

基于轮廓的汉字笔画分离算法

程立 王江晴 李波 田微 朱宗晓 魏红昀 刘赛

(中南民族大学 武汉 430074)

摘要 由汉字的轮廓提取笔画,其关键之处在于找到笔画相交处的交点。在提取汉字图像轮廓和特征点的基础上,提出了笔画分离算法,并通过程序实现了该算法。实验证明该算法对于提取印刷体汉字和无连笔的手写体汉字是有效的。

关键词 笔画分离,轮廓检测,特征点,凹点,轮廓跟踪

中图法分类号 TP391.1 **文献标识码** A

Algorithm on Strokes Separation for Chinese Characters Based on Edge

CHENG Li WANG Jiang-qing LI Bo TIAN Wei ZHU Zong-xiao WEI Hong-yun LIU Sai

(South-Central University for Nationalities, Wuhan 430074, China)

Abstract To extract strokes of Chinese characters from its contour, the key lies in seeking the intersection points of intersecting strokes. An algorithm used in separating strokes was proposed based on extracting contour of Chinese character picture and detecting the feature points. At last the algorithm was implemented by VC program. Test result shows the algorithm is effective in extraction of strokes of printed Chinese characters and handwritten Chinese characters without script strokes.

Keywords Strokes separation, Contour extraction, Feature point, Concave, Contour tracking

1 引言

汉字识别是模式识别的一个分支。经过科研工作者的不懈努力,汉字识别中的联机手写体识别和脱机印刷体识别已日趋成熟,出现了很多有实用价值的产品。目前只有脱机手写体识别还不能达到令人满意的效果^[1]。脱机手写体汉字识别具有广阔的应用前景,它可以广泛应用于各种纸质文档的高速自动录入、图书资料的数字化、邮件的自动分拣等方面。因此,如何解决这些问题已成为研究的热点。

汉字识别中,根据识别时依据的特征的不同,主要有两大类:统计特征和结构特征。其中结构特征是建立在笔画的基础之上,以笔画或笔画构成部件之间的空间结构关系来描述汉字。目前笔画提取算法主要有两类:基于细化图像的算法和基于原始图像的算法。基于细化图像的算法主要有细化法^[2]、轮廓信息法^[3]和段化法^[4]等。其中细化法容易造成伪笔段和形状扭曲,而且即使采用各种修正方法,也不尽如意^[2]。基于原始图像的方法则是根据原始图像的某些特征,用各个方向上的直线段去拟合图像的各部分,如行程长度编码法、轮廓信息法^[5]等,其优点是速度较快,但目前此类方法仅局限于印刷体汉字或字母的识别。

本文给出的方法是在获取汉字图像轮廓的基础上,检测出轮廓曲线上的特征点,包括凸点和凹点,然后按一定的方向

对轮廓进行跟踪,在跟踪的过程中,将相交笔画的交点,也即凹点连接起来,实现相交笔画的分离。该算法的特点在于:(1)无需对字符图像的大小进行规范化,而是直接对原图像的轮廓进行处理得到笔画。(2)每完成一次轮廓的跟踪可以提取一个完整的笔画,不需要再进行合成处理。(3)提取的笔画轮廓各种参数和特征点齐全,易于识别。实验表明,本文设计的笔画提取算法正确率较高,稳定性较好,而且对倾斜的字体不用校正也能进行正确的处理。

2 字体图像的轮廓提取

本算法是基于文字图像轮廓的,所以首先必须提取文字的轮廓。提取轮廓后,去掉了文字图像中的大量冗余信息,便于后续处理中获取轮廓中的各种特征点。

轮廓提取,也就是边缘检测,是图像处理中常见的处理方法。数字图像的边缘检测方法主要有:水平垂直差分法、Roberts算子法、Prewitt算子法、Sobel算子法、Log算子法、Canny算子法^[6]。这里采用的是水平垂直差分法,实现起来比较简单。在提取轮廓之后,得到的是多个封闭多边形,也就是每一个点都有且只有两个连接点,但在轮廓的锐角处可能会产生一些端点,这种点只有一个连接点,在此称为孤立点,为便于跟踪,需将这些点剔除。

经过处理后的轮廓形成一个封闭的复杂多边形,多边

到稿日期:2012-09-15 返修日期:2012-12-24 本文受国家自然科学基金项目(60975021),中南民族大学中央高校基本科研业务费专项资金项目(CZQ12005)资助。

程立(1976—),男,博士生,主要研究方向为图像处理、模式识别;王江晴 博士,教授;李波 博士;田微、朱宗晓、魏红昀 博士生。

形上的每一个点均有且只有两个连接点。

3 特征点的检测

3.1 特征点的检测

和细化后的骨架不同,字体轮廓上的所有点都有且只有2个连接点,不存在端点和分叉点的情况。笔画轮廓上的特征点主要有:笔画的起始和停止处的角点、笔画相交处的交点和笔画的转折点。图1是汉字“堆”的轮廓,其中编号为每个封闭轮廓上的特征点的序号。特征点的检测采用 Douglas-Peucker 算法,为能沿轮廓按一定方向(顺时针或逆时针)顺次获取轮廓上的特征点,需对传统 Douglas-Peucker 算法做相应改进^[7,8]。提取了轮廓上的特征点后,得到两个序列:轮廓上的点序列和特征点序列,可表示为:

edge[k]=(p₀,p₁,...,p_i,p_{n-1},p_n) (1)

fp(k)=(i₀,i₁,...,i_{m-1},i_m) (2)

式中,edge[k]为第k个轮廓的点序列,fp(k)为轮廓k的特征点序列,其中i_j表示第j个特征点在edge[k]中的序号,第j个特征点的上一点last为i_j-1,下一点next为i_j+1。

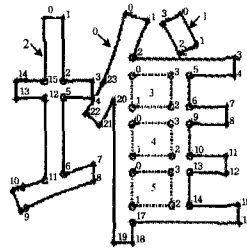


图1 “堆”字轮廓及其特征点

通过观察文字轮廓可发现,笔画相交处的点必定是内凹的点,即凹点(图1中标有小圆圈的为凹点,标星号的为凸点)。如图1中“堆”字“土”旁中间的横竖笔画相交处的4个特征点2,5,12,15都是凹点。

虽然汉字的结构非常复杂,但对于笔画的提取,总是将相交的笔画分离出来,所以有必要对汉字的笔画相交的形式进行分析,找出分离笔画的算法。

汉字笔画相交的形式可分为X型、T型、K型、L型^[9]以及粘连5种,如图2虚线框中所示。

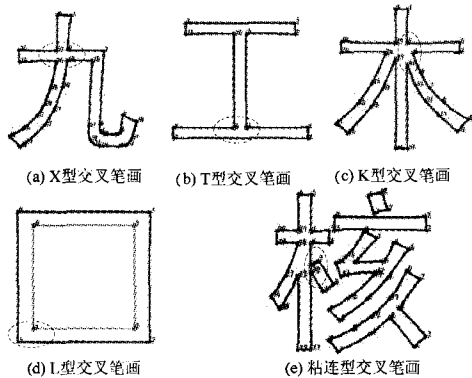


图2 汉字相交笔画的形式样例

显然,具有相交笔画的部件形成的轮廓必定是凹多边形(在此图1和图2中虚线多边形应看作凹多边形)。分离笔画,就是将相交处的凹点连接起来,即凹点匹配。所以提取相

交笔画的轮廓之关键是将笔画交接处的凹点连接起来。对于不同形式的笔画相交,凹点匹配的算法不同。具体到每种形式相交笔画的分离方法将在文中后续的凹点匹配中专门论述。

3.2 凹点的检测

多边形凹点可以通过叉乘结果的符号来确定。要确定某一特征点是凹点还是凸点,可通过求以该点为中心、其上一特征点和下一特征点相对于该点的坐标的叉乘,由结果的符号来判定该点是凹点还是凸点。如某特征点的上一特征点和下一特征点相对于该点的坐标分别为(x₁,y₁),(x₂,y₂),则其叉乘为:

cross=x₁×y₂-x₂×y₁ (3)

叉乘的符号除了与顶点的凹凸性有关外,还与特征点排序的方向有关。对于一个独立轮廓来说,第一个特征点一定是凸点,以此可以确定特征点是按逆时针还是顺时针排列的。另外,对于轮廓内缘来说,因为其空白区在轮廓内部,判别凹点时叉乘结果的正负与外轮廓恰好相反。最后,还可用特征点与其相邻的两特征点夹角进行限制,以剔除夹角接近直线的凹点。

3.3 特征点的编号

轮廓的特征点可以按顺时针或逆时针进行编号。为便于区分不同轮廓,这里对轮廓也进行了编号,如图1中小箭头末端及虚线矩形中央的数字便是不同轮廓的编号。轮廓的编号按程序处理中从上至下、从左到右轮廓最高点出现的次序编排。而且对于轮廓来说,有内缘和外缘之分。如果字体原始图像中有部件构成了封闭的连通区域,如汉字中的“口”、“田”等,这时抽取轮廓后,就会形成一个连通区域的外轮廓完全包含其内轮廓的情况。在此将外轮廓称为父轮廓,而将其所包含的内轮廓称为子轮廓。父轮廓和子轮廓均是一个部件的边缘。如图1中轮廓3,4,5均是轮廓0的子轮廓。将父轮廓和其子轮廓(可能没有)称为独立轮廓。独立轮廓即为一个部件的边缘,将独立轮廓的轮廓线之间的空白涂黑即为原始图像。

在给特征点编号时,将一个独立轮廓的父轮廓和子轮廓按相反顺序编号。如图1中实线轮廓编号均按顺时针,而轮廓0的子轮廓3,4,5则均按逆时针编号,这样就使得一个笔画轮廓上的特征点序列总是按编号增加排列的。如图1中“堆”字右边部件中间的竖笔,其特征点序列为:

(0,5)→(0,6)→(0,9)→(0,10)→(0,13)→
(0,14)→(5,2)→(5,3)→(4,2)→(4,3)
→(3,2)→(3,3) (4)

其中,括号中的第一个数字为轮廓编号,第二个数字为特征点编号,点(m,n)即表示轮廓m上的第n个特征点,将其称为广义点。有了轮廓点坐标系列和特征点编号系列,就很容易找到广义点的真实坐标。从式(4)可以看出,一个笔画的轮廓上的特征点是编号增大的点系列,其中广义点编号不连续的地方为笔画相交之处。由此可见,提取笔画的关键之处在于将笔画相交的地方的凹点连接起来,即凹点匹配。

4 凹点匹配

凹点匹配是提取笔画的关键。对于前文所述的5种相交的情况,其分离算法如下所述。

(1)X型:即汉字中两个笔画的中间相连,如图2中(a)图的“九”字。对此种笔画,配对凹点应满足关系:① $last, c1, c2, next$, 4点近似共线;②直线 $\langle c1, next \rangle$ 和 $\langle last, c2 \rangle$ 近似平行;③两配对点为满足条件①②中距离最近的两凹点。而且要提取X型相交的笔画,需两次匹配。如图2中(a)图(0,1),(0,2),(0,19),(0,20) 4点共线,(0,2),(0,3)连线与(0,18),(0,19)连线平行;(0,29),(0,30),(0,33),(0,0) 4点共线,(0,30),(0,31)连线与(0,32),(0,33)连线平行,此时有4个凹点,形成两对凹点对,即(0,2),(0,19)和(0,30),(0,33)。匹配条件为:

$$\left(\frac{360 - (\angle(last, c1, c2) + \angle(c1, c2, next))}{ALFA} \right)^2 + \left(\frac{distance}{DIST} \right)^2 < CR1 \quad (5)$$

$$\left(\frac{180 - \max(\angle(last, c1, c2), \angle(c1, c2, next))}{BETA1} \right)^2 + \left(\frac{\angle(\langle next, c1 \rangle, \langle c2, last \rangle)}{BETA2} \right)^2 + \left(\frac{distance}{DIST} \right)^2 < CR2 \quad (6)$$

式中, $last, next$ 表示写在其相邻位置的凹点 $c1$ 或 $c2$ 的上一特征点或下一特征点, $ALFA$ 和 $BETA1, BETA2$ 为设定的角度阈值, $distance$ 为 $c1$ 和 $c2$ 间距离, $DIST$ 为距离阈值, CRn ($n=1, 2, 3, 4$)为判据阈值。

X型交叉处在提取相交的第一个笔画后,与之相连的后续笔画的起止点在上述4个凹点中,但需重新配对,即将上述的两对凹点配对关系变为(0,2),(0,33)和(0,19),(0,30),后续笔画起止点为(0,2),(0,33)。

(2)T型:即汉字中一个笔画的一端和另一个笔画的中部充分相连,如图2中(b)图“工”字虚线框里结构即为典型例子。对于此种笔画,配对凹点应满足关系:① $next, c1, c2, last$ 近似共线;②直线 $\langle last, c1 \rangle$ 和 $\langle c2, next \rangle$ 近似平行;③两配对点为满足条件①②中距离最近的两凹点。如图2(b)中“工”字凹点(0,4)满足(0,5),(0,4),(0,9),(0,8) 4点共线,(0,3),(0,4)连线与(0,9),(0,10)连线平行,可提取笔画“竖”,并产生后续笔画起止点(0,4),(0,9)。凹点匹配条件可表示为:

$$\left(\frac{360 - (\angle(next, c1, c2) + \angle(c1, c2, last))}{ALFA} \right)^2 + \left(\frac{distance}{DIST} \right)^2 < CR3 \quad (7)$$

$$\left(\frac{180 - \max(\angle(c2, c1, next), \angle(last, c2, c1))}{BETA1} \right)^2 + \left(\frac{\angle(\langle last, c1 \rangle, \langle c2, next \rangle)}{BETA2} \right)^2 + \left(\frac{distance}{DIST} \right)^2 < CR4 \quad (8)$$

在T型交叉处产生的两个笔画,与提取的笔画相连的后续笔画起止点为该配对凹点。如此处两个笔画为(0,3)→(0,4)→(0,9)→(0,10)和(0,4)→(0,5)→(0,6)→(0,7)→(0,8)→(0,9),分别对应“竖”笔和“横”笔。其中广义点不连续的地方即为配对的凹点。

(3)K型:即汉字中3个或4个笔画相交于一处的结构,如图2中(c)图“木”字即为两个K型结构。对于K型相交结构的部件来说,满足当前轮廓所在笔画凹点 $c1$ 的4点共线和后续笔画的平行的匹配凹点不为同一点,而是两个不同凹点。如“木”字结构中,“竖”笔上的凹点2的4点共线凹点为14,即(0,1),(0,2),(0,14),(0,15)共线,凹点2的平行匹配凹点为5,即(0,2),(0,3)连线与(0,4),(0,5)连线平行。此时产

生两个新的凹点对(0,2),(0,6)和(0,6),(0,14)作为后续笔画起止点。所以K型相交笔画的分离算法采用的条件与式(5)、式(6)相同,但会得到两个不同的匹配凹点。

(4)L型:L型的相交笔画,是汉字中诸如“口”部的4个拐角的地方,如图2(d)所示。要进行处理,需首先判明拐角的方位,因为汉字中不存在横竖笔画在左上角和右下角相连,而右上角则一定相连,左下角有相连的也有不相连的,需进行区分。拐角的判别可表示为:

$$\begin{cases} |90 - \angle(last, corner, next)| < C \\ sign(x) = mid(x) - pc(x) \\ sign(y) = mid(y) - pc(y) \end{cases} \quad (9)$$

式(9)中的第一个表达式用于限定拐角角度在 90° 左右, C 为设定的阈值。而拐角 $corner$ 的方位由其上一点 $last$ 和下一点 $next$ 连线的中点 mid 的 x 坐标和 y 坐标与该点的坐标大小决定,即由坐标差的符号确定。式(9)中 pc 为广义点 $corner$ 的实际坐标点。图3示出右下角的情况。显然此时有 $sign(x) < 0, sign(y) > 0$ 。判明了右下角后,另外还需保证 $corner$ 两边的边长大于笔画宽度。在满足这些条件后,拐角 $corner$ 的下一点必须在位于其右上角独立轮廓的凹点中寻找(见图3中虚线所示)。最后需要注意的是,对于L型的相交笔画,两配对的点和凹点有所不同,此时两配对点总是一个为凸点,一个为凹点。

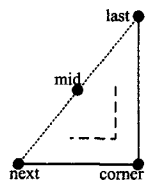


图3 右下角判别示意图

(5)粘连型:为两个笔画有部分相连的情况,如图2(e)所示。此时仅满足X型结构中的4点共线条件,且两凹点间距离小于平均笔画宽度。

在凹点的匹配过程中,不能预先知道是哪种笔画相交类型,在处理中需计算X型和T型的所有角度条件,由此得到相应的备选点。判据条件和其所得到的备选点的关系如表1所列。

表1 判据表达式和备选点关系

判据条件	式(5)	式(6)	式(7)	式(8)
备选点	cd1	cd2	cd3	cd4

除此之外,还需满足凹点对 $c1$ 和其匹配凹点 $c2$ 在距离上的约束,此处以笔画宽度的3倍为最大距离。由此可知,对于已知凹点 $c1$ 来说,根据4个判据条件会得到4个备选点,这4个备选点有可能存在,也有可能不存在,最后由表2确定选取哪一个作为与凹点 $c1$ 匹配的凹点 $c2$ 。

表2 备选点确定表

cd1	cd2	cd3	cd4	c2
P1	P1	ANY	ANY	P1
P1	P2	ANY	ANY	P1*
P1	NO	ANY	ANY	P1
NO	ANY	P3	P3	P3
NO	NO	P3	P4	P3*
NO	NO	NO	ANY	NO

表中 ANY 为任何结果,NO 表示不存在满足条件的点。其中带有 * 号的是需要做特殊处理的地方,此时 $cd1$ 和 $cd2$ 都存在且不相等,即 4 点共线的配对点和平行的配对点不是同一点,对应于上述 K 型相交的笔画。如何处理此问题在轮廓跟踪中将做进一步说明。

5 轮廓跟踪

笔画的提取是在对轮廓的跟踪过程中获取的,每完成一次跟踪可提取一个笔画的轮廓。在轮廓跟踪过程中,依据当前点是凹点还是凸点,按表 3 的次序寻取其下一点。

表 3 轮廓跟踪中下一点的选择次序表

c1 类型	首选	次选	末选
凸点	拐角	next	
凹点	凹点 c2	拐角	next

为保证路径唯一,在跟踪过程中对经历的特征点出现的次数进行记录,完成一个独立轮廓的所有笔画提取,满足条件为:(1)凸点只出现 1 次;(2)具有配对凹点的凹点出现 2 次。并且在轮廓的跟踪过程中,先提取的笔画轮廓中的配对凹点作为后续提取的笔画的起始点和终止点,整个笔画提取过程中凹点对的出现可以用树的结构来表示。现以图 1 中“堆”字的右边结构,即轮廓 0,3,4,5 组成的独立轮廓的笔画提取来进行说明。对于一个独立轮廓来说,提取第一个笔画时的起始点和终止点有两种方式,一种是起始点和终止点不同,一种是起始点和终止点为同一点。此处取同一点,即对独立轮廓进行跟踪,起始点和终止点均为 $(0,0)$,则提取该独立轮廓中的“撇”笔,其特征点系列为:

$(0,0) \rightarrow (0,1) \rightarrow (0,2) \rightarrow [(0,2), (3,0); (3,0), (0,20)] \rightarrow (0,20) \rightarrow (0,21) \rightarrow (0,22) \rightarrow (0,23) \rightarrow (0,0)$

其中,[]内为两对新产生的配对凹点,是因凹点 $c1(0,2)$ 的配对凹点备选点为 $cd1(0,20)$, $cd2(3,0)$,为表 2 中第 2 种情况,此时增加两对匹配凹点 $(0,2), (3,0)$ 和 $(3,0), (0,20)$,正好是最上面的“横”笔和左边的“竖”笔的起始点和终止点,方括号内的点不出现在“撇”笔的轮廓中。整个过程用图 4 的树结构表示。

图 4 的根节点为 $(0,0)$,即第一个笔画“撇”的起止点,其他节点为在提取笔画过程中的匹配凹点。图 4 中有些节点指出了所提取的笔画,而左子树的最下层的 5 个叶子节点并未指出提取的笔画,这是由于这 5 个叶子节点是其父节点 $(0,5), (3,3)$ 所提取笔画的轮廓的中间节点,即中间“竖”笔与 4 个“横”笔相交的地方。处理中笔画出现的顺序按照先根节点,然后同一层的左子树,再右子树,一层一层地提取出笔画。所以这里笔画出现的顺序为:撇横竖竖横横横。而且 5 个叶子节点中凹点对 $(0,6), (3,2), (0,9), (4,3), (0,10), (4,2)$ 以及 $(0,13), (5,3)$ 均为 X 型交叉处的凹点对经重新配对后的凹点对。显然,X 型交叉处的凹点对在相交两笔画中的配对组合不同。如图 1 中右边部件的中间竖笔和第二横笔为 X 型相交,在提取竖笔时凹点匹配组合为 $(0,6), (0,9)$ 和 $(4,3), (3,2)$,而在提取横笔时凹点对组合为 $(3,2), (0,6)$ 和 $(0,9), (4,3)$ 。左子树的 5 个叶子节点在完成相交的竖笔和横笔的提取后,使用次数均为 2 次,不再以这些节点为起止点提取笔

画,否则会出现笔画的重复。X 型相交处的凹点交换,对应于汉字四叉点、五叉点和六叉点,需进行交换的 4 个凹点分别近似构成矩形或平行四边形、直角梯形和梯形。如图 5 所示,标有数字的凹点分别近似构成上述 4 种图形。

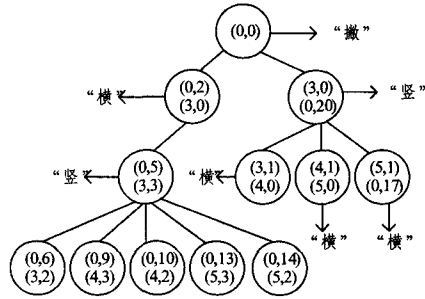


图 4 笔画起止点树形图

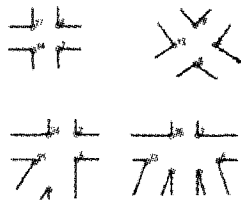


图 5 配对凹点需进行交换的 4 种情况

6 实验结果

以此算法设计了具体的 VC 实现代码,并对一定量的文字图像进行了笔画提取的处理。其中包括笔画从 2 笔到 30 笔的印刷体黑体汉字 200 个,以及部分手写体汉字。其中印刷体笔画提取正确率为 98%,手写体则受书写潦草程度影响,对带连笔手写体汉字提取不出连在一起的笔画,且如果轮廓出现大的凹凸不平,会严重影响笔画提取的正确率。图 6 和图 7 为黑体汉字和手写体汉字的处理结果,图 8 为选取的部分实验结果图。为比较基于轮廓的笔画提取算法和基于骨架的提取算法的处理速度,这里以图 8 中的汉字为例。表 4 中本算法时间为相应字体的笔画提取所需要的时间,而细化(采用查表法)时间仅仅为字体骨架提取时间。实验平台为 Windows XP 操作系统,CPU 频率 2.8GHz,内存 2G,编程环境为 VC6.0,实验样本中印刷体汉字为 WORD 中截取的 72 号黑体汉字,显示比例 500%(图 8 为缩小过的处理结果)。由表 4 可见,本文提出的算法相对于基于细化的笔画提取算法在速度上具有较大优势。



图 6 “堆”字的原始图像、轮廓及提取的笔画

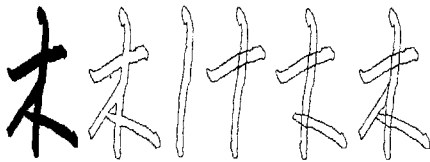


图 7 手写体汉字的笔画提取过程图

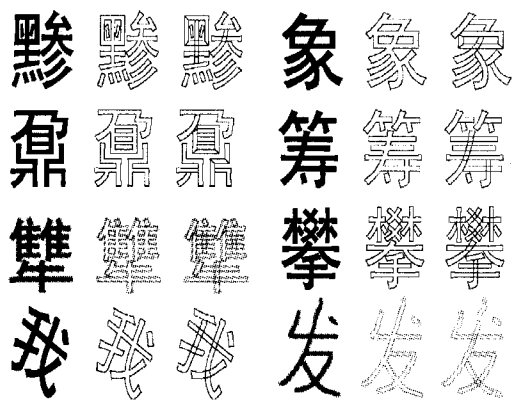


图8 部分实验结果图

表4 本算法时间与单字细化时间比较表(时间 ms)

汉字	本算法	细化时间	汉字	本算法	细化时间
黠	328	594	髯	282	687
象	250	797	群	265	656
鼎	282	735	攀	328	766
筹	281	734	我	218	859
攀	313	812	发	94	78

结束语 不同于基于细化图像和基于原始图像的笔画提取算法,本文提出了基于轮廓的汉字笔画提取算法。该算法首先检测文字的轮廓,然后提取轮廓的特征点,并在特征点中区分出凹点,最后对轮廓进行跟踪,找到笔画相交处的凹点,将两个匹配的凹点连接起来,完成一次轮廓的跟踪后,即可分

(上接第301页)

制节点对一个AS进行联合控制的情况下,为多个控制节点提供一致性视图的必要性,并给出了各个控制节点独自构建网络视图造成的问题;然后针对这些问题,提出了可信可控网络中为多个控制节点构建一致性视图的机制,包括主控制节点的选举算法和基于选举的一致性视图的构建算法,保证了视图版本一致性,并对该算法构建的视图的一致性进行了分析;最后在控制信息数据库的基础上,为控制节点提供了AS内设备粒度和协议粒度的一致性视图。

本文解决的一致性视图问题是可信可控网络中多个控制节点对AS进行联合控制的基础,然而多个控制节点对AS进行有效控制还涉及到AS外信息的共享问题。为了更好地提高网络控制的效率,建立多个控制节点AS间有效的信息共享机制将是非常有意义的工作。

参考文献

- [1] Bouabene G, Jelger C, Tschudin C, et al. The Autonomic Network Architecture(ANA)[J]. IEEE Journal on selected areas in communications, 2011, 28(1)
- [2] Iqbal H, Znati T. Distributed control plane for 4D architecture [C]// the Proceeding of Global Telecommunications Conference, Washington DC, US, 2010: 1901-1905
- [3] 吴杰. 分布式系统设计[M]. 北京:机械工业出版社, 2011

离出一个完整的笔画轮廓,实现笔画的提取。实验证明该算法对提取轮廓较平滑的字体笔画是有效的。

参考文献

- [1] 靳天飞. 脱机手写汉字识别中笔段提取算法研究[J]. 山东大学学报:理学版, 2008, 43(5): 39-44
- [2] Yu Qiao, Mikihiro N, Makoto Y. A framework toward restoration of writing order from single-stroked handwriting image[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(11): 1724-1737
- [3] L'Homer E. Extraction of strokes in handwritten character[J]. Pattern Recognition, 2000, 33(7): 1147-1160
- [4] 刘峡壁, 贾云得. 一种字符图像线段提取及细化算法[J]. 中国图象图形学报, 2005, 10(1): 48-53
- [5] 何浩智, 朱宁波, 刘伟. 基于骨架点分布规律的汉字笔段提取算法[J]. 计算机工程与应用, 2007, 43(22): 83-86
- [6] 韩丽娟, 王勇, 耿辉, 等. 一种新的边缘检测算法在图像处理中的应用[J]. 仪器仪表与分析监测, 2008(4): 18-20
- [7] 程立, 王江晴, 田微, 等. 手写体女书文字规范化处理程序研究[J]. 中南民族大学学报:自然科学版, 2012, 31(1): 93-96
- [8] 程立, 王江晴, 田微, 等. 改进 D-P 算法在图像轮廓平滑中的应用[J]. 计算机工程, 2012, 38(17): 232-234
- [9] 袁媛, 刘文才. 基于形状分割的手写汉字笔画提取方法[J]. 计算机工程与科学, 2010, 32(12): 57-60
- [4] King G T, Gendreau T B, Ni L M. Reliable election in broadcast networks[J]. Journal of Parallel and Distributed Computing, 2009(1): 521-546
- [5] SSF-Scalable Simulation Framework[OL]. <http://www.ssfnet.org>, 2010-04-23
- [6] Dolev D, Klawe M, Roth M. An $O(n \log n)$ unidirectional distributed algorithm for extrema finding in a circle[J]. Journal of Algorithms, 1982(3): 245-260
- [7] Zhang In, Perrig A, Zhang Hui. Centaur: A Hybrid Approach for Reliable Policy-Based Routing[C]// the 29th International Conference on Distributed Computing Systems(ICDCS). June 2012
- [8] Mans B, Santoro N. Optimal elections in faulty loop networks and applications[J]. IEEE Transactions on Computers, 1998, 47(3): 286-297
- [9] Wang P, Luo J Z, Li W, et al. Control Information Description Model and Processing Mechanism in the Trustworthy and Controllable Network[C]// Proceedings of the 11th IFIP/IEEE International Symposium on Integrated Network Management (IM09). New York, June 2011: 398-405
- [10] Wang Pan, Li Si-kun, Cai Xun, et al. The Construction A lgorithm of Binary Interval Tree Nodes based on Lattice Patition [J]. Journal of Computer Aided Design & Computer Graphics, 2012, 23(7): 1115-1130