基于层级策略的连续数字串识别的研究

汤　霖[1]　蔡莲红[2]
[1]（江门教育学院计算机系，广东江门 529000）
[2]（清华大学计算机系，北京 100084）
E-mail：China-tl@163.com

摘　要　在分析汉语数字串语音特点的基础上，设计出了基于层级策略的连续数字串识别系统。该系统先对连续数字串进行确定性的预分割，再用 Level Building 算法对每个分割段进行基于模板模糊分组的识别，在该识别结果的基础上利用加权矩阵识别算法进一步区分易混淆语音对。该系统在计算时间减少到原来的 35.2%的同时识别率提高到 94.08%。

关键词　语音识别　汉语数字串　分割

文章编号 1002-8331-（2003）21-0083-04　文献标识码 A　中图分类号 TP391.42；TP912.34

Mandarin Connected Digit Speech Recognition System Based on Hierarchical Method

Tang Lin[1]　Cai Lianhong[2]
[1]（Department of Computer Science and Technology，Jiang Men Teacher's College，Jiangmen，Guangdong 529000）
[2]（Department of Computer Science and Technology，Tsinghua University，Beijing 100084）

Abstract：By analyzing the characters of mandarin connected digit speech，a recognition system based on hierarchical method is given.In this system，the first step is to find the correct dividing point in digit string，then each segment is recognized by using level building method based on fuzzy digit pattern group，finally the weighted matrix algorithm is used to distinguish the confusable digit pair.Experiments demonstrate that the correct recognition rate of the system is improved to 94.08%meanwhile the recognition time is decreased to 34.2%.

Keywords：speech recognition，mandarin connected digit，segmentation

1　前言

连续数字串识别系统在语音拨号、声控留言、200 号业务以及身份鉴别等领域有着广阔的应用前景，因此，受到众多研究机构的重视[2][5][7][9]。连续数字串识别系统虽然是一个小词汇量识别系统（只有 10 个数字），但它的识别没有语法知识可利用，从而缺少语言模型对识别系统的支持，这对识别系统提出了更高的要求，同时也为研究语音参数的表达能力、鲁棒性，识别方法的有效性等提供一个良好的机会。实用环境中的连续数字串识别系统对抗噪性、鲁棒性、自适应性等都提出了很高的要求，这些要求促进了对连续数字串识别系统的各方面的研究。

在国外，英语的连续数字串识别系统早已取得了 99%[1]的识别率，而汉语的连续数字串识别的最好水平也就 98%[2]左右。这主要是因为汉语数字串语音的混淆度比英语数字串语音高，同时汉语数字串发音连续程度也高于英语数字串发音[3]，因此，提高汉语数字串识别系统的识别精度就必须提出针对汉语数字发音特点的识别方案来。

2　汉语数字串语音特点

数字的普通话发音通常有 11 个，即 0（ling），1（yao、yi），2（er），3（san），4（si），5（wu），6（liu），7（qi），8（ba），9（jou）。其中，包含 7 个声母（包括零声母）、9 个韵母。

在声母中，b 为爆破音，在语音波形图中可见到明显的成阻无音段，但有除阻不完全的现象，即在 b 的前面常有短促的噪声。

s 为摩擦音，q、j 为塞擦音，从时域特征看，这些音表现出非常高的过零率，但过零率受噪音高低、环境噪音等影响非常大，而频域特征对这些影响表现出一定的稳定性。在频谱图中可观察到这些擦音的频率主要分布在高频上。（主要能量都出现在 3500Hz 以上）。

在连读时，以上特性有较大的削弱。如连音的‘8’前有余波，‘7’、‘9’的阻塞变弱等。

同一数字在不同连音中，发声部位可能随前后音的不同而发生变化，如读‘28’，在快速连读时，因舌头不能由‘er’很快回位到‘a’，‘28’中‘8’的舌位就与单独发音时不同。另外，连音中数字的声调也因前后数字的不同而发生变化。这些是造成识别系统中‘2’与‘8’，‘6’与‘9’，‘0’与‘6’等混淆的主要原因，该类错误占系统识别错误的 75%以上。

数字中零声母的出现又带来了元音相连问题。如‘55’、

作者简介：汤霖（1962-），男，湖南醴陵人，讲师，硕士，主要研究方向：语音识别与处理。蔡莲红（1945-），女，北京人，教授，博导，主要研究方向：语音合成、语音处理、分布式人机交互。

‘22’、‘11’、‘71’等。这些连音，在波形图中无明显分割点，在语音特征上，也没有明显的分割信息，因此，经常分别误识为‘5’，‘2’、‘1’、‘7’等。这也是识别系统中，删除错误远多于插入错误的原因。

可喜的是，在汉语数字串中，各种连音的组合只有 100 个左右，可能的状态不太多。在训练数足够多时，可用穷举覆盖所有连音，而且，可分别针对不同连音采用不同的识别策略。如‘55’可用音调的不同来区分，‘22’可用能量峰的个数去辨别等。另外，还可将‘11’、‘22’、‘71’等整个连音作为模板。

3 系统组成

人们在念长数字串时，一般都是将长数字串分成 3 个数字或 4 个数字为单元来读，各单元之间存在停顿。如念 62774141：人们通常读成 6277、4141 等。因此，识别串长为 6 的连续数字串识别系统可涵盖各种长度的数字串识别任务了，片面追求超长数字串的识别没有实用意义，该系统就是以识别串长为 6 的连续数字串为目标。

连续语音识别主要采用两种技术：一是在识别阶段进行动态分割，直到动态搜索完毕，才能确定切分点，如 Sakoe 的 Two-Level 动态规化匹配法，Myers 和 Rabiner 的 Level-Building 动态时间伸缩法，Vintyuk 的 One-Stage 动态规划算法等；二是采用预切分，即预先将数字串一个个切分开来，再进行识别，这种技术的关键是采用什么准则进行切分，特别是在连音现象严重的地方。第二种方法比第一种计算量少几个数量级，但一旦分割错误，识别也必然错误，而且，对于汉语连音目前也没好的切分方法。因此，第二种方法很少使用。

系统采用预分割后再识别的办法来减少计算量。系统的预分割并不是将数字串中所有数字分割开来，而是将数字串中的确定性分割点分割开来。也即：分割点是 100%正确的语音分割点。经预分割后，整个数字串的识别就变成了几个短的数字串的识别。分割后的各语音段分别用 Level Building 算法识别出各段的最佳数字串，最后形成总的数字串。识别部分采用了两级识别的办法，第一级用 Level Building 算法识别出基本数字和确定分割点，第二级通过加权矩阵算法对上一级识别结果中易混淆的数字再进一步区分。系统框图如图 1 所示。

图 1 系统框图

在 Level Building 算法中，识别的过程可用图 2 (a)表示，设数字串有 N 帧，可以看出计算量正比于 N^2。分割成两段后，计算量最多时减少到 $N^2/4$，见图 2 (b)。

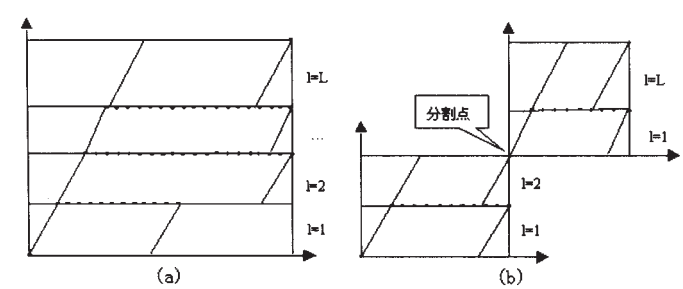

图 2 分割前后计算量的比较

4 特征提取

许多实验结果验证，MFCC 参数在语音识别中优于其他参数。这里也使用 MFCC 系数做语音特征参数。语音信号由 16KHz 采样 16bit 量化，经 $1-0.97z^{-1}$ 预提升，每帧为 512 点，帧移 256 点。

汉语数字语音中 4 个擦音的频率主要分布在 3500Hz 以上，为此，设计了一个高低频能量比参数，既：

$$R=\frac{\sum_{f_i>3500} f_i}{\sum_{f_i<3500} f_i}$$

其中 f_i 为某帧语音经 FFT 后的第 i 个频点。该参数可随 MFCC 的计算一同计算出来，同时该参数已对能量规一化。该参数对分割和模板选取起非常重要的作用。

基音在‘2’‘8’的分辨和‘55’的识别等起着非常重要的作用[4]，系统也将由改进的 AMDF 法计算出的基音数据经平滑后加入到语音特征参数中。

系统的特征参数用了 12 阶 MFCC 系数，相应的 MFCC 一阶差分和 MFCC 二阶差分，另外，加上规一化能量、一阶差分能量、二阶差分能量，以及上面的 R 参数及基音，共 41 维。

5 预分割

从前面的分析中，可以看到：爆破音和塞擦音的前部一般都有一个成阻阶段，在时域波形上，可观察到一段无音段。该无音段似乎可作为数字串中的确定分割信息。但是，在其它的观察中发现：有人念‘9’时，在[ou]的中间也有一段无音段，最长达 27ms，见图 3。

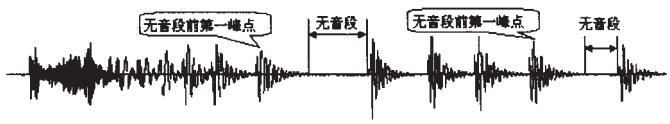

图 3 9 的语音波形

如果能有效区别这两种无音段的不同就可以用真正的无音段作数字串中的确定分割信息。从图 3 中，可以看到，‘9’中的无音段前语音波形幅度成衰减状态，为此，设计了 3 个参数：一是无音段前第一峰点到无音段的平均幅度之比，二为幅度衰减 10 倍所用的时间，三为第一峰点附近的能量与峰前能量比。通过设置经验阈值可以用这三个参数来判断此无音段是否为‘9’。

另外，数字中 4 个摩擦音在频谱上的表现也可以作为确定性分割的依据，摩擦音的特征 R 一般超过 10.0，而元音都在 0.10 以下，该参数有较强的区分能力，因此，设定一阈值就可以确定摩擦音的起始点，而该起始点就可作为确定性分割点。

由以上两条性质，可标定数字串的确定分割点。分割算法如下：

①当能量小于某域值时，为可能的分割点，跳到②，否则跳到③。

②判断该分割点是否是‘9’，非‘9’即为分割点，跳到④。

③特征 R 是否超过 10.0，是则为擦音分割点。

④语音结束否？，结束则停止分割算法，否则跳①继续。

6 识别

在 DTW 算法中，识别的准则为：

$$\underset{n}{\operatorname{argmin}}\{\sum_{i}\sum_{j=1}^{M}[x(i,j)-c(i,j,n)]^2\} \quad (1)$$

$x(i,j)$ 是待识样本第 i 帧的第 j 维特征，$c(i,j,n)$ 是第 n 个模板第 k 帧的第 j 维特征，M 为特征的维数。从公式中可以看到，每维特征对欧氏距离的贡献是一样的。但实际中的每维特征对不同音素的表达能力是不一样的，特别是对最小对来说。特征表达能力的不同表现在特征维上和时间轴上。为了强调特征的不同表达能力，可用加权矩阵同时在时间轴和特征轴上对每一帧的每一维特征加上不同的权重来强调特征对不同音素的区分性能。也就是将（1）式改为：

$$\underset{n}{\operatorname{argmin}}\{\sum_{i}\sum_{j=1}^{M}[(x(i,j)-c(i,j,n))^2\times w(i,j,n)]\} \quad (2)$$

加权矩阵 $w(i,j,n)$ 是通过训练过程得到的。如果对所有样本都训练一个加权矩阵将增大几倍的存储量同时增加大量的计算时间。在文章实现的系统中，采用两级识别算法，第一级用 Level Building 识别出最佳匹配数字和确定各数字的分割点，第二级识别才用加权矩阵算法对各易混淆数字对精确识别。第二级识别的关键在于由训练得到易混淆的数字对之间的加权矩阵 $w(i,j,n)$。

前面已讲到，在汉语数字串中，R 参数对 4 个摩擦音非常敏感，因此，在识别时，可以利用该参数的特点，将数字模板分为三类，一类 R>10.0：为 3，4，7，9。第二类为 R<0.10：0，1，2，5，6，8。第三类 0.10<=R<=10.0：为 1，6，9，在识别时根据 R 特征选择相应的模板类，由此，计算量可减少一半以上。

7 训练加权矩阵

汉语数字串识别中，大多数错误集中在 2-8，0-6，9-6，71-7，55-5，22-2 等。因此，对这些易混淆数字对可通过加权矩阵来区分，也即对每对易混淆数字对训练 4 个加权矩阵，如对 2-8，要训练 2-2、2-8、8-2、8-8 四个加权矩阵。训练算法如下：

设待识样本为 a，帧长为 N。识别结果中待识样本与非本类的最小距离模板为 b，待识样本与正确类的最小距离模板为 c，设 b 对 a 的加权矩阵为 $\sigma_{ba}(M,K)$，c 对 a 的加权矩阵为 $\sigma_{ca}(M,L)$，K 为 b 的帧数，L 为 c 的帧数。X 为易混淆语音对总数。

i.将分好类的训练样本用聚类法形成各类初始模板。将易混淆语音对中所列元素的加权矩阵的所有元素置 1，设置参数 θ 根据多次实验确定，一般取 0.3），x=1。

ii.设置最大循环次数 LOOP。

iii. 用第 x 对易混淆语音对所对应的所有同类训练样本进行识别实验。

当：$D_{DTW}(a,b)-D_{DTW}(a,c)>\theta\times D_{DTW}(a,c)$ 时，跳到 iv。

当：$|D_{DTW}(a,b)-D_{DTW}(a,c)|<=\theta\times D_{DTW}(a,c)$ 时，

也就是：

$$|\sum_{i=1}^{N}\sum_{j=1}^{M}[d_{DTW}(a(i,j),b(i,j))]\sigma_{ba}(i,j)-\sum_{i=1}^{N}\sum_{j=1}^{M}[d_{DTW}(a(i,j),c(i,j))]\sigma_{ca}(i,j)|<=\theta\times\sum_{i=1}^{N}\sum_{j=1}^{M}[d_{DTW}(a(i,j),c(i,j))]\sigma_{ca}(i,j)$$

计算：

$$D_e=\sum[d_{DTW}(a(i,j),c(i,j))\sigma_{ca}(i,j)-d_{DTW}(a(i,j),b(i,j))\sigma_{ba}(i,j)] \quad \forall d_{DTW}(a(i,j),c(i,j))\sigma_{ca}(i,j)\geq d_{DTW}(a(i,j),b(i,j))\sigma_{ba}(i,j)$$

$$D_r=\sum[d_{DTW}(a(i,j),b(i,j))\sigma_{ba}(i,j)-d_{DTW}(a(i,j),c(i,j))\sigma_{ca}(i,j)] \quad \forall d_{DTW}(a(i,j),c(i,j))\sigma_{ca}(i,j)\leq d_{DTW}(a(i,j),b(i,j))\sigma_{ba}(i,j)$$

当 $D_r\geq\eta\times D_e$ 时（η 由经验确定，一般取 0.5），认为参数可调。否则，跳到 iv。

设 α 为加权矩阵的调整因子，有：

$$D_e\times(-\alpha)\leq D_r\times(+\alpha) \quad \text{即：}\alpha\geq\frac{|D_e-D_r|}{D_e+D_r}$$

一般取：$\alpha=\frac{|D_e-D_r|}{D_e+D_r}+0.05$

调整加权矩阵：

$$\begin{matrix}\sigma_{ba}(i,j)=\sigma_{ba}(i,j)\times(+\alpha)\\ \sigma_{ca}(i,j)=\sigma_{ca}(i,j)\times(+\alpha)\end{matrix} \forall d_{DTW}(a(i,j),c(i,j))\sigma_{ca}(i,j)\leq d_{DTW}(a(i,j),b(i,j))\sigma_{ba}(i,j)$$

$$\begin{matrix}\sigma_{ba}(i,j)=\sigma_{ba}(i,j)\times(-\alpha)\\ \sigma_{ca}(i,j)=\sigma_{ca}(i,j)\times(-\alpha)\end{matrix} \forall d_{DTW}(a(i,j),c(i,j))\sigma_{ca}(i,j)\geq d_{DTW}(a(i,j),b(i,j))\sigma_{ba}(i,j)$$

iv.当所有样本计算完毕跳到 v，否则继续执行 iii。

v.当循环数小于 LOOP，并且调整后正确识别的样本数在增加或持平继续执行 iii。否则执行下一步。

vi.当 x≤X 时跳到 ii，否则训练结束。

8 结果与讨论

8.1 实验条件

由录音平台，在普通微机实验室中录制了 6 男、4 女各 80 个数字串，含 250 个数字。以 wav 格式保存。录制的数字串长从 1 个数字到 6 个数字，平均串长 3.125 个数字，每人读 5 遍。人员来自全国各地，以普通话朗读，但带有一定的地方口音，年龄大多在 20~24 岁，最大的 49 岁。数字串中，数字和数字连音出现的次数基本相同。在录音过程中，包含自然的开门关门声音、喘气声、咂嘴声等，这些可通过有效的端点检测算法滤除掉。对发音速度不过分限制，最快的每秒钟平均 4.7 个数字，最慢的每秒钟只有 2.92 个数字。

8.2 实验方法

用 4 人（2 男 2 女）前三次的单数字（数字串长为 1）和第一次的双数字（串长为 2）作为训练集，所有语音为识别集。在训练时用人工判别语音的起止点。训练采用最大距离分裂与 K-mean 算法。

实验平台为 VC++6.0。

8.3 实验结果

表 1 识别结果

	插入错误率	删除错误率	替换错误率	数字识别率	串识别率
一级识别	0.60	1.08	7.20	91.12	80.62
二级加权矩阵算法	0.60	1.08	4.24	94.08	83.56

表 2 识别时间比较

采用方法	识别所用时间（秒）	平均每数字时间（秒/数字）	预分割正确率
无预分割	8951.00	0.716	
有预分割	3152.00	0.252	100%

表 3　替换错误分布表

误识情况	一级识别误识次数（占替换错误的百分比）	二级识别误识次数（占替换错误的百分比）
2 误识为 8	82（41.67%）	49（46.23%）
8 误识为 2	54（26.11%）	31（29.25%）
0 误识为 6	34（13.89%）	22（20.75%）
7 误识为 1	7（6.67%）	3（2.83%）
9 误识为 6	3（1.67%）	1（0.94%）
总计	180（100%）	106（100%）

8.4　结论

从实验结果中可以看到：采用预分割法，计算时间减少到原来的 35.2%，说明该方法是有效的。没有达到理论上 25%以下，主要是因为预分割是以 100%分割正确为原则的，在语速很快时，分割难度加大，为保证分割正确分割点就减少了。在观察通话者同数字串的不同语速的语音的识别时间时，就发现识别时间随语速的加快而增加。

采用二级识别法错误减少了 41%，效果非常显著，这主要是加权矩阵算法强调了各维特征参数内在的对不同音素的不同的表达能力。系统的实验只是初步的结果，系统性能的提高有待进一步深入研究。（收稿日期：2003 年 4 月）

参考文献

1.Y Normandin et al.High-performance connected digit recognition using maximum mutual information estimation[J].IEEE Trans Speech and Audio Processing,1994,2(2):299~311

2.李虎生，刘加，刘润生.高性能汉语数码串语音识别[J].电子学报，2001，(6)

3.顾良，刘润生.汉语数码语音识别：发展现状、难点分析与方法比较[J].电路与系统学报，1997，(4)

4.顾良，刘润生.利用声调判别提高汉语数码语音识别性能[J].清华大学学报（自然科学版），1998，(9)

5.王成友，汤叔祺，梁甸农.一个基于 LB/IHMM 的高性能汉语连接数字语音识别系统[J].国防科技大学学报（自然科学版），1998，(3)

6.关存太，陈永彬，吴伯修.HMM 语音识别模型与一种修正训练算法[J].东南大学学报，1994，(1)

7.许海天，吴及，王作英.汉语连续数字串语音识别系统[J].计算机工程与应用，2002，38(1)：97~98

8.顾良，刘润生.改进汉语数码语音识别中的语音特征提取性能[J].电路与系统学报，1997，(4)

9.徐文盛，戴蓓倩，方绍武等.特定人汉语数码语音抗噪识别方法[J].电路与系统学报，2000，(1)

（上接 82 页）

Connectors：gcn；

Relations：SQL（gcn，转储数据库），SQL[（Warn，NLProcess），gcn]，Winsock[（服务程序，Warn），Wsock]，DCOM（DCOM 代理，服务程序）；}

3.2.3　服务程序子部件

Component DCOM 召唤部件：ActiveX 控件{

Description：进行具体的召唤工作；

Interface：man_summon；

Contract：DCOM 方法调用、Winsock 方法调用；}

Component Reckon {

Description：进行计算和累计工作；}

Component Wsock：Winsock {

Description：发送召唤反馈信息和报警信息；

Interface：sendData；}

Connector gcn：数据库连接 {

Description：转储数据库和应用客户端之间的连接；

Interface：Open；

Contract：SQL 语句；}

Connector Ccn：数据库连接 {

Description：原始数据库和应用客户端之间的连接；

Interface：Open；

Contract：SQL 语句；}

System 服务程序 {

Description：进行转储计算和重算工作；

Components：DCOM 召唤控件，reckon，Wsock；

Connectors：gcn，Ccn；

Realations：SQL（gcn，转储数据库），SQL（Ccn，原始数据库），Winsock[（reckon，DCOM 召唤控件），Wsock]，SQL[reckon，（gcn，Ccn）]；}

3.2.4　系统的设计约束和实现

以上体系结构设计对后期的具体设计和代码生成具有全局的指导和约束作用：原始数据库和转储数据库分别是一个 oracle 数据库；其余的 5 个部件则分别为单独的可执行应用程序。体系结构给出的是高层组成，在每个部件中可以给出下一层的结构描述，例如：各个应用中的数据库连接可以采用 ODBC 数据源或 oracle 数据连接引擎；除数据库以外的 5 个部件都放置一个 Winsock 控件，利用 sendData 接口方法进行信息交互；应用的手工召唤功能是通过调用功能服务的 DCOM 组件的 man_summon 接口方法完成的，它有召唤的厂站名、时间范围两个参数；监控部分都含有以曲线和表格形式显示数据库数据的部件；对异常数据的报警和处理是一个 ActiveX 控件，分别出现在三个监控部件中。

这样一来，整个系统的最上层结构以及各个部件的主要构成已经得到全面完整的表达。据此，在代码生成的支持下可以获得整个系统各个部件的工程文件和代码结构。

4　结束语

该文提出的体系结构描述方法和符号体系，已经得到实现，建立了一个基于体系结构描述支持的软件工程开发的支持工具。在此基础上，目前各类部件的建模和生成工具正在设计和构造过程中。（收稿日期：2002 年 7 月）

参考文献

1.Mary Shaw，David Garlan.Software Architecture[M].北京：清华大学出版社，1998：183~212

2.周莹新，艾波.软件体系结构建模研究[J].软件学报，1998，9(1)

3.冯铁，张家晨，陈伟等.基于框架和角色模型的软件体系结构归约[J].软件学报，2000，11(8)：1078~1086

4.骆华俊，唐稚松，郑建丹.可视化体系结构描述语言 XYZ/ADL[J].软件学报，2000，11(8)：1024~1029

5.杨卫东，于卫，蔡希尧.软件体系结构的描述方法研究[J].计算机研究与发展，2000，37(10)

6.耿刚勇，李渊明，仲萃豪.基于部件的应用软件系统的体系结构及其开发模型[J].计算机研究与发展，1998，35(1)

7.邓勇，丁峰，沈钧毅.基于 UML 的软件体系结构建模方法的研究[J].小型微型计算机系统，2001，22(10)

8.http：//www-2.cs.cmu.edu/~acme/adltk/adls.html

9.http：//www-2.cs.cmu.edu/~acme

10.http：//www.isr.uci.edu/projects/xarchuci