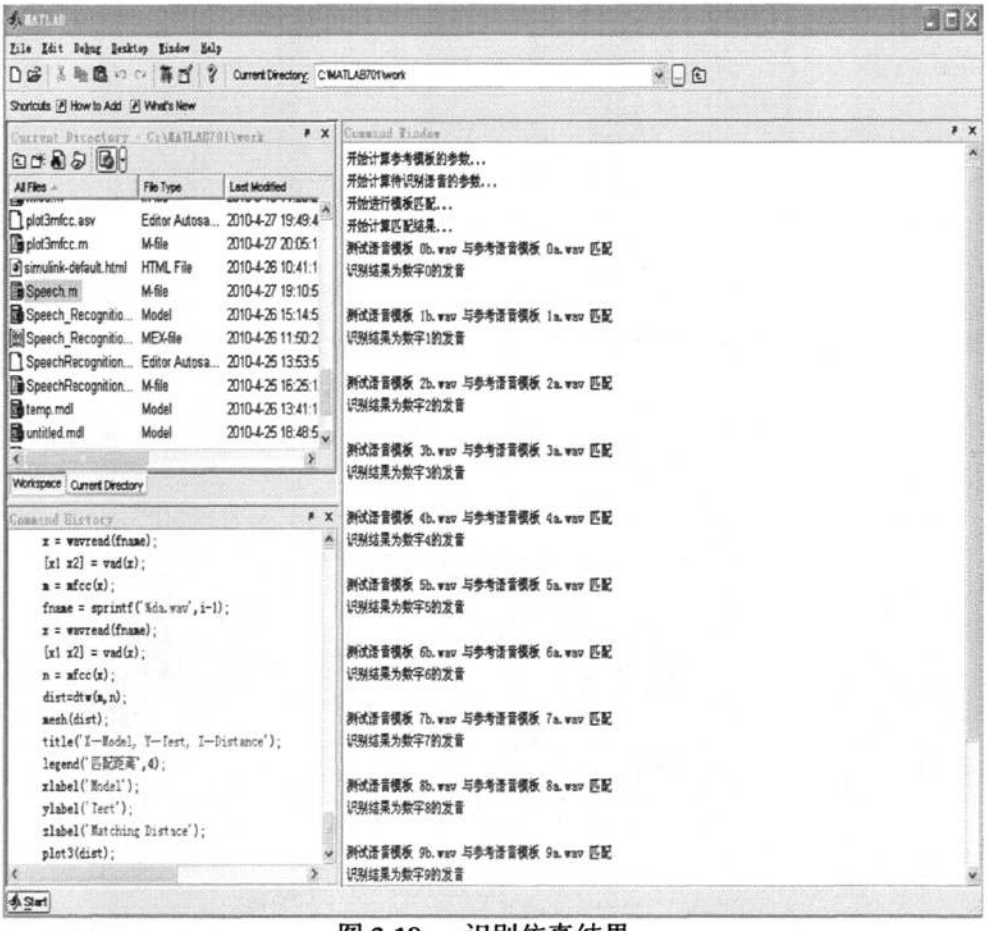


图 3-19　识别仿真结果

那么具体实现语音处理的仿真模型又是怎么样的呢？这个模块有三大部分，首先是语音采集和去噪部分，然后是短时能量与过零率分析部分，最后便是参数提取部分。具体如下图 3-20 所示：扬声器端是经过去噪处理之后的声音信号，从 EnergyScope 示波器可以得到短时平均能量的频谱，从 ZcrScope 示波

器可以得到过零率的情况，从 LPCScope 可以获得 LPC 参数的幅值情况，而具体的过零次数可以通过 Display 处获得。同时从 LPCSpectrum 处可以得到 LPC 参数的频谱图。

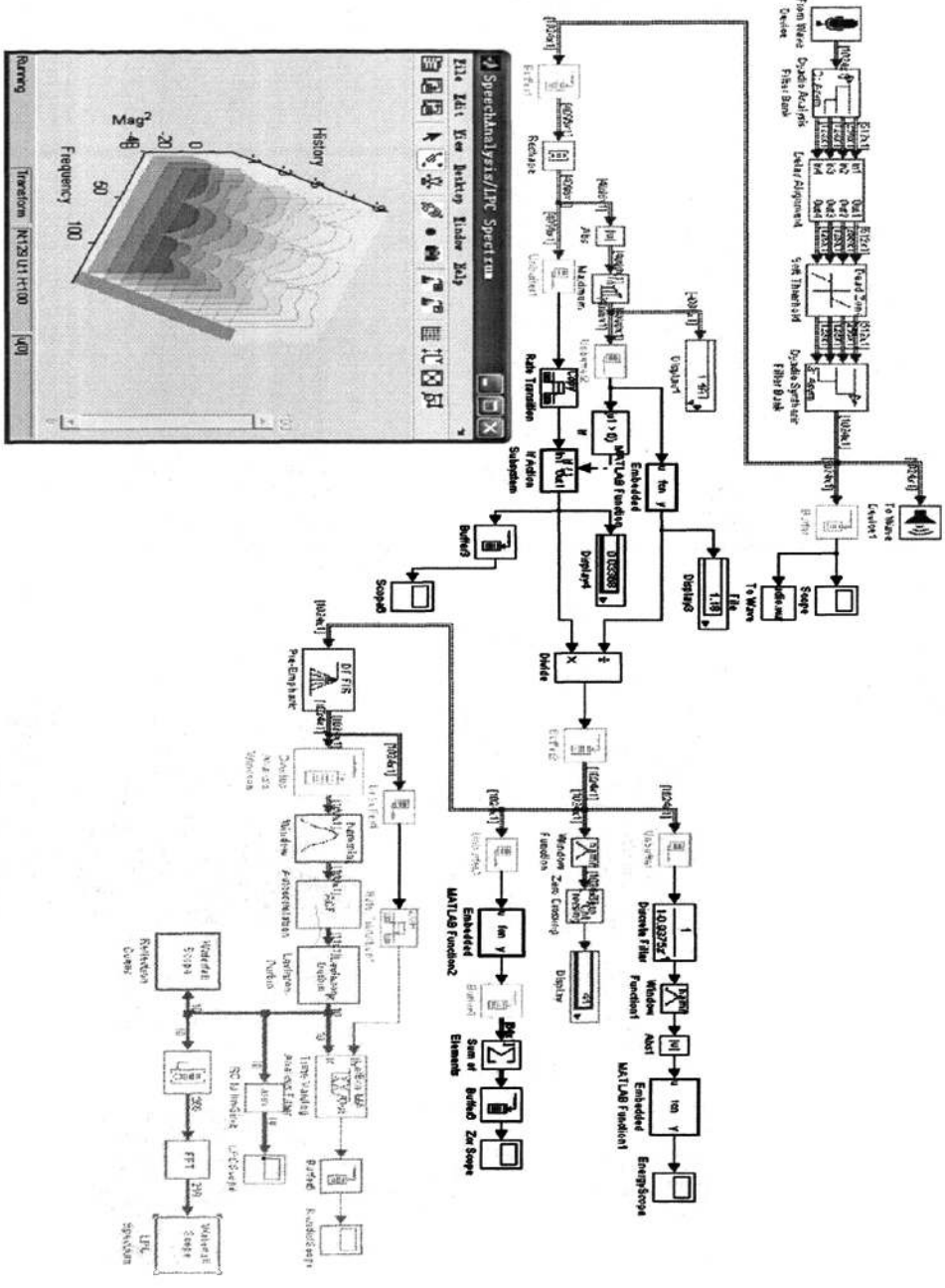


图 3-20　Matlab 的语音处理的仿真模型

3.4 DTW 算法在车载语音识别系统上的实现

DTW 算法在车载语音识别系统上的实现需要经过两个步骤。因为用 DTW 算法进行语音识别首先要获取到参考语音模板，那么 DTW 算法在车载语音识别系统上实现的第一个步骤是对系统进行语音训练。将训练成功的语音作为参考模板语音存入到系统的内存中。第二个步骤就是将待测试语音与参考模板语音进行语音识别。在系统读取到待识别语音时，将待识别语音分别与参考模板库中的每个模板语音进行匹配。选取匹配结果最优的参考模板语音作为识别结果。DTW 模型与算法应用在车载语音识别系统上能很简单的就实现语音识别的功能。小汽车一般工作在噪声比较大的环境中的，这就要求车载语音识别系统处于噪声比较恶劣的环境下也能正常的工作。本文通过前几节对 DTW 算法的改进，改进后的 DTW 算法是通过统计获取车载环境中的噪声短时能量以及短时过零率，然后通过这些频谱值来确定门限，同时在噪声与语音信号交织的情况中，采用加权的办法来进行语音端点检测的。这样即使在噪声比较嘈杂的情况下也能比较准确地识别语音信号的端点。而且由于 DTW 模型及其算法能直接通过硬件来实现，那么在实际应用中能通过硬件与软件结合的方法来实现，这样能让系统更加的稳定而且运行速率比较快。那么在解决车载环境比较差而要求汽车电子部件能稳定且能持续运作的情况下，可以对 DTW 算法在算法进行改进优化，使得软件系统运行更加稳定性能更加可靠。在比较嘈杂的环境下，可以采用硬件屏蔽和软件算法处理相结合的方法来对噪声进行滤除。总之，DTW 模型与算法在车载语音识别系统上的应用会获取到比较理想的结果。

3.5 本章小结

本章主要是对语音识别算法进行改进与优化并对改进的算法进行实现。首先介绍了传统 DTW 算法在语音识别过程中的应用，然后针对 DTW 算法进行了改进，其中重点是对端点检测的方法作了改进，将可变窗长的语音端点检测与双门限端点检测方法结合起来，同时在 DTW 语音识别的过程中采用了松弛起点与终点的办法，将改进后的仿真结果与传统语音算法 DTW 的仿真结果作了比较，对语音识别做到了优化处理。最后介绍了改进后的 DTW 算法在车载语音识别系统上的实现。

第 4 章　语音识别系统的硬软件设计

4.1 语音识别系统的硬件设计

目前，基于车载语音识别的嵌入式系统设计方法多种多样，而很多都是基于 ARM 和 DSP 的。虽然 ARM 和 DSP 这样的处理器功能很强大而且处理速度也很快，但是价格也很贵。若仅仅是应用于语音识别系统中的话，ARM 和 DSP 的选用就显得有些大材小用了。本文选用 SPCE061A 来作为语音识别系统的核心处理器，在此基础上来完成语音识别系统的设计。本文选用凌阳 61 单片机的原因是在同样能实现语音识别功能的前提下，SPCE061A 的价格最低。

4.1.1 语音识别系统的架构说明

本文选用凌阳公司生产的16位单片机SPCE061A为核心处理器，虽然它不像ARM处理功能那么强大，也不能达到DSP的处理速度。SPCE061A它的指令系统能提供具有较高运算速度的16位×16位的乘法运算指令和内积运算指令，在应用中增添了类似DSP的功能，运用在复杂的数字信号处理方面既便利，又比专用的DSP芯片廉价得多[32]。在语音识别系统中，SPCE061A它有内置的麦克风方便语音数据的输入与采集，而且语音口具有自动增益控制的输入功能，在MIC 输入端带有自动增益（AGC）控制功能。

SPCE061A它具有较高的处理速度能够快速的处理复杂的数字信号。本文采用SPCE061A作为一个16位控制器，在61板通过语音口采集到语音信号后对语音信号进行处理，然后由61板来进行识别[33]。本系统要实现对小车的控制，还需要RFID模块进行射频传控，让射频接收端接收到发射信号后用51的最小系统来控制小车响应相应的动作。RFID模块有很多种，本文选择了315MHZ的DF收发模块来实现射频通信，采用51单片机来完成编码与解码的功能。下图4-1DF发射模块原理图。

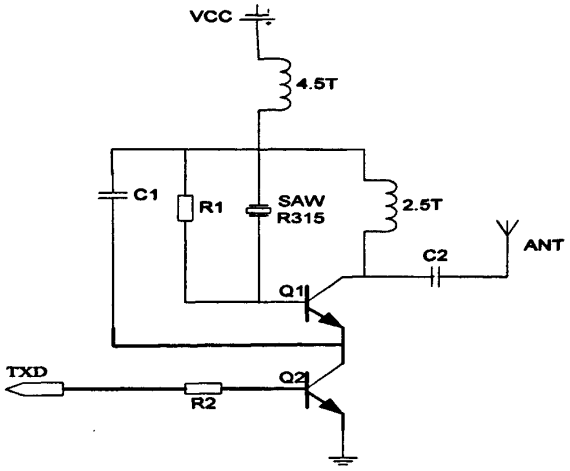


图4-1　DF数据发射模块原理图

本文配备的接收模块选用超外差式的DF接收模块。超外差式接收机的优点是频率稳定，抗干扰能力好，和单片机配合时性能比较稳定。DF模块有一种重要的用途就是配合单片机来实现数据通信。本文将51单片机与接收模块连接，当超外差式RX3310接收模块接收到信号后送51单片机处理，用51单片机来控制小车的相应动作[34]。由于只控制小车让其响应少量的语音指令动作，所以需要经DF收发模块传送的控制信息量比较少。我们用发送特定频率方波的方法代替普通的按照某种协议发送字节流的方法来让小车响应特定的动作。这样处理也能够很好的解决超外差接收机的灵敏度问题。即使由于超外差接收机的灵敏度低造成部分数据丢失，但持续发送的方波波形还是能够让接收端接收到所需要的数据。接收模块原理图如下4-2示。

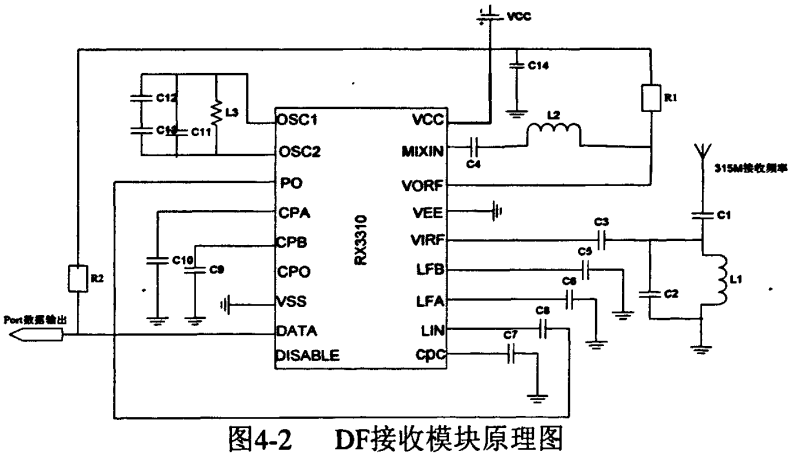


图4-2　DF接收模块原理图

那么在 SPCE061A 成功进行语音识别后，它就通过自己的 GPIO 口向 51 发送一个字节的数据。51 不断的扫描自己的 IO 口，在接收到 61 发来的数据后根据它所接收到的一个字节的数据来产生相应方波并送到无线发送端。在接收端除了接收模块外还需要一个 51 最小系统来控制外围电路驱动电机进行不同方向的运转。

在接收端除了上面的 DF 模块外，还有 51 对电机的控制部分。其电路原理图如下 4-3 所示[35]。

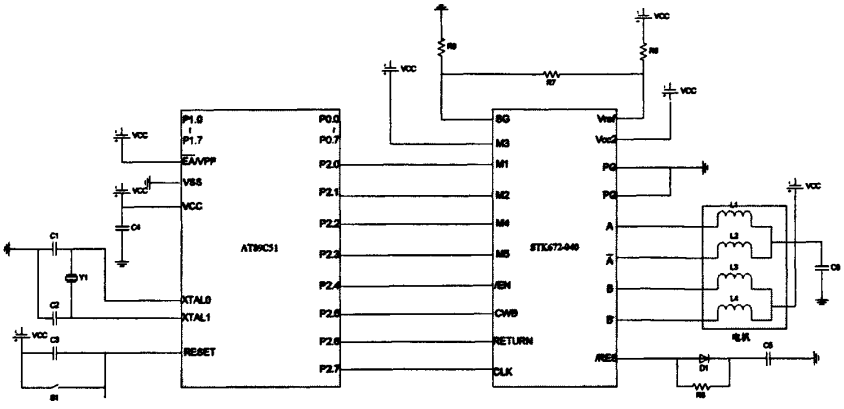


图4-3　电机控制部分原理图

4.1.2 语音识别系统的方案设计

具体模块与架构都已经在 4.1.1 节已作详解。那么对这个语音识别系统进行整体方案设计呢？首先系统进行语音训练，选取语音模板。将 SPCE061A 的语音口进行语音采集，在成功获取语音模板后存入 SPCE061A 的 Flash 中。依次获取模板，在建立完参考语音模板库之后系统开始进行语音识别。SPCE061A 在读取到待识别语音后，将待识别语音与参考模板语音进行匹配识别。识别成功后，SPCE061A 通过它的 GPIO 口向 51 发送一个字节数据，51 在扫描 IO 口数据获取到 61 发来的数据后，便开始产生方波。61 所发送的这一字节数据是根据所识别的结果来发送的，例如 10 个模板，若识别结果为数字 1 则在一个周期内发送一个高电平信号，依此类推，若是 0 则发送 10 个。51 在接收到这些信号后按照指定格式进行编码，传送给发射端[36]。在接收端接收到发射端传来的信号后，接收端将信号进行相应的处理。信号经过 51 解码后(这个解码规则可以根据所建立的模板库的数量来确定识别后所对应的识别结果，然后设定一个次序编号，则发送一个这样的编号数据，让 51 来进行解码)，51 便可以发出相应的方波来

控制小车的动作。因为小车所需要的控制指令不多，因此只需很小的频率段就能够满足指令数量的要求。例如 1-2KHZ 的频率段，以 200HZ 步进，即 10 种频率的方波便可满足系统的要求。具体的原理图如下图 4-4 所示。

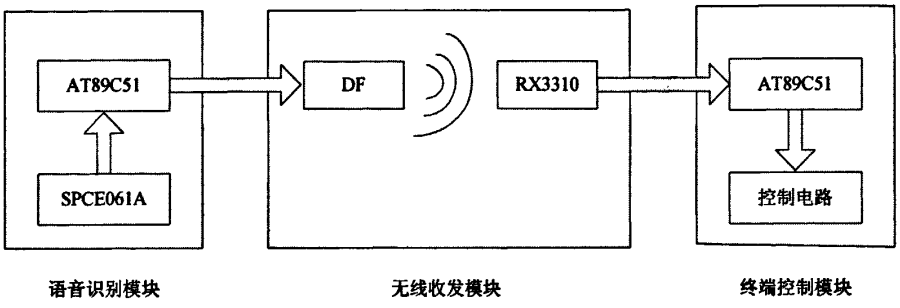


图4-4　基于RFID语音识别系统原理图

SPCE061A 负责进行语音识别处理，实时通过语音采集口来接收语音信息，一旦识别到有意义语音指令就进行语音匹配从而产生识别结果。它将识别结果送 51 后，51 对所接收到的数据进行编码并由发射端发送。中间由无线收发模块进行通信。接收端接收到的数据信息传给接收端的 51 系统进行解码处理[37]。51 在识别到处理方式后发送相应的方波信号给电机控制部分，最后由电机控制部分来控制小车的相应动作。

本文避免了将 SPCE061A 通过串口直接与无线发送端连接，因为 SPCE061A 发送控制方波到小车会造成系统误码率比较高，小车响应的效果会比较差。为了解决这个问题，本文加入了 AT89C51，让语音通过 61 识别后的结果送 51 进行编码处理送至发射端，这样就不会产生系统误码率较高的问题了。如果 61 识别到待测试语音在参考模板库中找不到相匹配的语音，那么就不会给 51 发送一个字节数据，而是会通过系统语音提示没有找到所匹配的语音。当然系统中仍然存在着会影响系统性能的因素。一般超外差接收模块比较灵敏，容易收到外界频率的干扰影响，所以最好我们能将接收段的 AT89C51 与接收模块隔离开，或者换 pic 单片机，这样就可以避免造成比较大的影响。但一般情况下只要超外差接收模块与高频单元不是太近通常不会造成比较大的影响。

其次便是电机控制这部分了，由于小车的动作所需电机的个数为 2 个，分别响应不同方向的动作，这样就需要通过 51 来控制不同的电机动作[38]。同样也可以加入部分外围电路来控制，具体的实现方法可见 4.2.2 节。51 对电机的控制部分框图如下图 4-5 示。

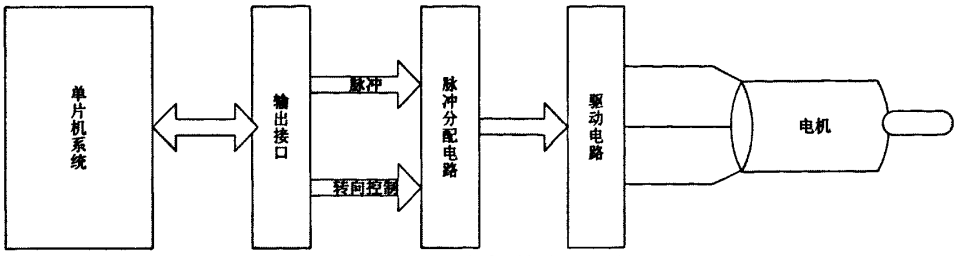


图 4-5　电机控制框图

4.2 语音识别系统的软件设计

虽然凌阳 SPCE061A 有自带的语音识别函数库，但本文采用自编的语音识别程序。将系统的语音识别模块程序单独的完成，系统要进行识别时自动调用这个程序模块就可以实现语音识别功能了。在 SPCE061A 上只需要完成控制功能。于是需要 SPCE061A 直接来完成的部分就比较少，这样可以提高系统速率。整个系统软件部分可分为语音识别部分、控制部分、响应部分。

4.2.1 语音识别模块的 C 实现

孤立词的每一个词都对应一个参考模型，这个模型是由这个词的多遍重复发音再经过特征提取而得到的。在识别阶段，系统把输入语音的特征矢量序列与参考模型进行匹配计算，取最接近的参考模板语音为识别结果[39]。在进行识别处理过程中可以设定一个最大匹配测度值，如果待测试语音与参考模板库中的参考模板语音的最小匹配距离值仍然大于所设定的值，则认为待测试语音不在所识别模板库的范围内，系统便不响应。

那么具体语音识别就如第 2 章所介绍的流程一样，如图 2-7 所示，首先建立模板库，在建立模板库之前首先得进行语音的预处理，然后进行特征参数提取，将特征参数存入指定数组中(当然这个数组的数据也可以写入到 SPCE061A 的 Flash 中存储起来，下次就没有必要再进行训练建立模板库了)。然后对待测试语音做同样的预处理与特征参数提取，然后将这些特征参数进行对照，计算匹配距离失真度，从中选取最佳匹配。本文选用的是 DTW 算法，找出最优路径进行匹配从而找出与测试语音匹配的参考模板语音。最终实现识别功能[40]。具体的函数功能见下表 4-1 所示。

表4-1　函数一览表

函数名	功能	输入	输出
ham(float *r,short *nad,int iw)	给信号加汉明窗	r 初始语音信号、nad 预处理后的信号、iw 窗的大小	0
correl(float*rsam,float*rcor,int iw,int ip)	计算信号的自相关函数	rsam 加汉明窗后的数据、rcor 自相关函数、iw 窗的大小、ip 阶数	无
coref(int ip,float *cor,float *alf,float *ref,float *resid)	计算线性预测系数和反射系数	ip 阶数、cor 自相关函数、alf 线性预测系数、ref 反射系数、resid 预测误差	无
alfcep(int ip,int n,float *cep,float *alf)	由 LPC 预测系数计算 LPC 倒谱系数	ip 阶数、n 倒谱系数阶数、cep 倒谱系数、alf 预测系数	无
cepmel(float *cep,float *mel)	由 LPC 倒谱系数计算 MEL 倒谱系数	cep-LPC 倒谱系数、mel-Mel 倒谱系数	无
distance(float *dsb,float *mb,int m)	求两帧数据的欧氏距离	不同文件中的两帧数据	求出的距离
min(float a,float b,float c)	求 DTW 最佳距离	求得的与两个模板库之间的距离	最佳距离值
dtw(float *dsb, int m,int n)	求得匹配的模板库的距离	最佳距离值	与模板库的最优距离

本文就将待识别的语音文件命名为 1.wav，设置的模板库文件为 2~4.wav。现在就可以进行识别对照，看待测试的语音信号到底与哪个语音信号匹配。由于参考模板库中的参考模板的特征参数提取方法与待测试语音的特征参数提取方法完全一样，所以若是在比较安静的实验室或者噪声环境不是比较嘈杂的情况下，待识别语音与参考模板的匹配距离应该为 0。本文在处理过程中所选用的窗长是 160 以及获取 MFCC 系数的阶数 12 都不变，而只是系统选用了在相对比较嘈杂的外界环境下进行语音识别。语音信号处理过程中仍然选择频率为 8kHz 的采样频率进行采样，帧长 160，帧移 40。经过试验统计，在本系统中，若待识别语音与参考模板语音的最小匹配距离超过 0.35，那么所识别的语音信号与参考模板库的语音便会有差别。所以在本程序中若最小匹配距离超过 0.35，系统便会认为没有与待测试语音匹配的参考模板。本文所用到的编译器是 C-Free 3.5[41]。语音信号匹配的结果如下图 4-7 所示。

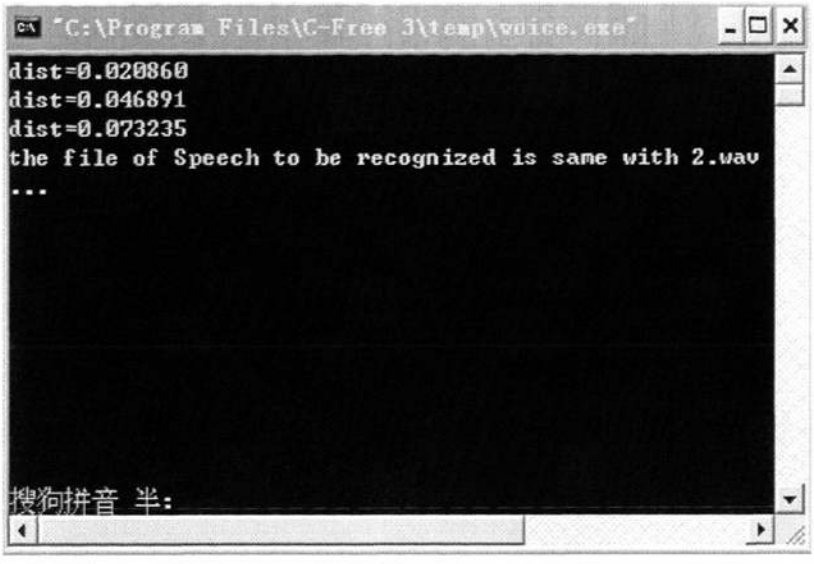


图 4-7　程序运行结果

4.2.2 语音识别系统的软件实现

整个系统软件部分的实现包括对训练与识别以及控制部分的程序实现。整个系统的流程图如下图 4-8 所示。

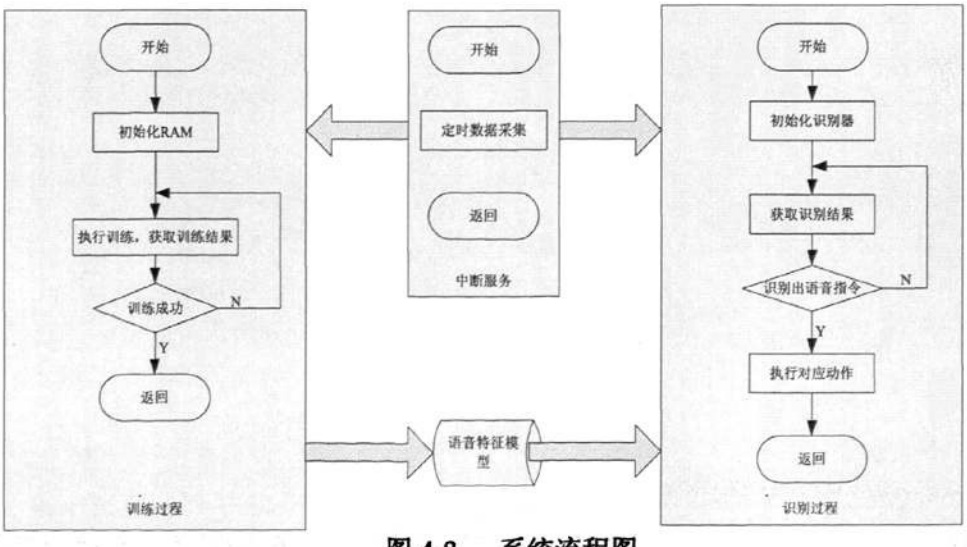


图 4-8　系统流程图

首先进行语音训练。在按下按键听到系统提示音后便开始进行训练，SPCE061A 读取到语音口采集到的语音信号。在训练成功后系统同样会有提示音

提示“训练成功”。依次进行语音的训练，待训练结束后，语音模板便已经建立。然后系统就开始进行语音识别。在识别环节中，存在着系统是否已经训练过的问题。为了避免重复训练，就需要首先判断系统是否训练过。如果系统已经训练过，那么 SPCE061A 会开始不断分析处理麦克风收到的语音信息，将它与参考语音模板进行对照匹配。若待测试语音识别成功 SPCE061A 则会向 51 发送识别结果，然后就开始接收和处理新的语音信号，进行下一次识别。当然在这一过程中，因为 61 语音口进行语音采集是不可能间断的，那么系统将 61 采集到的语音信号先存放于缓存中。待识别完成后就会自动从缓存中取数据。那么系统是如何判断小车是否已经经过训练？关键是利用了 SPCE061A 中 Flash 的一个特殊单元，它是语音存储区的首位置。Flash 在初始化以后，如果是没有进行过写操作的话，那么该单元的内容就会被设置为“0Xffff”，如果进行过训练并成功存储了语音模型的话，该单元会被设置为“0X0055”，这个值是由辨识器自动生成的[42]。 于是我们就可以根据这个单元的值来判断系统是否经过了训练。那么 51 在接收到识别结果后，会向射频发射模块发出相应的编码信号，待接收模块接收到后送 51 处理，从而来控制电机的运转，让小车响应语音的相应动作。那么在 SPCE061A 上实现的具体流程如下图 4-9 所示。

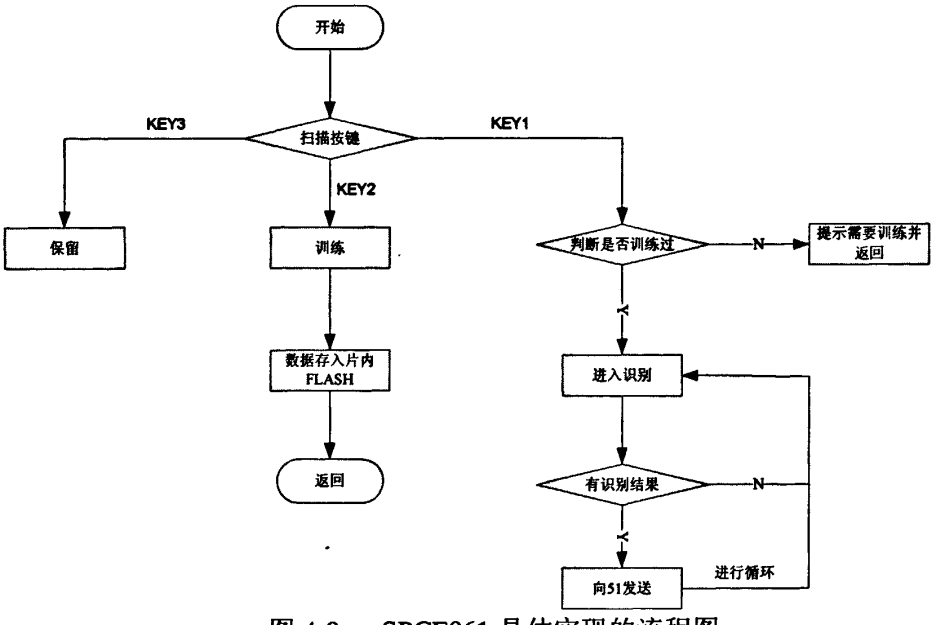


图 4-9　SPCE061 具体实现的流程图

由上图可知，本文将 KEY3 键预留以后做其它的用途，便于可再扩展。将 KEY2 键设置为用来进行语音训练的键。系统将训练成功后的数据存入 FLASH

中，以便后面识别时用。KEY1 键是用来进行识别的按键，同时本文还将它也定义为进行重新训练的按键。系统一旦运行之后就会不断地扫描 KEY1 这个键的按键情况，如果检查到 KEY1 键被按下，程序首先会把语音参考模板特征参数存储区的 0xe000~0xe100 单元这一页擦除，接着并会进入一个死循环来等待复位的到来。复位到来之后，如果程序检测到训练标志即 0xe000 单元内容为 0xffff 的话，它便认为小车是没有经过训练的，于是就会要求对它进行新的语音训练。如果 0xe000 单元内容为 0X0055 就会进入到识别过程，完成下面的语音匹配的任务。

然后便是如何来实现电机的驱动和控制，从而来实现由语音控制小车响应相应动作的功能。本系统所用的小车为双电机四轮驱动结构，左侧的两个轮子由左电机驱动，右侧的两个轮子由右电机驱动。单片机对小车的方向控制是通过一个 H 桥电路完成的。如图 4-10 所示，该 H 桥电路主要由三极管 Q2、Q3、Q7、Q8 组成，把 Q2、Q3 归为一组，Q7、Q8 归为另一组。另外还有两个辅助三极管 Q1、Q6，Q1 负责控制 Q2、Q3 的导通与关断，Q1 导通激发 Q2、Q3 导通，Q1 关断的同时 Q2、Q3 也关断。Q6 负责控制 Q7、Q8 的导通与关断，其工作过程同 Q2、Q3。

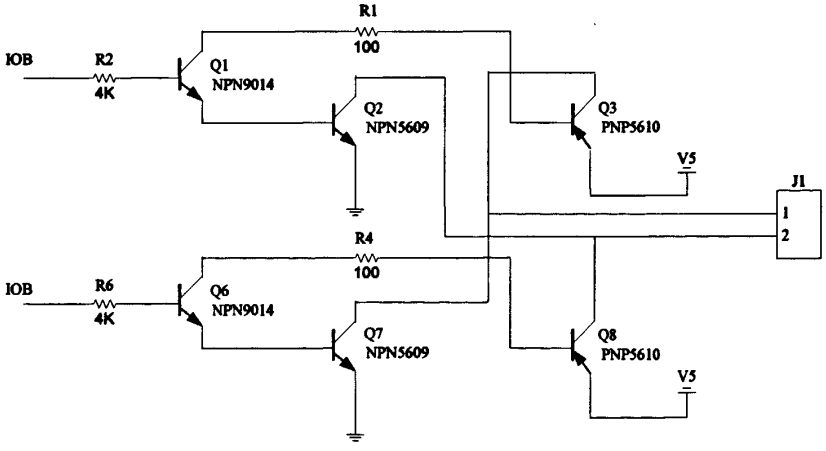


图 4-10　控制电机运转方向电路图

如果让 Q2、Q3 导通 Q7、Q8 关断，电流会流经 Q3、负载、Q2 组成的回路，加在负载 Load 两端的电压左正右负，如图 4-11 所示，此时电机正转，如果让 Q7、Q8 导通 Q2、Q3 关断，电流会流经 Q8、负载、Q7 组成的回路，加在负载 Load 两端的电压为左负右正，此时电机反转，对应图 4-12 所示。另外如果让

Q2、Q3 关断 Q7、Q8 也关断，负载 Load 两端悬空，此时电机停转。这样就实现了电机的正转、反转、停止三态控制。

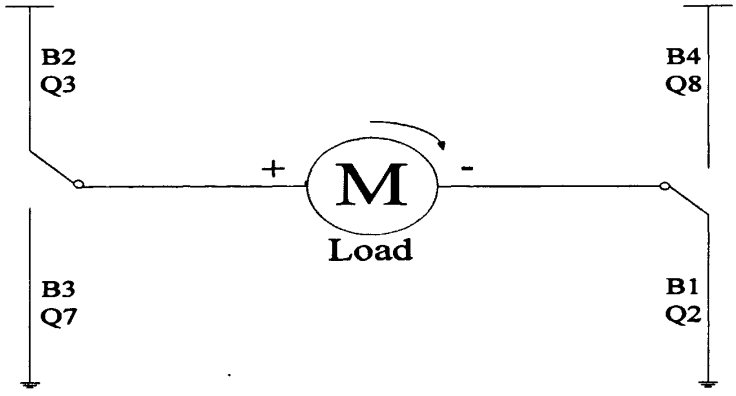


图4-11　B1、B2工作时的H桥电路图

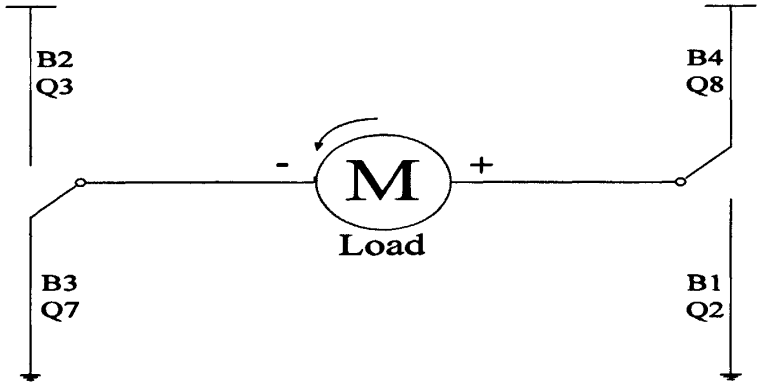


图4-12　B3、B4工作时的H桥电路图

由于 Q2、Q3，Q7、Q8 的导通和关断是通过 Q1、Q6 控制，而 Q1、Q6 的导通和关断又是通过 IOB10、IOB11 控制的，所以电机的状态还是通过 I/O 端口来控制的。表 4-2 描述了 IOB10 和 IOB11 所控制电机运行状态与端口数据的对应关系。

表 4-2　I/O 端口状态与电机运行状态的对应关系

IOB11—IOB10	Q1	Q6	Q2、Q3	Q7、Q8	电机
00	关断	关断	关断	关断	停转
01	导通	关断	导通	关断	反转
10	关断	导通	关断	导通	正转

我们对小车的左右轮的转动状态进行编码，用 4bit 数据来控制小车的运动方式。对控制板的输入端口 IO 口的 B 端口的 10 位至 13 位来写入数据便可以实现对小车的左右车轮的电机的完全控制。下面是对小车运行状态和 IO 口数据的编码表，当然其它的编码则可以预留以后的扩展或者是增加新的功能。

表 4-3　小车运行状态和 IO 口数据的编码表

IOB10—IOB13	左电机	右电机	小车
0000	停转	停转	停止
0001	反转	停转	右后转
0010	正转	停转	右前转
0100	停转	反转	左后转
1000	停转	正转	左前转
1001	反转	正转	逆时针旋转
0110	正转	反转	顺时针旋转
1010	正转	正转	前进
0101	反转	反转	后退

然后将程序下载到板中，使用 PROBE 口进行烧写至控制板中。系统所制定的语音口令信号是可以直接对小车说前进，或者倒车、左拐、右拐等的语音指令。小车如果识别出指令会有一个回应信号，告知主人它要执行的动作，然后执行该动作，动作执行完毕后小车会停下来，并结束待命状态。

4.3 车载语音识别系统的模拟

本系统是如何对车载语音识别系统进行模拟的呢？首先考虑到汽车所工作的环境以及汽车本身的性能要求，本系统就必须结合实际的车载环境来进行相应的处理如噪声隔离以及系统与外界的互干扰处理等等。语音采集这部分不需要做什么大的变动，仅仅只是需要在语音输入段前后做一些相应的去噪滤波处理便可。车载语音识别系统的开启也必须如本系统一样设置按钮，不过仅仅只是起到系统的开关作用而已。那么仅仅一个按钮我们又如何进行语音训练的呢？这就需要在最初的情况下，也就是在买车的时候进行训练。当然系统可以设置多套语音模板，这样就可以进行多人识别了。因为车主或许希望能让自己的家人也能拥有此项控制语音的权利。那么针对不同的控制人我们可以设置不

同的语音模板。在识别时可以将多套模板库进行编号分类，将待测试语音与每套模板中的参考语音模板进行匹配，如果获取到最小值则匹配。而且我们同样可以在以后的识别过程中，设置更换小车主人的语音指令以及增加小车主人的语音指令。对于更换小车主人这样的语音指令，系统的处理方法是将语音识别系统进行硬复位，这样语音识别系统就必须进行重新训练。系统在第一次买车时进行语音训练之后只要是系统硬复位了，系统就会自动的提示进行语音训练，训练成功后就会将语音模板存储起来，这样以后识别便不需要再进行训练。在语音训练时需要增加“更换主人”这条语音的训练，增加了这条语音指令系统就可以响应小车主人这样的语音指令与操作。对于“增加主人”这条语音指令，系统首先会检查内存，如果内存已满，系统会做出提示。如果未满，且有足够的存储空间存储另一套语音模板库的话系统便提示“开始训练”的语音信号。训练成功便存储待以后识别时用。当然在更换主人的时候，“增加主人”的这条语音信号同样需要训练。当然在系统处理过程中，系统会逐步提示小车主人对需要训练的语音指令做相应的训练的。这样车载语音识别系统就可以简化语音训练达到语音训练既简单又方便的功能。若一旦语音模板库太多，则语音有很多套参考模板这样就会影响到系统整体的处理速率。最后对于车载语音识别系统可以用到 can 总线进行通信，将语音处理模块与识别模块连接起来，这样既可以消除电子部件之间的相互干扰，又能提高系统响应速率。对于车载语音识别系统对电机的驱动与控制方法与本文所设计的系统解决办法一样。在实现了电机控制和语音识别功能后，整个车载语音识别系统便能通过系统 can 总线进行通信，最终实现语音识别系统的整个功能。

4.4 本章小结

本章主要就语音识别系统的硬件与软件设计部分作了分析与介绍。硬件部分主要包括语音识别部分、射频通信部分、电机控制部分。软件部分主要包括对语音识别过程的程序实现，首先是训练部分，其次是中断控制服务部分，最后是识别响应部分。最后本文就本系统对车载语音识别系统的模拟作了分析与思考。

第 5 章　总结与展望

5.1 论文工作总结

本文所设计的系统是基于特定人小词汇量的语音识别系统，本文对传统 DTW 算法做了改进，重点研究了对端点检测的优化方法，结合可变窗长和双门限的语音端点检测方法，通过 MATLAB 仿真将传统端点检测方法检测出来的结果与改进后的端点检测方法检测出来的结果进行了对比分析，然后对短时能量与短时过零率选取不同的加权系数，在改进后的端点检测方法的基础上作了进一步优化。在语音识别匹配的处理过程中，本文采用了整体路径约束的 DTW 算法，将匹配路径约束在斜率为 1/2 和 2 的平行四边形区域内。而且本文结合松弛起终点的 DTW 算法，采取不固定匹配路径的起点与终点的方法来进行语音的匹配，这种松弛起终点的方法对匹配路径的选取更具有一般性。最后对语音匹配采用模糊算法进行识别处理，将待识别语音的 MFCC 参数与参考模板的 MFCC 参数进行二次模糊度计算，找出与待测试语音匹配距离最小的参考语音模板作为识别结果，通过实验仿真得到改进后的算法对语音信号的处理结果。通过 Matlab 的 Simulink 工具设计了对语音采集与去噪，获取到语音信号的短时平均能量与短时过零率以及语音信号的特征参数的仿真模型。在系统设计上本文采用 SPCE061A 进行语音的采集以及模板的存取，并调用本文已经编制好的语音识别程序模块进行语音识别，然后将识别结果作为返回值返回，等识别结束后 61 便通过它的 GPIO 口向 51 板发送一个字节的数据，51 在扫描 IO 口数据获取到 61 发来的数据后，根据 61 发来的一个字节的数据开始产生相应的方波。因为小车控制指令不多，只需很小的频率段，例如 1-2KHZ 的频率段，以 200HZ 为步进，即 10 种频率的方波便可满足系统的设计要求。然后信号经过 DF 收发模块，接收端收到信号后送接收端的 51 板来处理。在对小车的控制部分，采用 H 桥电路用单片机来控制电机的运转从而实现对小车运转方向的控制。因为小车的运行状态比较少，不同频率的方波与 51 单片机的 IO 口数据的编码相对应。而小车的动作又与 IO 口的编码数据相对应。本系统便是通过以上的方法来实现语音控制小车的功能。现将主要工作总结如下：

（1）本文介绍了动态时间规整(DTW)、隐马尔可夫模型(HMM)、人工神经

网络(ANN)。比较了这三种语音识别方法的优缺点。对于本文所研究的特定人小词汇量语音识别系统，DTW 更加适合。因为 DTW 算法简单而且无需进行大量的训练。对于特定人小词汇量的车载语音控制系统，DTW 已能满足要求。

（2）首先是对传统 DTW 算法作了改进与优化，重点介绍了对端点检测方法的优化处理。同时将整体路径约束的 DTW 算法与松弛起终点的 DTW 算法相结合，就改进后的 DTW 算法与传统 DTW 算法进行了比较。改进后的 DTW 算法优于传统 DTW 算法。

（3）本文采用 SPCE061A 进行语音识别处理，并在硬件上结合 51 与射频模块来实现对小车的语音控制功能。系统选用自制的语音识别功能程序。该程序采用改进后的 DTW 算法来实现语音识别功能。若待测试语音通过 SPCE061A 识别到的结果为参考模板中的第 6 个模板语音。而系统将 2kHZ 的频率段划分为 10 段，取 200HZ 为步进。那么在发射端的 51 单片机会给 DF 发射端发送基频 +200*6HZ 频率的方波信号。然后在接收端的 51 端进行解码分析，将频率划分的 10 段与它的 IO 口的编码相对应。而 IO 口的编码正好对应小车的相应动作。通过这种方法来实现语音对小车的控制。

5.2 工作展望

本文研究对象是特定人小词汇量的语音识别系统。本系统还不够完善，仍然需要进一步的改进与优化。现就工作中出现的问题，本文提出以下几点改进意见。

（1）对于 DTW 算法，除了在端点检测方面和匹配距离计算方面的改进外，还需要对 MFCC 系数的获取方法进行改进。针对所采样的窗长以及帧移，不同环境与情况下选取最合适的窗长来进行处理。

（2）若希望语音识别功能能更好的实现则需要对噪声以及人的生理发音进行研究。对系统抗噪能力进行改进，因为车载语音识别系统是工作在噪声比较大的环境中，这就需要系统即使在噪声情况比较恶劣的环境下也能进行精确的语音识别。

（3）对语音训练方法的改进，找到更简单与灵活的训练方法。

（4）选取更优的硬件系统来实现系统抗干扰能力强，稳定性能高的语音识别功能。同时设计一种方法能更好的将语音识别系统与车载设备进行 can 总线连接。

致 谢

首先要非常感谢我的导师黄涛老师，在对本论文所提到的语音识别系统实现的过程中，黄老师给了我很多的帮助和指导。黄老师他不厌其烦的引导我完成系统每个部分的设计，并指出在系统实现的工作过程中应该注意哪些问题和应该应用到什么关键的技术。而且在论文完成的过程中，黄老师更是指导了我撰写的整个思路和流程。

黄老师他治学严谨，学识渊博，工作积极认真。在平时的学习研究中，黄老师都很注重积极创新。在三年的时间里，黄老师都一直悉心的帮助和指导我们。他对学生的关心和对教学事业无私的奉献，深深地感动着我。他那严谨的治学态度以及不畏艰难的钻研精神给我留下极为深刻的印象。他的教导和勉励时刻提醒和激励着我要刻苦学习和努力工作。

同时要感谢同组的卢珞先和廖传书老师，感谢他们在我平时的学习和研究中所给予的指导和帮助，感谢他们对我的关心和支持。感谢和我一个实验室的所有同学，感谢他们在学习上的给我的帮助和鼓励。

衷心感谢答辩委员会的所有老师，感谢你们为评阅本论文而付出的点点滴滴！感谢硕士三年来我所有的老师，感谢他们对教育事业呕心沥血的奉献。同时还要感谢信息工程学院的所有老师，感谢他们对我的支持和帮助。最后感谢所有给予我帮助和鼓励的同学和朋友。

参考文献

[1] 陈尚勤等. 近代语音识别. 成都：电子科技大学出版社，1991

[2] 马俊. 语音识别技术研究. 哈尔滨：哈尔滨工业大学出版社，2004

[3] Vincent G. Duffy, Richard Linn & Ameersing Luximon, Voice Recognition Based On Human-Computer Interface Design. Computers & Industrial Engineering, 1999, Vol.37:300-306

[4] 冯坚. 下一代车载通讯系统. 汽车与配件，2007，vol.18:24-28

[5] 刘旺，杨殿阁，连小珉. 车载导航人机语音交互系统的实现. 电子产品世界，2007，V01. 5：127-130

[6] 刘晓辉，陈启军. 基于语音识别的车载导航系统研究. [硕士学位论文]，同济大学，2008年

[7] Kondoz AM. Digital Speech-Coding for low bit rate communication systems. IEEE Press, 2005(2):840-842

[8] 姚天任. 数字语音处理. 武汉：华中科技大学出版社， 1992. 60-93

[9] K.H.Davis, R.Biddulph, S.Balashek. Automatic Recognition of Spoken Digits. Acoust. Soc. Am , 1952, 24(6)：630-640

[10] 陈杰，张玲华. 说话人识别中语音特征参数的研究. 信息技术. 2006，Vol30 Nol1.

[11] 胡航. 语音信号处理. 哈尔滨:哈尔滨工业大学出版社，2000. 167-169

[12] 王炳锡，屈丹，彭煊. 实用语音识别基础. 北京：国防工业出版社，2005

[13] Chen Jing dong(Bell Laboratories, Lucent Technologies). Cepstrum derived from differentiated power spectrum for robust speech recognition. Speech communication. 2003.

[14] 黄德智，杨鸿武，蔡莲红. 语音信号的加权 Mel 倒谱分析. 信号处理. 2006，vol22 N06.

[15] Dupont, Stephane, Cheboub, Leila. Fast speaker adaptation of artifical neural networks for automatic speech recognition. IEEE International Conference on Acoustics, Speech and Signal Processing-Proceddings, 2000

[16] 汤升庆. 车载语音识别的应用设计，[硕士学位论文]，武汉理工大学，2007

[17] 杨建刚，王伟臻．基于神经网络的语音识别研究．[硕士学位论文]，浙江大学，2008年

[18] R.P.Lippman. Review of neural networks for speech recognition. Neural Computation，1989，1(1)：1-38

[19] 夏峰，陆珂伟，陈启军．语音控制的多功能车载终端系统的设计与实现，[学术论文]同济大学，2008

[20] Deherty J , Porayath R. A robust echo canceller for acoustic environments. IEEE Trans On Circuits and Systems, 1997 , 44:389 -398.

[21] Biing-H Juang，Sadaoki Furui. Automatic Recognition and Understanding of spoken Language，A First Step Toward Natural Human Machine Communication. Proceedings of the IEEE.2000，88(8):1140-1168

[22] D.R.Reddy. An Approach to Computer Speech Recognition by Direct Analysis of Speech Wave，Tech. Report No. C549，Computer Science Dept.，Stanford Univ.，Sep. 1966：143

[23] 林波，吕明．基于 DTW 改进算法的孤立词识别系统的仿真与分析．信息技术，2006年第 4 期

[24] Erdogan AT , Kizilkale C. Fast and low complexity blind equalization via sub gradient projections. IEEE Transactions on Signal Processing, 2005 , 53 (7):2513 - 2524.

[25] O'Shaughnessy，Douglas. Efficient automatic speech recognition. ASTED International Conference on Internet and Multimedia Systems and Applications，2004.

[26] 郑阿奇．MATLAB 实用教程．北京：电子工业出版社，115-247

[27] 江官星，王建英．一种改进的检测语音端点的方法．微计算机信息期刊，2006 年第 22 卷第 5-1 期

[28] Ming Dong，Jia Liu and Run sheng Liu. Speech Interface ASIC of SOC Architecture For Embedded Application, International Conference on Signal Processing，2002(3)：19-23

[29] 崔光照，吴晓平，路康．基于改进的 DTW 算法的仿真与分析．福建工程学院学报，第 2 卷第 2 期

[30] 刘长明，任一峰．语音识别中 DTW 特征匹配的改进算法研究．中北大学学报，2006年第 1 卷第 27 期

[31] 钟珞，何平．模式识别．武汉：武汉大学出版社，2006 年 122-136.

[32] 凌阳公司．SPCE061A.pdf

[33] 王茜，姚娅川．基于 SPCE061A 单片机的语音识别系统开发．四川理工学院学报(自然科学版)，2005．(01)26-28

[34] 胡汉．单片机原理及其接口技术．北京：清华大学出版社，1996

[35] 何立明．MSC-51 系列单片机应用系统设计．北京：北京航天航空大学出版社，1995

[36] Sujay.P，Rhishikesh.L，Siddharth.V. On design and implementation of all embedded automatic speech recognition system. IEEE Transactions on VLSI Design，2004 21(3)：127-132.

[37] 李晶皎．嵌入式语音技术及凌阳 16 位单片机应用．北京：北京航空航天大学出版社，2006

[38] 郑建光，金碧波，章　皓．基于 8051 单片机语音控制系统的实现．自动化与仪器仪表，2006 年 第 2 期

[39] 孙振安，孙捷．小词汇量非特定人语音识别在嵌入式系统中的应用．计算机工程，2006.

[40] T.K. Vintsyuk．Speech Discrimination by Dynamic Programming，Kibemetika，Jan. -Feb，1968 4(2):81-88

[41] Kirk Zurell．嵌入式系统的 C 程序设计．北京：机械工业出版社，2002

[42] 杨占军，杨英杰，王强．基于 DSP 的语音识别系统的设计与实现．东北电力大学学报，2006 年 4 月 第 26 卷第 2 期