

文章编号: 0529-6579(2000)01-0013-05

基于骨架的一种签名分段算法^{*}

伍小明, 陈懿新

(中山大学数学系, 广州 510275)

摘 要: 提出了一种基于骨架的签名分段算法. 依照这种算法对签名进行分段, 有效地实现了签名结构复杂度均衡的分段, 取得了较好的结果.

关键词: 自动签名识别; 签名分段; 方向码; 复杂度

中图分类号: O235 **文献标识码:** A

自 80 年代末以来, 就有学者^[1, 2] 对签名分段方法进行研究, 提出了一些签名分段算法. 他们大多采用的是像手写识别处理中的笔画划分的方法, 但通常人们的签名并不会具有明显的字符笔画特征, 因此, 基于笔画进行分段的方法具有很大的局限性. 本文基于对签名图像结构复杂性均衡分解的思想, 提出了一种基于骨架的签名分段算法. 采用这种方法首先对签名进行骨架提取, 然后依照结构复杂度均衡的原则, 对骨架进行分段, 最后把所得的骨架分段复原成原始签名中的分段.

1 骨架提取

骨架提取是自动签名识别中关键的预处理之一. 一般来说, 对签名的骨架提取有以下要求:

- 1) 骨架要保持原来签名的连续性;
- 2) 要提取成一像素的骨架, 骨架应接近原笔划的中心线, 最好做到 8-连接;
- 3) 不产生严重的畸变, 对噪音不敏感;
- 4) 保留签名原有的拓扑、几何特征, 特别是一些明显的拐角不应被去掉.

基于以上基本要求, 本文采用如下算法:

设 p 为签名二值图像中一点, 点 p 的 3×3 邻域中像素点按如下方法编号: 以点 p 为中心, 右邻域编号为 p_0 , 逆时针方向依次编号为 $p_1, p_2, \dots, p_7$. $Value(p)$ 代表点 p 的像素值 (1 为签名图像点, 0 为背景点), $Sum(p)$ 代表点 p 的 3×3 邻域中除 p 之外的其它 8 个点的像素值之和. 按以下算法确定点 p 的像素值 $Value(p)$. 整个骨架提取过程中, 点 p 在签名图像中从左到右、从上到下有序地进行, 直到没有可剔除的像素点为止.

^{*} 基金项目: 国家自然科学基金资助项目 (197710989)

收稿日期: 1999-03-16 作者简介: 伍小明 (1948 ~) 女, 副教授.

算法:

- 1) 若 $Value(p) = 0$, 则 $Value(p)$ 仍为 0;
- 2) 若 $Value(p) = 1$, 而 $Sum(p) = 0$, 则 $Value(p)$ 取为 0;
- 3) 若 $Value(p) = 1$, 而 $Sum(p) \geq 6$ 或 $Sum(p) = 1$, 则 $Value(p)$ 仍为 1;
- 4) 若 $Value(p) = 1$, 而 $Sum(p) \geq 2$ 且 $Sum(p) \leq 5$, 则按如下 2 步确定 $Value(p)$ 的值;

① 设 p_i 为 p 的 3×3 邻域中的一点, 标号 i 从 0~7 依顺序分别进行以下判断: 遇 i 为偶数则参考其前后相邻标号的各 2 个点的像素值是否全为 0, 若是, 则直接转至 5), 否则, 转至 ②; 遇 i 为奇数则参考其前后相邻标号的各 1 个点的像素值是否全为 0, 若是, 直接转至 5), 否则, 转至 ②;

② 若 $Sum(p) = 2$ 或 $Sum(p) = 3$, 则置 $Value(p) = 0$; 若 $Sum(p) = 4$ 或 $Sum(p) = 5$, 则再进行判断, 视标号为 p_0, p_2, p_4, p_6 的点的像素值是否全为 1, 若是, 则置 $Value(p)$ 为 1, 否则, 置 $Value(p) = 0$;

5) 结束.

2 骨架分段描述

在二值图像处理中, 经常用到的一种方向码(图 1a), 可以用它来描述图像的边沿骨架等图像特征, 在本文中把这种方向码称为方向码 A.

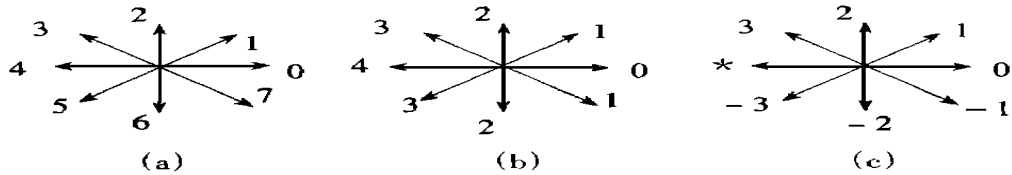


图 1 3 种不同的方向码

Fig.1 Three different direction codes

针对签名图像分段复杂性的描述和分段特征的提取的需要, 本文还采用了另外的 2 种方向码, 分别称为方向码 B(图 1b)、方向码 C(图 1c). 图 1c 中的 * 表示一未定值.

本文采用的分段算法为:

- 1) 从整个签名图像的骨架中提取出并记下所有的交叉点和端点;
- 2) 从上到下, 从左到右搜索第 1 个端点;
- 3) 从该端点开始沿骨架进行步进, 按方向码 B 计算前进中的复杂度. 若复杂度达到 1 个给定的阈值后(阈值的大小由实验要求的精确度确定, 一般情况下, 对于西文签名取 8~15 为佳, 中文签名取 15~20 为佳), 即把刚走过的全部点作为一个分段, 抹去刚走过的除交叉点之外的点, 直到抹去骨架中所有的点.

所涉及的规则:

规则 1 复杂度的定义与计算.

设 p_0, p_1, p_2, p_3 为签名图像骨架中的 4 个点, 所谓由 p_2 沿骨架步进到 p_3 相对于 p_0p_1 的复杂度是由向量 p_2p_3 与向量 p_0p_1 的夹角 θ 按方向码 B 计算所得.

默认情况下, 当前步的复杂度的计算由当前点与它的前 1 点构成的方向与当前点的前 2

点构成的方向的偏移量按方向码 B 计算所得. 例如, p_2p_3 与 p_1p_2 成 $+45^\circ$ 的偏角, 则 p_2 到 p_3 的复杂度为 1.

规则 2 交叉点的处理.

签名图像中必然会存在交叉点, 签名骨架中也应该保留这些交叉特性, 因此在分段过程中必须处理交叉点的问题, 本文采用以下方法.

① 按前进中的最小复杂度进行. 图 2a 中, 从 p_1 到 p 时, 按复杂度最小原则走向 p_2 , 这是符合真实的签名的场景的.

② 在实现时还可以采用一种记帐的方法, 就是连续取交叉点的各可能的方向中的几个点做检验, 分别找出其与交叉点在签名骨架中的反向对应点, 按方向码 B 计算复杂度, 比较后取复杂度最小的方向作为步进的方向. 例如图 2b 中 p 为交叉点, 当从 p_1 经 p_2 到 p 时, 因 p 为交叉点, 则开始记帐, 分别对 p_3, p_4, p_3', p_4' 和 p_3'', p_4'' 进行考虑, 这里只考虑 p_3, p_4 与 p_3', p_4' 作为例子. 按 p_3, p_4 计算, p_3p 相对于 p_2p 的复杂度为 1, p_4p 相对于 p_2p 的复杂度为 2, 总和为 3. 按 p_3', p_4' 计算, $p_3'p$ 相对于 p_2p 的复杂度为 3, $p_4'p$ 相对于 p_2p 的复杂度为 2, 总和为 5. 取较小者, 故按 p 到 p_3 进行.

规则 3 断笔处理.

在签名中经常会有断笔现象, 骨架提取之后应该保持这些断笔特性. 在按骨架进行分段的过程中, 断笔的处理是一个困难的问题, 本文采用了如下的方法, 取得了较好的效果.

在遇到断笔时, 即把刚走过的点作为一个分段, 不考虑其复杂度的大小. 然后, 由当前点和前一点可以形成一直线, 在这条直线前进方向上的两边范围内寻找新的端点, 作为下一个分段的开始.

3 对应段的图像复原

所谓对应段的复原, 即由上面分段算法所得的骨架分段, 恢复该骨架分段原签名图像中的对应分段部分.

段的复原困难在于交叉点的复原, 本文在交叉点的复原处理上采用如下特殊方法.

① 根据前面分段算法记下的骨架的交叉点, 在原始签名中构建每个交叉点处各笔段在该处的重叠区域.

图 3 中 p 是骨架图像中的 1 个交叉点, g_1, g_2, g_3, g_4 是由 p 出发的 4 个骨架分支, 虚线之内表示原始签名的笔画, 分别在 g_1 与 g_2, g_2 与 g_3, g_3 与 g_4, g_4 与 g_1 之间寻找离点 p 距离最小的背景点, 在图 3 中可以找到 p_1, p_2, p_3, p_4 , 按顺时针首尾连接这些点成一多边形, 多边形之内的点即是交叉点 p 处的重叠区域.

② 构建了所有交叉点的重叠区域以后, 就可以开始分段复原了, 步骤如下:

第 1 步判断当前段中是否有交叉点, 若有, 则记下该交叉点重叠区域, 设为 S ;

第 2 步考察该分段的 2 个端点 d_1, d_2 , 若 $d_i (i=1, 2)$ 不是骨架图像中的端点, 则

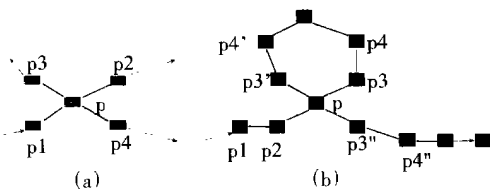


图 2 交叉点的处理

Fig. 2 Illustration for the manipulation of crosspoints

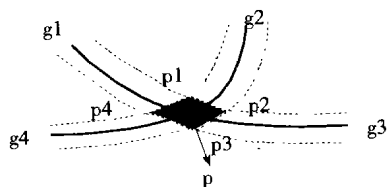


图 3 交叉点

Fig. 3 One case of crosspoint

把该分段中与 d_i 相邻的点记为 d_{ni} , 则由 d_i 与 d_{ni} 成一直线 L_d , 过 d_i 在 L_d 的垂直方向作一直线 L_{dv} , 在直线 L_{dv} 的 d_{ni} 所在的一边找出所有与 L_{dv} 邻接 (距离不大于 2) 且与 d_i 连通的点, 把这些点再加上 d_i 组成点集 Dot .

第 3 步对于当前操作段的每个点 p_i , 若 $p_i \in Dot$; 若 $p_i \in S$, 把 p_i 相邻的签名像素点加入到该分段中去, 否则, 把 p_i 相邻且属于 S 的签名像素点加入到该分段中去, 进行迭代直到不能加入新的点为止.

经过复原处理即可进行各种分段的结果实验了.

4 实验结果和分析

图 4 中签名 1、签名 2 是用本文的算法对同一个人在不同场境下签下的 2 个本人签名的分段结果, 签名 3 是他人模仿的签名分段结果.

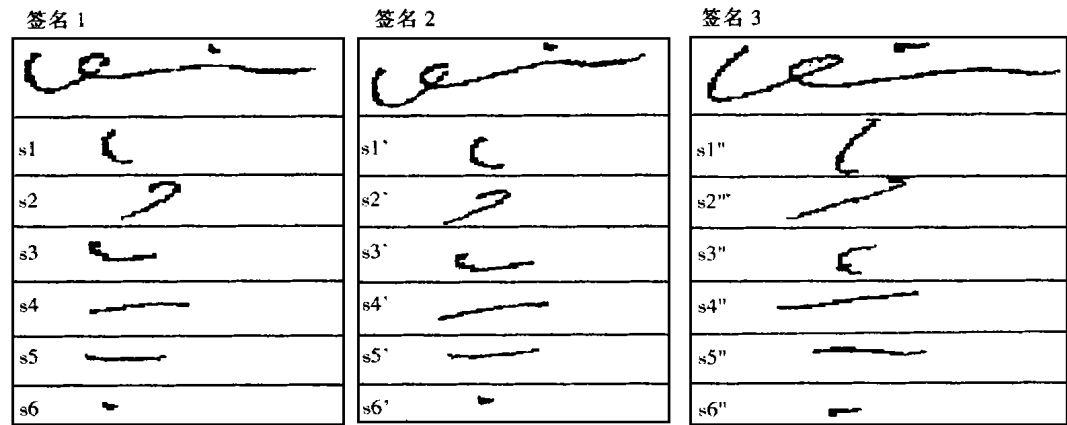


图 4 签名分段结果
Fig.4 Result of signature segments

对应段结构相关性分析如下.

1) 骨架分段结构描述. 骨架分段被描述为 2 个多维向量, 向量的每 1 个分量表示分段中的每 1 步的方向值, 这个方向值由以下计算: 从端点开始的第 1 步方向值为 0, 随后当前步的方向值由前 1 步与当前步的偏角按方向码 C 计算. 因为每个分段都有 2 个端点, 因此分别从分段的始点和终点开始进行计算便可以得到 2 个结构描述向量, 分别表示为 V_s 和 V_e .

2) 结构特征的提取. 把 1) 所得的向量的分量个数记为 n , 取 V_s 向量的第 $[n * i / 8]$ (i 取 1~8 的整数, $[]$ 表示取整) 个分量组成结构特征向量的前 8 个分量, 取 V_e 向量的第 $[n * i / 8]$ (i 取 1~8 的整数) 个分量组成结构特征向量的后 8 个分量, 即可提取出分段的结构特征向量.

3) 计算相关度 S . 按以下公式计算

$$S(X, Y) = 2 \sum_{i=0}^{15} X_i Y_i \sqrt{\left(\sum_{i=0}^{15} (X_i)^2 + \sum_{i=0}^{15} (Y_i)^2 \right)}$$

其中, X 和 Y 分别代表 2 个对应的分段的结构特征向量, X_i , Y_i 分别表示 X 和 Y 的第 i 个分量, 计算结果见表 1.

表 1 签名相关度

Tab. 1 Corelation degree of signatures

签名 1	S1	S2	S3	S4	S5	S6
签名 2	0.975	0.908	0.985	0.947	0.928	0.880
签名 3	0.765	0.810	0.825	0.796	0.818	0.535

通过以上实验可以看出, 本文提出的基于骨架的签名分段算法, 由骨架把签名分为段, 这种分段算法既达到了把签名的整体复杂度均衡局部化的目的, 又基本上保持了真实签名分段在数量上和结构上的稳定性, 降低了各种由于外部因素引起的类内特性之间的差别, 突出了类间特性的表现, 有利于签名鉴别的进行.

参考文献:

[1] BROMLEY J. Signature verification using a ‘ siamese’ time delay neural network[J] . Internat J Parttern Recognition and Artifitial Intelligence, 1993, 7(4): 669 ~ 688.

[2] 柯晶, 乔谊正. 联机中文签名鉴别的一种局部弹性匹配方法[J] . 中文信息学报, 1998, 12(1): 57 ~ 62.

A Framework-Based Signature Segmentation Algorithm

WU Xiao-ming^{*}, CHEN Yi-xin

Abstract:A framework-based signature segmentation algorithm is presented.Applying this algorithm to segment signature, the authors effectively implement the even structure complexity segment and achieve a fine result.

Keywords:automatic signature verification; signature segment; direction code; degree of complexity

^{*} Department of Mathematics, Zhongshan University, Guangzhou 510275, China

(C)1994-2019 China Academic Journal Electronic Publishing House. All rights reserved. <http://www.cnki.net>