

用图样匹配法在计算机上自动识别语音*

俞 铁 城

(中国科学院物理研究所)

提 要

本文中提出了一个机器识别语言的模型。根据这个模型,提出一种将图样在时间域上规正的新方法。最后给出一组实验结果,正确识别率超过99%。这表明,该方案能应用于某些自动控制系统中。

一、引 言

长期以来,研究语言声学的人们对于自动识别语音做了很多工作^[1-15]。由于发音的长短给识别比较带来很大的困难,过去曾有不少人采用多种办法来解决这个问题,如切尾^[3],补零^[3],线性伸缩^[3,10],时间轴对数压缩^[8]等等。这些办法有明显的缺陷,如线性伸缩办法,显然跟实际的语音发声过程不符,一般地一次发音的长短主要由稳态的元音部分持续期不同而引起的,象爆发辅音(p,t,k,b,d,g)固有的音程短,采用线性加插法显然是不恰当的。近年来,采用动态编程的办法^[12,16],可将持续期不同的两列参数进行比较,是个比较成功的办法,但是耗时特别长,要做硬件实时处理比较困难。本文提出一个简单而有效的规正时间域的新方法。

二、识 别 模 型

一次发音中,元音的稳态部分,就其时间-频率-幅度三维频谱(或其它参数)来看,不发生显著变化,提供给我们的信息量除标志其何类元音外,只是持续期的长短。这在将频谱按音素类别分类中明显可以看到^[6],这对于机器识别语言来说,它有很多的多余信息。而在一个音向另一个音过渡阶段,三维频谱(或其它参数)发生比较激烈的变化,这对于机器识别语言来说,包含有极大的信息量。为此,我们设想一个识别模型,它能充分强调过渡音的重要性,而能压缩掉元音的稳态部分。

我们的识别模型中,利用三维频谱作为识别参数。而对每一次发音均选择出一定数目的频谱,这就是在时间域上规正频谱数目。详细办法见第三节。我们用相邻两时刻频谱值之差的绝对值(称为频谱的声刺激量),来衡量语音变动的激烈程度。对过渡音,这个量大,选取该时刻频谱的权重就大;反之,对于稳态部分,这个量小,选中的权重就小。

* 1976年2月24日收到;1976年8月10日收到修改稿。

为了在幅值方面进行规正,我们对每个选中频谱进行归一化处理.由于我们在选频谱上给了权重,所以让每一个选中频谱具有同样的权重参与图样匹配,而不是整个数据归一^[3],后者将对小能量的谱给予较小的权重参与图样匹配运算.

我们的识别模型不考虑换人.换人问题这样来解决:让新的发音人发一次样板音.

我们的识别方案是通用性的,即词表大小、具体的呼音项目没有限制,可以随意更换.

三、时间域规正办法

设在时刻 T_i 的语言频谱为

$$BP(T_i) = \{A_{i1}, A_{i2}, \dots, A_{ij}, \dots, A_{iL}\}, \quad (1)$$

其中 $A_{ij}(j=1, 2, \dots, L)$ 是 T_i 时刻第 i 通道里的能量,频带通道数目为 L .

假定采样周期为 T (秒),则

$$T_{i+1} - T_i = T. \quad (2)$$

考虑有 N 个频谱的一段语音,则 $i=1, 2, \dots, N$. 得到一个数字频谱图:

$$\{BP\} = \begin{bmatrix} A_{11} & A_{12} & \dots & A_{1L} \\ A_{21} & A_{22} & \dots & A_{2L} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \dots & A_{NL} \end{bmatrix}, \quad (3)$$

其中, A 的左脚标指时间序号,右脚标指通道序号. 除

$$\delta_1 = \sum_{j=1}^L A_{1j}, \quad n=1 \text{ 外}, \quad (4)$$

定义相邻两个时刻的频谱产生的声刺激为

$$\delta_n = \sum_{j=1}^L |A_{nj} - A_{n-1j}|, \quad n=2, 3, 4, \dots, N. \quad (5)$$

一段语音的总声刺激便是

$$\Delta = \sum_{n=1}^N \delta_n. \quad (6)$$

如果对一段语音作 M 次判断,即对 M 个频谱进行判别,则可能找出平均声刺激

$$\Delta_0 = \Delta/M, \quad M \leq N. \quad (7)$$

初步实验结果,选 $M \approx (2-3)$ 音节数就够了.

寻找这 M 个选中频谱的过程详细描述如下:

1) 清声刺激累加器 $R = 0$.

2) 依次取下一个 δ_i , 加进累加器 R ,

$$R \leftarrow R + \delta_i. \quad (8)$$

3) 若 $R \geq \Delta_0$, 则选中这个第 i 谱, 转往 5).

4) 否则, 不选中这个第 i 谱, 声刺激累加器只是增加了 δ_i , 转往 2).

5) 记住这个第 i 谱, 并给它编上选中谱的序号 p ; 同时, 声刺激累加器减少 Δ_0 ,

$$R \leftarrow R - \Delta_0. \quad (9)$$

6) 之后,查一下选中了 M 个谱没有,若还不够,转往 2)继续找,否则即告结束。

四、图 样 匹 配 法

对于用上面的步骤选中的 M 个频谱,写成

$$\{SBP\} = \{\bar{A}_{pi}\} \quad \begin{matrix} p = 1, 2, \dots, M; \\ i = 1, 2, \dots, L. \end{matrix} \quad (10)$$

对其中的每一个选中频谱都进行归一化处理,以达到克服发音强度变化的目的:

$$a_{pi} = \bar{A}_{pi} / \left(\sum_{j=1}^L \bar{A}_{pi}^2 \right)^{1/2} \quad \begin{matrix} i = 1, 2, \dots, L; \\ p = 1, 2, \dots, M. \end{matrix} \quad (11)$$

在计算机上工作时,先发一遍字表中的每一个项目,都按上述办法处理,并记住它们的 $a_{pi}^{(k)}$ ($p = 1, 2, \dots, M; i = 1, 2, \dots, L$), 作为样板音数据;其中上方指标(k)指第 k 个样板音。发样板音需按既定顺序,以便计算机认定。

然后,就可发待识别的音。用同样的办法得到待识别音的经正规化的数据:

$$a_{pi}^* = \bar{A}_{pi}^* / \left(\sum_{j=1}^L \bar{A}_{pi}^{*2} \right)^{1/2} \quad p = 1, 2, \dots, M, \quad (12)$$

其中($*$)指待识别音。用匹配法找最接近的音为识别结果。第 k 个样板音的得分为

$$DF(k) = \sum_{p=1}^M \sum_{i=1}^L a_{pi}^* a_{pi}^{(k)} \quad k = 1, 2, \dots, B, \quad (13)$$

其中 B 是字表中成员总数,在这 B 个得分中挑出最大者,例如 $DF(K)$ 为最大,则识别结果即是第 K 个项目。

五、实 验 步 骤

1. 设备

实验是在实时频谱分析系统(B/K3347)上进行的。该系统包括:

频率分析仪 2130, 控制和显示仪 4710, 计算机 7504 (Varian 620), 电传打字机 6401 (ASR33 型), 纸带穿孔机 6301, 纸带阅读机 7102。这里, 频谱分析仪 2130 是有 33 个通道的 1/3 倍频程滤波器组(中心频率从 25 赫到 20 千赫共 30 个通道, 外加参考值通道, 计权网络(A, B, C, D 任选一个)通道及全通道共 33 个通道)。检波时间常数 20 毫秒。计算机是 Varian 620, 字长 16 位, 存取周期 1.8 微秒, 内存 8K 字。

另外, 话筒用的是国产 CH11 型电容式话筒。

2. 程序

编了一个程序, 其粗框图见图1。该程序用符号语言写成, 以节省内存及加快程序的执行速度。说明几点:

1) 判定一个语音的开始与结束的办法是, 由全通或接近全通的 C 网络通道的能量超

过某个阈值(例如 20 分贝),判定一个音的开始;由该通道的能量连续低于某个阈值(例如 25 分贝),达一定的时间(例如 100 毫秒),则判定一个音的结束。

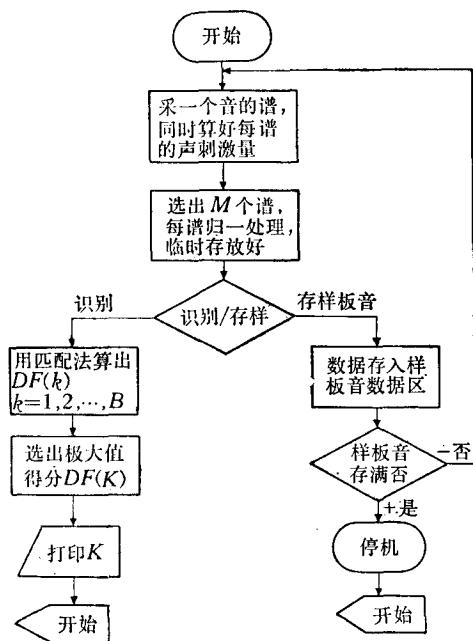


图 1 识别程序的粗框图

声相当高(69 分贝(A)). 发音时,话筒放在嘴唇侧前方 2—5 厘米处。

2) 为了克服像关门声及咳嗽声等短脉冲噪声的干扰,要持续一定的时间,例如 200 毫秒以上的音才认为是语音。

3) 原始频谱数据为分贝值,程序可直接对分贝值运行,也能将它转化为线性值运行。

4) 程序的主要运算部分用浮点运算.因此比较费时,对于字表为 10 个音,通道数目选用 8,样谱数选用 5 的情况,程序运行约 100 毫秒;对于发音长度为 300 毫秒的单音节字,则 1 秒钟能识别约 2 个音。

下面的实验结果,全部为对分贝值数据进行运算的。

3. 实验环境

全部实验都是在计算机房即席呼叫进行的。该机房是普通办公室隔出来的,未加声学处理。机房的空气调节器打开着,本底噪声相当高(69 分贝(A)).

六、实验结果

下面的实验结果中,SPS 代表每个样板音选用频谱数目;TDS 代表选用的滤波通道数目。对于不同的 TDS 值所相应的选用滤波通道情况见表 1。

表 1 TDS 与滤波通道选用对照表

TDS 值	中 心 频 率 (赫兹)														
	160	200	250	315	400	500	630	800	1K	1.2K	1.6K	2K	2.5K	3.1K	4K
6	×			×			×			×			×		×
8	×		×		×		×		×		×		×		×
10				×	×	×	×	×	×	×	×	×	×		
12			×	×	×	×	×	×	×	×	×	×	×	×	
14		×	×	×	×	×	×	×	×	×	×	×	×	×	×
16	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×

说明: 上表中,中心频率项下有“×”者,表示这个通道被选用了。注意,当 TDS = 10, 12 时,相当于一般的电话频段

实验 1

发音人 Y; 材料 0, 1, 2, ..., 9. 呼 120 遍,其中“9”只呼 119 遍;错误 4/1199;

SPS = 3; TDS = 10 (相当于电话频段).

		识别结果									
		0	1	2	3	4	5	6	7	8	9
口呼输入	0	118									2
	1		120								
	2			120							
	3				120						
	4					120					
	5						120				
	6		1					119			
	7	1							119		
	8									120	
	9										119

实验 2

发音人 Y; 材料 0, 1, 2, ..., 9. 呼 30 遍; 错误 1/300; SPS = 3; TDS = 8.
错误情况: 0 错成 4, 1 次.

实验 3

发音人 R (女); 材料 0, 1, 2, ..., 9. 两次各呼 10 遍; 错误 3/100 及 1/100; SPS = 6; TDS = 8.

错误情况 2 错成 3, 1 次; 3 错成 9, 1 次; 7 错成 3, 2 次.

实验 4

发音人 Y; 材料 十个人的名字 (均为三音节), 发 60 遍; 错误 1/600; SPS = 6; TDS = 8.

错误情况: 5 错成 8, 1 次.

实验 5

发音人 Y; 材料 四字成语二十个. 呼 55 遍, 其中三个是关门噪声 (不计), <13> 多呼一次, <19> 少呼一次; 错误 26/1097; SPS = 10; TDS = 8.

错误情况 1 错成 2, 1 次; 13 错成 4, 2 次; 14 错成 5, 1 次; 17 错成 11, 1 次; 17 错成 13, 21 次.

实验 6

发音人 Y; 材料 同实验 5, 用乡音 (吴语) 呼 20 遍; 错误 0/400; SPS = 10; TDS = 8.

错误情况 全部正确.

实验 7

发音人 Y; 材料 二十句话 (每句七字). 呼 20 遍; 错误 3/400; SPS = 14; TDS = 6.

错误情况 18 错成 16, 1 次; 19 错成 16, 1 次; 19 错成 17, 1 次.

实验 8

发音人 Y; 材料 0, 1, 2, ..., 9; SPS = 10; TDS = 16.

为了检查发音的持续时间可在多大范围内变动而专门对口呼数字进行了一次实验.

下表中列出的是实验结果。这些发音均识别正确。表中列出的数字是毫秒数。进行本实验时,受当时程序设计限制,音的最大持续时间为 680 毫秒。

		发音持续时间变化范围									
		材料									
		0	1	2	3	4	5	6	7	8	9
长 音		290	380	480	450	550	590	470	520	470	470
			500						500		
			510						670		
			480						680		
			540								
			650								
短 音		160	250	270	180	230	270	220	230	200	200
			210						260		
			330						190		
			210								

七、讨 论

1. 呼一遍样板音,有时会引起个别样板音的正确识别率不高,这时宜重换一下这个样板音。或者样板音本身就要求呼几遍以取得平均效果,从而改善识别结果。实验 5 中,26 个错误,有 22 个错误出现在“一团和气(17)”上。

2. 一个人的发音,家乡音显然比较稳定。(实验 5 与实验 6 相比较)。因此,实用时,宜用乡音。

3. 实验 3 是由一位未在本系统环境中发过音的人呼的结果。第一遍错 3/100,第二遍错 1/100。可见,“陌生人”很易掌握使用。

4. 对本程序系统作一些修改,例如,训练一遍改成几遍,字表中加进一个纠错指令,前一个识别结果若不对,则呼纠错指令,重呼该音,作了一些这样的修改后,可付诸实用。

5. 我们提出的识别方案中,将样板音参数在时间域上规正化的办法,可靠地解决了发音长短的问题。而且,从实验结果看,这个办法不仅适用于单音节字,也完全适用于多个音节的词组或语句。除为了说明问题起见而列出的实验 3 和实验 5 外,正确识别率均在 99% 以上。

6. 换人(或换字表)只需新的用户(或老用户)重呼一遍样板音(或呼一遍新样板音),之后便可开始识别,训练机器,极为方便。

7. 由于计算机内存的限制(8K 字)及为了摸索规律而占用了较多的内存,字表只限于约 20 个单位。下一步正在作的工作是,对程序作修改,以期能扩大字表并加快识别速度达到实时处理;进一步简化方案,摸索制备硬件的可靠而方便的方案。

参 考 文 献

- [1] D. R. Reddy, *Proc. of the IEEE*, **64**(4) (1976), 501.
- [2] K. H. Divis, R. Biddulph and S. Balashek, *JASA*, **24**(6) (1952), 637.
- [3] P. Denes and M. V. Mathews, *JASA*, **32** (11) (1960), 1450.
- [4] H. Kusch, *NTZ*, **18** (1965), 57.
- [5] L. Gilli and A. R. Meo, *Acustica*, **19** (1) (1966), 38.

-
- [6] W. Bezdel and J. S. Bridle, *PIEE*, **116** (4) (1969), 617.
 - [7] K. Hiramutsa and K. Kotoh, Reports of the 6-th ICA, II Speech, (1968), p. B-119.
 - [8] S. Washizawa *et al.*, Reports of the 6-th ICA, (1968), p. B-4-7.
 - [9] S. Inoue, A. Kurematsu, Reports of the 6-th ICA, (1968), p. B-4-5.
 - [10] L. C. W. Pols, *IEEE Trans. on Computer*, **C-20**(9) (1971), 972.
 - [11] F. ItaRura, *IEEE Trans. on ASSP*, **ASSP-23** (1) (1975), 67.
 - [12] M. R. Sambur and L. R. Rabiner, *BSTJ*, **54** (1) (1975), 81.
 - [13] J. L. Flanagan, "Speech Analysis Synthesis and Perception" (1965), New York.
 - [14] *IEEE Trans. on Audio and Electroacoustics*, **AU-21** (3) (1973).
 - [15] B. S. Atal and L. R. Rabiner, *IEEE Trans. on ASSP*, **ASSP-24** (3) (1976).
 - [16] H. Sakoe, S. Chiba, The Proc. of the 7-th ICA, **3** (1971), 65.

ON-LINE AUTOMATIC RECOGNITION OF SPOKEN SOUNDS BY A PATTERN MATCHING METHOD

YU TIE-CHENG

(Institute of Physics, Academia Sinica)

ABSTRACT

A model for speech recognition by computer is suggested. In which a new method of pattern normalization in the time domain is presented. A program for recognizing spoken sounds has been developed on a Varian 620/L computer. Finally, a set of experimental results is given, the correct recognition rate being greater than 99%. It is shown that this new method can be used in certain automatic control systems.