

DOI: 10.19322/j.cnki.issn.1006-4710.2016.01.007

一种书法字骨架提取优化方法

张九龙, 王夏妮, 张镇东, 郭璐铭

(西安理工大学 计算机科学与工程学院, 陕西 西安 710048)

摘要: 书法字的骨架提取即细化是书法风格研究的重要先决步骤。针对常见细化算法结果中产生的锯齿、非单像素、分叉点畸变和毛刺等问题进行改进。引入一种旋转不变性的细化算法得到汉字的中轴线。接着分以下两步进行优化: 首先在保证图像连通的条件下进行单像素化处理; 其次引入最大圆方法进行分叉点合并。实验结果表明, 两步法得到的细化结果是单像素且无毛刺和分叉点的, 为后续笔画提取和书法风格研究奠定良好的基础。

关键词: 图像形态学; 书法; 骨架提取; 分叉点检测; 分叉点合并

中图分类号: TP391.4 **文献标志码:** A **文章编号:** 1006-4710(2016)01-0035-04

An optimization algorithm for Chinese characters skeleton extraction

ZHANG Jiulong, WANG Xiani, ZHANG Zhendong, GUO Luming

(School of Computer Science and Engineering, Xi'an University of Technology Xi'an 710048, China)

Abstract: Character skeleton extraction is one of the important preprocessing steps for the calligraphy style evaluation. Traditional thinning algorithms have the drawbacks of zigzag, nonsingle pixel, fork point distortion, spurious branches, etc. This study aims at solving these problems. Based on a rotation invariant algorithm the provisional skeleton is obtained, with two steps taken to optimize the result. First, with the guarantee of the connectedness of skeleton, we perform a process by which to reduce the skeleton to single pixel in width. Second, we employ the maximum circle method to merge the fork points. The experimental results show that the two-step optimization can obtain a better skeleton compared with some common methods.

Key words: image morphology; Chinese calligraphy; skeleton extraction; fork point detection; fork point merging

细化的常见方法有距离变换和轮廓点删除两大类。距离变换类的方法对轮廓的细微变化不具有鲁棒性,故本文主要研究轮廓点删除算法。Suen^[1]提出了一种经典的快速并行细化算法,每次迭代主要去除图像的边界点并保留骨架的关键点,在保持笔画骨架连通性的同时提高了细化的速度和效率,但效果较为粗糙。Gonzalez 将骨架定义为目标轮廓点中最大内切圆的圆心的集合^[2],但会出现较多骨架断裂。S. Huang^[3]运用 Bernstein-Bezier 曲线对细化结果进行拟合来改善分叉点偏移的现象,贾云德在字符图像线段提取及细化过程^[4]中应用了此方法。刘宏申^[5]采用形态学方法中的击中击不中方法进行笔画细化,但其中如何选取结构对元素对细化

结有很大影响。在书法文化计算及应用方面,相关的部分文献有吴江琴的书法字检索^[6]和邓茜芸的瓦当文字识别^[7]等研究。

在击中击不中运算中,通常设置模板从四个或八个方向剥离像素。这类算法应用于书法汉字细化的缺陷是有些方向上的笔划效果不佳,因为汉字笔划并非只有水平、垂直和对角线方向。本文采用 Ahmed 提出的旋转不变性细化算法^[8],即 A-W 方法。该方法能得到字符的中轴线并较好地保持原字符的形状特征。但该方法的骨架可能存在非单像素宽度,故引入 Rockett 的单像素化算法^[9]进行改进。虽然改进后骨架为单像素的,但分叉现象仍然存在。具体表现为一个四分叉点会畸变为两个三分叉点等

收稿日期: 2015-11-25

基金项目: 国家自然科学基金资助项目(61401355);陕西省教育厅专项科研计划资助项目(2013JK1135);西安市碑林区科技计划资助项目(GX1410)

作者简介: 张九龙,男,副教授,博士,研究方向为图像处理与模式识别。E-mail: zhangjiulong@xaut.edu.cn

现象。采用分叉点及特征点判断算法^[10],确定所有的分叉点和端点。最大圆方法^[3]是进行分叉点合并有效的一个方法,本文在确定分叉点后引入该方法合并分叉点,取得了优于上述单个方法的细化结果。

1 A-W 细化算法

A-W 细化算法包括若干种细化模板,通过迭代匹配,当中心像素点与模板匹配时,则删除该点,直到图像不再变化为止。设待处理图像为二值图像,1为目标像素点,0为背景像素点。每次迭代的目标是删除边界像素点,但前提是不会破坏图像的连通性。以目标像素点和它的八邻域像素点组成 3×3 大小的目标窗口,判断是否符合以下 4 种情况之一。

1) 若八邻域全为 0,中心像素点则为孤立点,该点不应该被删除,否则会删除整个符号。

2) 若八邻域全为 1,则中心像素点不属于边界,因此不应删除。

3) 若八邻域的环状结构中有 1 到 7 个连续的 0,那么就会有 7 到 1 个连续的 1。这类分为 3 个子类:①八邻域中仅包括 1 个 1,7 个 0;②八邻域中只包括 2 个连续的 1,6 个 0;③八邻域中包括至少 3 个连续的 1,其它为 0。

4) 八邻域中白色像素点不是以连续的形式出现,针对所出现的几种情况均可做相应处理。

以上 4 种情况可以归结为 20 种模板^[8],在每次迭代中,将考虑所有的像素点并且删除和模板匹配

区域的中心点。通过重复上述过程直到图像不再变化为止,这样就得到了骨架。

2 两步法骨架优化算法

采用 A-W 算法进行初步细化后,会存在一些缺陷。如在笔画的撇和捺的部分存在非单像素宽的线,在笔画交叉的地方存在多个分叉点,在笔画端点处出现毛刺,这些都导致骨架失去了原先笔画的特性。

2.1 单像素化处理

首先,检测两个像素宽的地方,即水平方向上两像素宽 $[0\ 1\ 1\ 0]$ 和垂直方向上两像素宽 $[0\ 1\ 1\ 0]^T$,并利用骨架的连通性进行单像素处理。算法的基本思想是为每个中心像素点和它的八邻域构造一个无向连通图,若删除中心像素点不会造成不连通,则删除,否则不应该删除。

图 1 为处理前的两像素宽的情况。其中,(a)为某次细化后的初步结果,中间的两个像素点 1 组成了水平方向上双像素宽的线;(b)为构造的无向连通图,可见若删除 x_0 不会影响图的连通性,所以可以将其删除;(c)为无向连通图对应的邻接矩阵。

图 2 列举另一种中心点不应被删除的情况。(a)为初步细化的结果;(b)为构造的无向连通图,若删除 x_0 ,则会导致图不连通,所以这种情况下不应该删除。

为每个符合两像素宽的像素点及其八邻域构造邻接矩阵,更方便进行判断是否应该删除中心像素点。图 3 为单像素化处理前后的局部放大图。

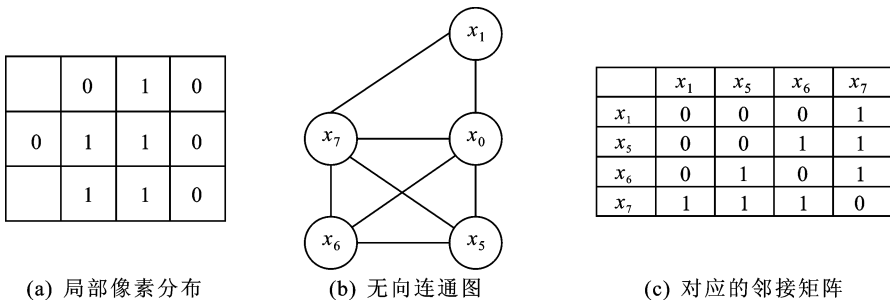


图 1 细化后局部像素情况
Fig. 1 Local pixels after thinning

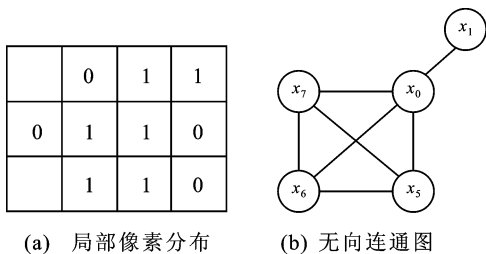


图 2 中心像素点不应该被删除的反例
Fig. 2 Counter example for the fact that center points should not be deleted

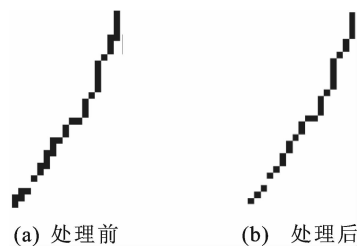


图 3 单像素化处理结果对照
Fig. 3 Result with reduced single pixel in width

2.2 分叉点合并

单像素化处理之后,还没有解决分叉点畸变的缺陷。例如图4所示,原来的一个4分叉点被转为两个3分叉点,并且在笔画端点处有毛刺。在此首先检测分叉点和端点等特征点,然后引入最大圆方法进行分叉点合并和毛刺去除。

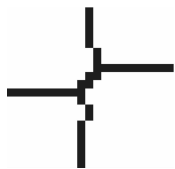


图4 一个4分叉点转化为两个3分叉点
Fig.4 Two 3-fork points split by a 4-fork point

2.2.1 特征点检测

用集合 S_d 来表示特征点, $S_d = S_e \cup S_f$, 其中 S_e 是端点, S_f 是分叉点。 $S_f = \{p \mid A(p) \geq 3, B(p) \geq 4\}$, $A(p)$ 是交叉数,是该像素周围8个像素中按 p_1, p_2, \dots, p_8 的顺序中0~1的数量, $B(p)$ 是像素点 p 八近邻中黑色像素点的个数, $B(p) =$

$\sum_{i=1}^8 p_i$, 其中 $p_i (i = 1, 2, \dots, 8)$ 是 p 点的八近邻。

2.2.2 最大圆算法描述

最大圆算法是以原始笔画内的最大内切圆为范围,将圆内距离近的分叉点聚类为一个点,从而将多个分叉点合并为最终的特征点。下面给出该算法的描述:

- 1) 以特征点为圆心,初始半径 $R = 0$;
- 2) 圆心不变,半径值 $R = R + 1$,画圆;
- 3) 判断圆内像素点是否都在笔画范围内,由于笔画中的像素点值为1,所以只需检测圆内所有像素点的值是否全部为1,“是”则执行第二步,“否”则认为已经超出笔画范围,取 $R = R - 1$ 为最大圆半径;
- 4) 进行特征点聚类合并。计算每一对特征点之间的距离,对每一对特征点 f_1, f_2 ,若 $Distance(f_1, f_2) \leq R_1 + R_2$,则这两个点聚为一类。随后以每一类的聚类中心点来代替这类中所有的点。

最大圆方法的结果如图5所示。分叉点处细化的放大如图6所示,可以看出交叉点处的细化结果有了很大改善。

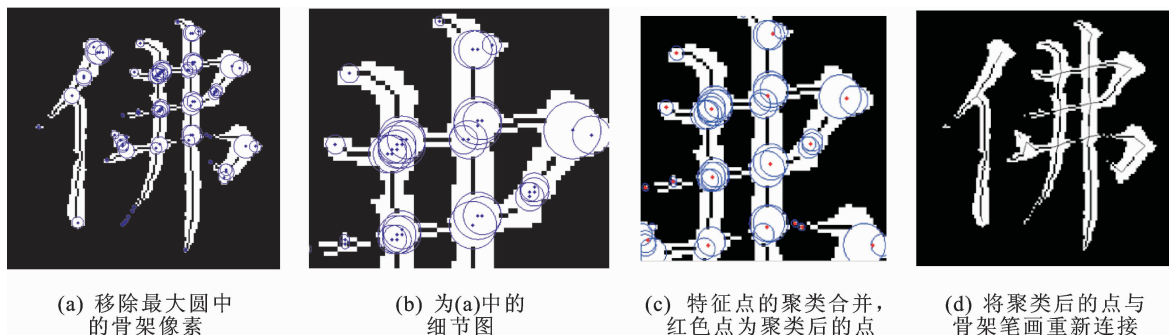


图5 分叉点合并
Fig.5 Fork points merging

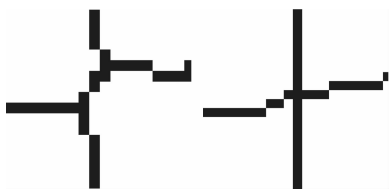


图6 交叉点处细化结果放大图对比
Fig.6 Amplified result of thinning at fork point

3 实验及比较

本文总共进行了3部分实验,分别为书法字的各算法细化结果比较、若干标准字库上的结果比较以及算法性能分析实验。现分别描述如下。

实验1 书法字体的各算法细化结果比较实验

以《多宝塔碑》中的佛字为例进行实验,该字纵横及交叉笔划具有一定的代表性。将本文方法与Z-S方法、最大内切圆方法、击中击不中方法、A-W方法进行对比,细化结果如图7所示。可以看出本

文算法去除了毛刺、分叉等缺陷,优于其他算法。

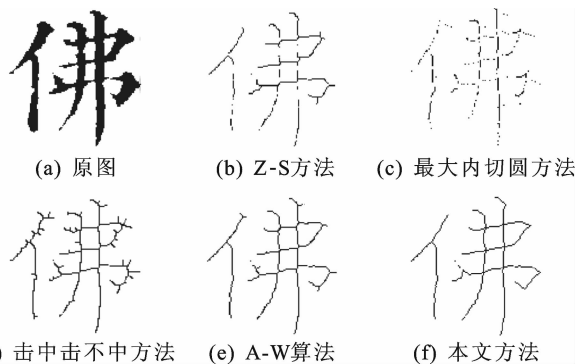


图7 几种骨架提取算法结果的比较
Fig.7 Comparison of skeleton extraction of several algorithms

实验2 若干标准字库上的细化结果比较实验
为了验证算法的结果,在楷体、宋体、黑体、隶书等标准字库上运行了实验。每种字库均采用 96×96 的

模板,各字体抽取 1 000 个汉字进行细化。部分细化对照结果见图 8 所示。图中每一个字的三种细化效果从左至右依次为 Z-S 方法、A-W 方法及本文方法。



图 8 Z-S 方法、A-W 方法及本文方法的效果

Fig. 8 Thinning results of the use of Z-S,

A-W and our method

可以看出,在常见三种字体上,本文方法均取得了优于 Z-S 方法和 A-W 方法的效果。

实验 3 算法的性能比较实验

骨架像素点的个数和端点个数也是用来衡量细化算法优劣的一种指标。很多细化算法都会产生非单像素宽的线和毛刺,多种细化算法得到的骨架像素的个数和端点个数可以用来表征算法性能。仍以实验 1 中的佛字为例,所得结果见表 1。

表 1 几种细化算法性能比较

Tab. 1 Performance comparison of several thinning algorithms

算法	骨架像素个数	端点个数
Z-S 方法	500	35
最大内切圆方法	337	79
击中击不中方法	690(去毛刺后)	47
A-W 方法	581	23
本文中的方法	554	22

从表 1 可以看出,本文方法虽然骨架像素个数多于 Z-S 方法、最大内切圆方法,但主要是因为这两种方法存在断裂和不连通。本文方法的骨架像素数少于击中击不中方法、A-W 方法,因为这两种方法存在非单像素宽的线条且存在毛刺。本文端点个数最少证明连通性好以及毛刺少。这说明骨架多处断裂、连通性差会导致骨架像素个数过少及端点个数多,而非单像素宽的线过多将导致骨架像素个数多。

4 结 语

本文研究书法字的骨架提取算法,针对常见算法出现的锯齿、非单像素、分叉点畸变和毛刺等问题,主要从单像素化和分叉点检测及合并两个方面进行了改进。所引入的单像素化是在保证图像连通

的条件下进行的,不会出现骨架断裂现象。接着应用最大圆方法进行分叉点合并。实验结果表明,本文取得的骨架无毛刺和分叉点,优于其它算法。

参考文献:

- [1] ZHANG T Y, SUEN C Y. A fast parallel algorithm for thinning digital patterns[J]. Communications of ACM, 1984, 27(3): 236-239.
- [2] GONZALEZ R C, WOODS R E. Digital image processing [M]. 2nd ed. Lebanon: Prentice Hall, 2002: 420-453.
- [3] LIAO C W, HUANG J S. Stroke segmentation by Bernstein-Bezier curve fitting[J]. Pattern Recognition, 1990, 23(5): 475-484.
- [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.
LIU Xiabi, JIA Yunde. A character image line-segment extraction and thinning algorithm[J]. Journal of Image and Graphics, 2005, 10(1): 48-53.
- [5] 刘宏申. 击中击不中变换在笔画细化中的应用[J]. 安徽工业大学学报(自然科学版), 2002, 19(3): 251-253.
LIU Hongshen. Application of hit-miss transformation in thinning chokes of character[J]. Journal of Anhui University of Technology, 2002, 19(3): 251-253.
- [6] 俞凯,吴江琴,庄越挺. 基于骨架相似性的书法字检索[J]. 计算机辅助设计与图形学学报, 2009, 21(6): 746-751.
YU Kai, WU Jiangqin, ZHUANG Yueting. Calligraphy characters retrieval based on skeleton similarity [J]. Journal of Computer-Aided Design & Computer Graphics, 2009, 21(6): 746-751.
- [7] 邓茜芸,周成全,武仲科,等. 面向瓦当文字识别的改进水平集骨架提取[J]. 中国图象图形学报, 2014, 19(9): 1324-1331.
DENG Qianyun, ZHOU Mingquan, WU Zhongke, et al. Skeleton extraction using improved level set methods in eaves' text recognition [J]. Journal of Image and Graphics, 2014, 19(9): 1324-1331.
- [8] AHMED M, WAED R. A rotation invariant rule-based thinning algorithm for character recognition[J]. IEEE Transaction on Pattern Analysis and Machine Intelligence, 2002, 24(12): 1672-1678.
- [9] ROCKETT P I. An improved rotation-invariant thinning algorithm[J]. IEEE Transaction on Pattern Analysis and Machine Intelligence, 2005, 27(10): 1671-1674.
- [10] LIU K, HUANG Y S, Suen C Y. Identification of fork points on the skeletons of handwritten Chinese characters[J]. IEEE Transaction on Pattern Analysis and Machine Intelligence, 1999, 21(10): 1095-1100.

(责任编辑 杨小丽)