基于禁止搜索的非线性时间匹配优化算法 [1]

梅晓丹 孙圣和

(哈尔滨工业大学自动化测试与控制系 哈尔滨 150001)

摘 要 动态时间规整算法 DTW(Dynamic Time Warping) 作为一种非线性时间匹配技术已成功地应用于语音识别系统中。DTW 算法使用动态规划技术来搜索两个时间序列的最优规整路径，虽然这种算法计算量小，运算时间较短，但只是一种局部优化算法。禁止搜索 TS(Tabu Search) 算法是一种具有短期记忆的广义启发式全局搜索技术，适用于解决许多非线性优化问题。本文将该技术用于语音识别系统中，提出了基于禁止搜索的非线性时间规整的优化算法 TSTW，使得时间规整函数尽可能逼近全局最优。仿真结果表明，TSTW 比 DTW 有更高的识别率，且运行时间比遗传时间规整算法 GTW 大大减少。

关键词 禁止搜索，语音识别，动态时间规整，非线性时间匹配

中图号 TN912.3

1 引 言

在语音识别系统中，小词汇表及单字的识别通常采用模板匹配法。它将输入语音分为一系列语音帧，并且每一帧都提取出一个表征语音信息的特征矢量 (即输入语音模式)。在训练阶段，把特征提取后的语音模式作为参考模式存入模板库中。在识别阶段，待识别的语音也要经特征提取形成测试模式，然后把该模式与模板库中的参考模式相比较，即计算它们之间的谱失真，最后根据识别规则选出与测试模式最接近的参考模式作为识别结果输出。

由于不同人说话速度有很大差异，并且同一个人在不同时间说话速度也不尽相同，从而导致非线性波动和时间轴不匹配，直接影响语音识别系统中模式匹配的程度，成为语音识别的主要问题。日本学者 F. Itakura 把动态规划的概念用于解决孤立词识别时说话速度不均匀的难题，提出了 DTW 算法 [1]，该算法简单且训练和搜索时间较快，取得了较好的效果。然而该算法是一种局部最优算法，每一步搜索都是根据局部优化的判断进行的，因此整个时间规整路径达不到全局最优，且由于局部搜索是递归进行的，对全局路径的标准化及规整子路径的加权都无法全局衡量，使得识别结果显得粗糙，从而影响了其性能 [2−4]。文献 [2] 中提出了遗传时间规整方案 GTW，取得了很好的识别效果。

禁止搜索算法 [5] 是带有短期记忆的全局优化技术。其基本思想是通过一系列移动来搜寻可行解的搜索空间并且禁止目前迭代的某些搜索方向以避免死循环而跳离局部极小。这些移动部分地或完全地记录在禁止表中，禁止以后迭代中的重复操作。它使用多个候选解以引导搜索向最优目标函数值的方向进行。同时，算法作次优的移动允许从先前的局部最优点出发继续搜索。禁止搜索算法还使用禁止表来避免重复搜索，即在搜索过程中记忆已搜索过的解的关键特征，并在以后的搜索中根据禁止表中所记忆的特征来阻止搜索过程回到以前已经搜索过的解，从而使得搜索过程经济有效地进行。

为了提高动态时间规整算法的性能，本文提出用禁止搜索的思想来尽可能找到全局优化的时间规整路径。为了分析方便并考虑语音识别系统的特点，本文主要讨论了该算法在语音单字识别系统中的性能和结论。

2 DTW 的基本原理

在语音识别系统的相似度比较阶段，为了使两个模式各帧的时间轴能够匹配必须进行时间规整，动态时间规整算法是解决该问题的重要工具。DTW 算法的示意图如图 1 所示，参考模式 $\boldsymbol{R}$ 的各帧标注在纵轴上，测试模式 $\boldsymbol{T}$ 的各帧则标注在横轴上。

[1] 2000-03-06 收到，2000-07-28 定稿

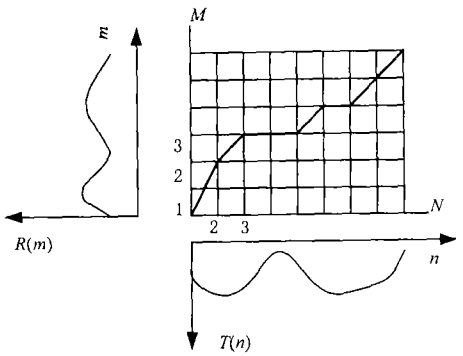

图 1 DTW 算法示意图

图中 m 为参考语音帧的时序标号，M 为该模式包含的总帧数。n 为测试语音帧的时序标号，N 为该模式包含的总帧数。它们之间的相似度可以用短时谱失真度 $D(\boldsymbol{T},\boldsymbol{R})$ 来表征。DTW 算法的目的就是求一条从 (n_1,m_1) 到 (n_N,m_N) 的路径，使短时谱失真度 $D(\boldsymbol{T},\boldsymbol{R})$ 最小。设点 (n_i,m_i) 的失真为 $d(n_i,m_i)$，从 (n_1,m_1) 开始到 (n_N,m_N) 的短时谱失真度 $D(\boldsymbol{T},\boldsymbol{R})$ 为

$$D(\boldsymbol{T},\boldsymbol{R})=\sum_{i=1}^{N}d(m_i,n_i) \tag{1}$$

一般时间规整路径可用如下规整函数表示

$$m_i=W(n_i),\quad i=1,\cdots,N \tag{2}$$

两个模式的匹配程度由所有匹配帧的加权短时谱失真度来描述。它可定义为

$$D_W(\boldsymbol{T},\boldsymbol{R})\sum_{i=1}^{N}\frac{d(m_i,n_i)\cdot a}{A} \tag{3}$$

其中 a 为子路径加权系数，A 为归一化系数。

此外，规整过程要注意以下几点：(1) 起始点与结束点由于噪声很容易波动而不精确，所以路径的始点与终点可以有一松弛以免影响匹配的准确性，即开始点可以为 $(1,2)$, $(1,3)$；结束点可以为 $(N,M-1)$, $(N,M-2)$。(2) 规整函数要保证单调性和局部连续。(3) 如果两个模式的长度相差很大，那么规整路径将过陡或过缓，从而使时间规整没意义。因此规整路径的斜率必须有所限制。这里的斜率是指子路径的始点和终点在纵轴与横轴上的步进长度之比。H. Sakoe 和 S. Chiba 在文献 [6] 中提出了 4 种斜率限制方案，如图 2 中所示，所有图的右上角 e 为各个子路径的终点。图 2(a) 中 3 种子路径斜率分别为 0，1，$+\infty$；图 2(b) 中五种子路径斜率分别为 1/3，1/2，1，2，3；图 2(c), 2(d) 依次类推。匹配时必须根据两个模式的长度差异选择合适的斜率限制方案，防止规整路径的斜率过陡或过缓。若两个模式的时间长度相差不大，图 2(c) 和 2(d) 的方案较合适；若相差较大，则图 2(a) 和 2(b) 的方案较合适。(4) 规整区域的定义目的也是为防止不适当的时间匹配，通常规整区域由以下表达式确定

$$Q_{\min}\le\frac{m_i-1}{n_i-1}\le Q_{\max},\quad Q_{\min}\le\frac{M-m_i}{N-n_i}\le Q_{\max} \tag{4}$$

其中 $Q_{\min}$ 和 $Q_{\max}$ 分别为规整函数的最大斜率和最小斜率。

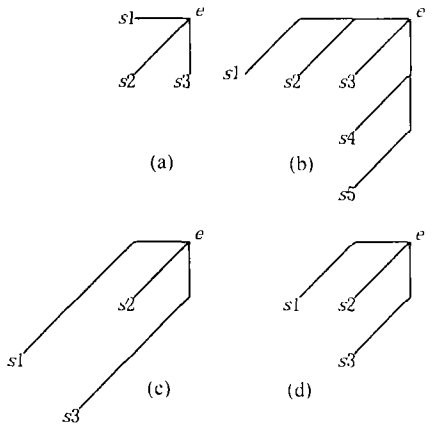

图 2　四种斜率限制方案

3 TSTW 基本原理

禁止搜索算法的基本思想是通过一系列移动来搜寻可行解的空间，禁止某些搜索方向以避免死循环 (避免重复搜索路径) 和跳离局部极小 (全局优化) 。这些移动的特征部分地或完全地记录在禁止表中，以禁止后面的迭代重复前面进行过的搜索动作。基本的禁止搜索算法可以大致描述如下：

```
禁止搜索算法
{
    随机产生初始解并给出当前解和最优解;
    如果迭代次数没达到
    {
        在当前解的邻域内产生测试解;
        求出各测试解相应的目标函数值;
        按目标函数值及禁止表来更新和修改当前解、最优解和禁止表以求逐渐向最优解逼近;
    }
    最终的最优解即为所求;
}
```

禁止搜索算法的关键在于如何描述一个解以及如何产生当前解的邻域解。下面我们将介绍 TSTW 的解描述方案、参数选择问题和邻域解的生成。

本文采用的规整方案为图 2(b) 的斜率限制方案，它共有 5 种子路径，可以用集合 A 来描述，即 $A=\{s1,s2,s3,s4,s5\}$ ，那么整个规整路径 $\boldsymbol{P}$ 可以看作由一串子路径构成，即 $\boldsymbol{P}$ 为 $p_1 \rightarrow p_2 \rightarrow \cdots \rightarrow p_i \rightarrow \cdots \rightarrow p_k$ ，其中 $p_i \in A$, k 为路径 $\boldsymbol{P}$ 所包含的子路径数目。例如图 3 的规整路径由 5 条子路径连接而成，即 $k=5$ ，它可以描述为 $\boldsymbol{P}$ 为 $p_1 \rightarrow p_2 \rightarrow p_3 \rightarrow p_4 \rightarrow p_5$ ，那么 $p_1=s1$, $p_2=s3$, $p_3=s4$, $p_4=s2$, $p_5=s3$ 。图中的 $1,2,3,4,5$ 分别是各子路径的终点。

TSTW 算法的一个解就是一条路径。其最优解就是一条使测试模式和参考模式具有最小失真度的路径 $\hat{\boldsymbol{P}}$，其对应的时间规整函数 $\hat{W}$ 满足

$$\boldsymbol{D}_{\hat{W}}(\boldsymbol{T},\boldsymbol{R})=\min_{W}\boldsymbol{D}_{W}(\boldsymbol{T},\boldsymbol{R}) \tag{5}$$

禁止搜索算法的 4 个主要参数为：解改动概率阈值 P_m ，迭代次数 I_m ，每次迭代测试解个数 N_S 和禁止表长度 T_S 。解改动概率阈值 P_m 定义了由当前解生成邻域解所作的改动量占当

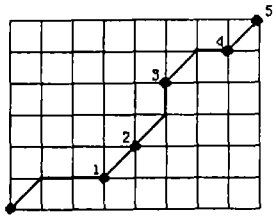

图 3 一个规整路径的例子

前解的百分比。它取决于参考模式和测试模式的帧数和特征矢量的维数。这些参数值越大，P_m 应选择得越小，本文取 $P_m = 4\%$ 。迭代次数 I_m 和每次迭代测试解个数 N_S 共同确定了禁止搜索算法的所能搜索的解个数为 $I_m \cdot N_S$ 。本文选取 $I_m = 200, N_S = 20$ 。通常，禁止表的长度取决于每次迭代的测试解个数，所以本文选取 $T_S = 20$ 。禁止搜索算法最关键的问题是如何产生邻域解。较简单的方法就是对当前解作随机变动。本文采取的方法是任意选取两个子路径进行交换而构成邻域解。

基于上述思想，本文的 TSTW 算法可以描述如下：

TSTW 算法

{

随机产生初始路径 INIT ；

令当前路径为 CURR=INIT ；

令最优路径 BEST=INIT ；

如果搜索迭代次数没达到 I_m

{

* 在 CURR 路径的邻域 (允许区) 内产生 N_S 个邻域路径;

求出各邻域路径相应的目标函数值;

按目标函数值从优到差排列这 N_S 个邻域路径;

对 N_S 个邻域路径中从优到差判断是否为禁止路径或者虽是禁止路径但是其目标函数值比最优路径的目标函数值还好，则把它作为新的 CURR 路径；否则，继续测试下一个测试解;

如所有邻域路径都是禁止的，则转到步骤 * ；

如禁止表溢出了，把表中的最旧值去掉;

把新的 CURR 路径插入禁止表;

}

最终的 BEST 路径即为所求;

}

与文献 [2] 中的遗传时间匹配算法 GTW 相比，这个算法在搜索过程中加入了记忆，即将搜索过的路径特征作为禁止标志存储，避免后面重复搜索，从而大大减少了搜索优化路径的时间，且由于是全局优化，识别率也较 DTW 算法高。

4 仿真实验

为了验证算法的有效性，本文对 DTW 算法， GTW 算法和 TSTW 算法都进行了仿真实验并对识别率及运行时间加以对比。数据库采用 4 个人对 10 个汉字的发音记录，每人把每个字说 10 次，得到 400 个发音，提取其特征矢量，把其中的 70% 作为参考模式， 30% 作为测试模式。数据采样率为 8kHz ，每个采样占 8 位，每 160 个采样为一帧，特征矢量通过 10 阶倒谱分析获得，即参考模式和测试模式中特征矢量采用了倒谱。其短时谱失真度利用倒谱定义为

$$D_W(\boldsymbol{T}, \boldsymbol{R}) = \sum_{i=1}^{N} \frac{|h_{xi} - h_{yi}| \cdot a}{A} \tag{6}$$

实验中采用的倒谱加权窗函数为 $W_n(k)=1+(p/2)\sin(k\pi/p)$，即语音数据第 n 帧的倒谱用下式代替 $h_n(k)=W_n(k)h_n(k)$，其中 $k=1,2,\cdots,p$; p 是预测阶数，此处为 10。

实验所采用的规整方法如图 2(b) 所示，即规整过程的每一步有五种前进方向 $s1$, $s2$, $s3$, $s4$, $s5$，其子路径加权系数 a 对应分别为 5，3，1，3，5；规整的斜率规定最大值 $Q_{\max}$ 为 3，最小值 $Q_{\min}$ 为 1/3，规整区域为此斜率对应的区域。规整的起始点和结束点松弛误差的取值不超过 3。DTW 算法，GTW 算法和 TSTW 算法的实验结果如表 1 所示。

表 1　DTW，GTW 和 TSTW 的平均运行时间 (s)

I_m(次)	DTW(s)	GTW(s)	TSTW(s)	Δ(%)
–	0.037	–	–	–
20	–	0.205	0.103	50.2
40	–	0.366	0.211	57.7
80	–	0.698	0.464	66.5
100	–	0.861	0.547	63.5

表 1 给出了三种算法的平均运行时间并进行了比较，其中 I_m 表示 GTW 算法和 TSTW 算法的迭代次数，Δ 表示 TSTW 算法与 GTW 算法的平均运行时间比。DTW，GTW 和 TSTW 三种算法的识别率分别为 92%，96%，96%。可以看出，TSTW 算法的识别率比 DTW 高，与 GTW 算法接近，但平均运行时间只有 GTW 算法的一半，识别效率显著提高。

表 2 记录了对 10 个字的识别情况。其中 M_S 表示相同字识别的平均谱误差，即参考模式和测试模式为相同字时谱误差的平均值，M_d 表示不同字识别的平均谱误差，即参考模式和测试模式为不同字时谱误差的平均值。

表 2　DTW 和 TSTW 的平均谱误差对比

标号	DTW			TSTW		
	M_s	M_d	M_d-M_s	M_s	M_d	M_d-M_s
1	0.904	3.933	3.029	0.922	4.201	3.279
2	0.794	4.139	3.345	0.908	5.198	4.290
3	0.833	4.980	4.147	0.813	5.259	4.446
4	0.812	5.681	4.869	0.751	5.907	5.156
5	0.931	6.198	5.267	1.009	7.028	6.019
6	0.927	3.909	2.982	0.997	4.297	3.300
7	0.823	5.856	5.033	0.961	6.179	5.218
8	1.048	3.900	2.852	1.124	4.265	3.141
9	0.786	3.668	2.882	0.801	4.401	3.600
0	0.978	4.012	3.034	1.032	4.813	3.781

在比较两种算法时，我们采用 $|M_d-M_S|$ 来比较算法在易混淆字识别时的性能，$|M_d-M_S|$ 的大小表征了识别能力的强弱。由表 2 可以看出，TSTW 比 DTW 有更强的识别易混淆字的能力，识别效果更好。

5 结　论

本文提出了一种基于禁止搜索的动态时间规整算法 TSTW。通过全局搜索技术解决了非线性时间匹配问题，克服了 DTW 算法的局部最优和 GTW 算法运行时间长的缺点，文中建立了单字识别的模型并对两种算法进行了仿真测试。仿真结果表明 TSTW 算法比原始的 DTW 算法具有更高的识别率，同时表明在易混淆字的识别方面 TSTW 算法具有更强的识别力和更好的鲁棒性，且运行时间相对 GTW 算法大大减少。

参 考 文 献

[1] F. Itakura, Minimum prediction residual principle applied to speech recognition, IEEE Trans. on ASSP, 1975, ASSP-23(2), 67–72.

[2] S. K. Wong, C. W. Chau, W. A. Halang, Genetic algorithm for optimizing the nonlinear time alignment of automatic speech recognition system, IEEE Trans. on Industrial Electronic, 1996, IE-43(5), 559–566.

[3] H. F. Silverman, D. P. Morgan, The application of dynamic programming to connected speech recognition, IEEE ASSP Magazine, 1990, 38(7), 7–24.

[4] L. R. Rabiner, S. E. Levinson, Isolated and connected word recognition-theory and selected applications, IEEE Trans. on Communications, 1981, COM-29(5), 621–658.

[5] F. Glover, M. Laguna, Tabu Search, Kluwer Academic Publishers, London, 1997, I chapter.

[6] H. Sakoe, S. Chiba, Dynamic programming algorithm optimization for spoken word recognition, IEEE Trans. on ASSP, 1978, ASSP-26(2), 43–49.

AN ALGORITHM FOR OPTIMIZING THE NONLINEAR TIME ALIGNMENT BASED ON TABU SEARCH APPROACH

Mei Xiaodan Sun Shenghe

(*Dept.of Automatic Test and Control, Harbin Institute of Technology, Haerbin* 150001, China)

Abstract Dynamic Time Warping(DTW) has been widely used in speech recognition systems as a nonlinear time alignment technique. It uses the dynamic programming technique to search the optimal warping path for two time sequences. Although this algorithm needs less computation and shorter training and searching time, it is a local optimization algorithm. The Tabu Search(TS) algorithm is the generalized heuristic global search technique with short-time memory, and suitable for solving many nonlinear optimization problems. This paper applies this technique to speech recognition systems, and presents a new algorithm for optimizing time warping based on TS approach, which makes time warping functions optimized globally. Simulation results show that TSTW has better time warping performance than DTW and GTW.

Key words Tabu search, Speech recognition, Dynamic time warping, Nonlinear time alignment

梅晓丹： 女，1974 年生，博士生，研究方向为语音信号的识别与压缩.

孙圣和： 男，1937 年生，教授，从事计算机测试与控制，信号处理与系统辨识，数据压缩的研究.