

指纹图像细化算法的改进算法

王 蒙

(厦门大学计算机科学系 福建 厦门 361005)

摘要: 对指纹图像算法进行了较深入的研究, 分析了 OPTA 算法和改进的 OPTA 算法的优缺点, 提出了一种新的细化算法, 经过实验证明, 该算法能够很好的满足细化的要求, 细化完全彻底, 细化后保持了纹线原有的拓扑结构和细化特征, 而且光滑无毛刺, 运算速度也很快。

关键词: 指纹; 图像细化; 模板

Improved Thinning Algorithm of Fingerprint Image Thinning Algorithm

Wang Meng

(Department of Computer Science, Xiamen University, Xiamen, 361005)

Abstract: The thinning algorithm for fingerprint image is studied deeply in this article. Moreover, Both the advantages and disadvantages of OPTA and the improved OPTA thinning algorithm are analyzed., and a new improved algorithm is presented. Through experiments, the new algorithm has been proved to have many advantages such as complete thinning, quick speed, smooth skeleton, maintenance of minutiae, an so on.

Key words: fingerprint, image thinning, template

一个完整的自动指纹识别系统(AFIS)主要包括指纹采集、指纹图像预处理、指纹特征提取、指纹对比几个模块, 指纹图像预处理是一个重要的部分, 它的处理效果直接影响后续的特征和指纹对比, 而指纹细化又是预处理中的一个重要环节, 因为一般的特征提取都是在细化的基础上进行的, 如果细化不好, 将无法使用常规的特征提取算法提取细节特征信息。

细化处理是指在指纹图像二值化以后, 在不影响纹线连通性的基础上, 删除纹线的边缘像素, 直到纹线为单像素宽为止。理想细化后的纹线骨架应该是原始纹线的中间位置, 并保持纹线的连续性、拓扑结构和细节特征。一种好的细化算法应该满足下列条件^[2]:

- (1) 收敛性: 迭代必须是收敛的。
- (2) 连接性: 不破坏纹线的连接性。
- (3) 拓扑性: 不引起纹线的逐步吞食, 保持原图像的基本结构特性。
- (4) 保持性: 保持纹线的细节特征。
- (5) 细化性: 骨架纹线的宽度为 1 个像素, 即单像素宽。
- (6) 中轴性: 骨架尽可能接近纹线中心线。
- (7) 快速性: 算法简单, 速度快。

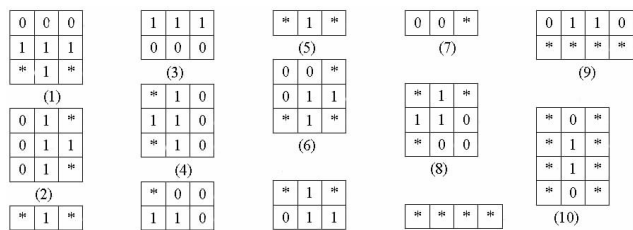
细化算法的种类很多, 按照细化顺序来看主要分为 3 类: 串行、并行和混合细化。改进的 OPTA(含 OPTA)是目前使用较多的一种细化算法。它们是串行细化算法, 其原理构造一定的消除模板和保留模板, 将二值化后的指纹图像和模板比较, 决定是否删除某点的像素值。这种算法能够基本保证单像素宽, 但细化后会产生很多毛刺, 而且实验发现, 经细化后的分叉点处不是单像素宽, 如果不加处理, 这些不足会在特征提取的时候, 导致相当多的伪特征点出现, 极大的影响指纹识别的准确性, 因此提出一种效果好的指纹图像细化算法就显得很重要。

1. OPTA 算法简介^[1]

该算法使用了 10 个基本模板, 如图 1 所示, 其中(1)~(8)为删除模板, 用来删除边界元素, (9)~(10)为保留模板, 用来防止过度删除。其中“1”代表黑点, “0”代表白点, “*”代表“0”或“1”, 以下相同

OPTA 的算法步骤是:

- (1) 删除图像中满足(1)~(8)之一, 但不满足(9)或(10)的像素点;
- (2) 反复执行上一步, 直到图像不再发生变化。



图一

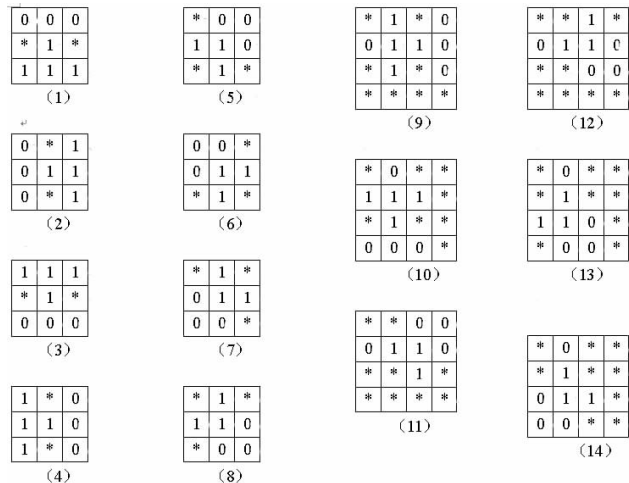
细化效果图见图六(2)

2. 改进的 OPTA 算法简介^[2]

该算法使用了 14 个基本模板, 如图 2 所示, 其中(1)~(8)为删除模板, 用来删除边界元素, (9)~(14)为保留模板, 用来防止过度删除。

OPTA 的算法步骤是:

- (1) 删除图像中满足(1)~(8)之一, 但不满足(9)~(14)的像素点;
- (2) 反复执行上一步, 直到图像不再发生变化。



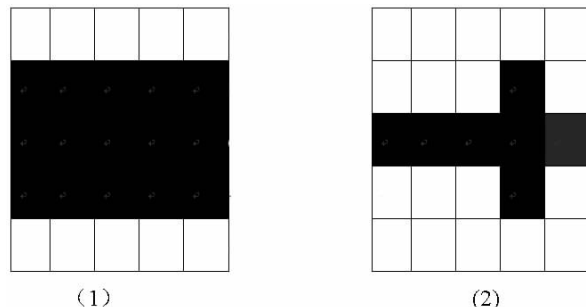
图二

细化效果图见图六(3)

3. 两种算法不足分析及改进策略

可以看出, 改进的 OPTA 算法毛刺减少了很多, 经研究发现, 毛刺的出现对纹线方向十分敏感, 见图()。纹线方向角在第二象限的时候容易出现毛刺, 特别是近似水平和垂直的时候, 毛刺的出现尤其明显, 且以垂直向上、水平向左和水平向右的毛刺居多, 故推测毛刺的产生和消除模板的不完全对称有关。

为了说明问题, 举两幅图来说明。



图三

图(1)是原始的二值化图像,(2)是经典的 OPTA(结果含浅色块)和改进的 OPTA(结果不含浅颜色)算法运算的结果,对第一行第三列因符合保留模板故保留,以下同。对第二行第四列在第一扫描时也符合改进后的 OPTA 算法保留模板 13,但在第二此扫描时,因符合删除模板 4,不再符合任何保留模板,故最后得以删除。

根据以上分析,新算法要对保留模板进行一定的限制,应该在保留模板中剔除下面的情形,见图四,并要进行相应的旋转,构建其它方向的模板。

4.新的细化算法

在新的细化算法中,根据 OPTA 和改进的 OPTA 算法,结合快速细化算法 [2](Quick thinning algorithm)的思想,提出一套新的模板,如图 3 示。其中(1)-(8)删除模板,(9)-(14)是保留模板,(15)-(18)是去除保留模板去除的 4 种情况。

0	0	0	0
0	1	1	0
1	1	1	*
*	*	*	*

图四

算法步骤是:

(1) 删除图像中满足(1)-(8)之一,但不满足去除(15)-(18)的 4 种情况的(9)-(14)的像素点;

(2) 反复执行上一步,直到图像不再发生变化。

<table border="1"><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>* 1 *</td></tr><tr><td>* 1 *</td></tr></table> (1)	0	0	0	* 1 *	* 1 *	<table border="1"><tr><td>0</td><td>*</td><td>*</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>*</td><td>*</td></tr></table> (2)	0	*	*	0	1	1	0	*	*	<table border="1"><tr><td>*</td><td>1</td><td>*</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table> (3)	*	1	*	0	1	1	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>* 1 *</td></tr><tr><td>* 1 *</td></tr></table> (4)	0	0	0	* 1 *	* 1 *
0	0	0																													
* 1 *																															
* 1 *																															
0	*	*																													
0	1	1																													
0	*	*																													
*	1	*																													
0	1	1																													
0	0	0																													
0	0	0																													
* 1 *																															
* 1 *																															
以上 4 个模板中的 "*" 不全为 "0"																															
<table border="1"><tr><td>* 0 0</td></tr><tr><td>1 1 0</td></tr><tr><td>* 1 *</td></tr></table> (5)	* 0 0	1 1 0	* 1 *	<table border="1"><tr><td>0 0 *</td></tr><tr><td>0 1 1</td></tr><tr><td>* 1 *</td></tr></table> (6)	0 0 *	0 1 1	* 1 *	<table border="1"><tr><td>* 1 *</td></tr><tr><td>0 1 1</td></tr><tr><td>0 0 *</td></tr></table> (7)	* 1 *	0 1 1	0 0 *	<table border="1"><tr><td>* 1 *</td></tr><tr><td>1 1 0</td></tr><tr><td>* 0 0</td></tr></table> (8)	* 1 *	1 1 0	* 0 0																
* 0 0																															
1 1 0																															
* 1 *																															
0 0 *																															
0 1 1																															
* 1 *																															
* 1 *																															
0 1 1																															
0 0 *																															
* 1 *																															
1 1 0																															
* 0 0																															
以上为删除模板																															
<table border="1"><tr><td>* 1 * 0</td></tr><tr><td>0 1 1 0</td></tr><tr><td>* 1 * 0</td></tr><tr><td>* * * *</td></tr></table> (9)	* 1 * 0	0 1 1 0	* 1 * 0	* * * *	<table border="1"><tr><td>* * 0 0</td></tr><tr><td>0 1 1 0</td></tr><tr><td>* * 1 *</td></tr><tr><td>* * * *</td></tr></table> (11)	* * 0 0	0 1 1 0	* * 1 *	* * * *	<table border="1"><tr><td>* * 1 *</td></tr><tr><td>0 1 1 0</td></tr><tr><td>* * 0 0*</td></tr><tr><td>* * * *</td></tr></table> (13)	* * 1 *	0 1 1 0	* * 0 0*	* * * *																	
* 1 * 0																															
0 1 1 0																															
* 1 * 0																															
* * * *																															
* * 0 0																															
0 1 1 0																															
* * 1 *																															
* * * *																															
* * 1 *																															
0 1 1 0																															
* * 0 0*																															
* * * *																															
<table border="1"><tr><td>* 0 * *</td></tr><tr><td>1 1 1 *</td></tr><tr><td>* 1 * *</td></tr><tr><td>0 0 0 *</td></tr></table> (10)	* 0 * *	1 1 1 *	* 1 * *	0 0 0 *	<table border="1"><tr><td>* 0 * *</td></tr><tr><td>* 1 * *</td></tr><tr><td>1 1 0 *</td></tr><tr><td>* 0 0 *</td></tr></table> (12)	* 0 * *	* 1 * *	1 1 0 *	* 0 0 *	<table border="1"><tr><td>* 0 * *</td></tr><tr><td>* 1 * *</td></tr><tr><td>0 1 1 *</td></tr><tr><td>0 0 * *</td></tr></table> (14)	* 0 * *	* 1 * *	0 1 1 *	0 0 * *																	
* 0 * *																															
1 1 1 *																															
* 1 * *																															
0 0 0 *																															
* 0 * *																															
* 1 * *																															
1 1 0 *																															
* 0 0 *																															
* 0 * *																															
* 1 * *																															
0 1 1 *																															
0 0 * *																															
以上为保留模板																															
<table border="1"><tr><td>0 0 0 0</td></tr><tr><td>0 1 1 0</td></tr><tr><td>1 1 1 *</td></tr><tr><td>* * * *</td></tr></table> (15)	0 0 0 0	0 1 1 0	1 1 1 *	* * * *	<table border="1"><tr><td>1 1 1 *</td></tr><tr><td>0 1 1 0</td></tr><tr><td>0 0 0 0</td></tr><tr><td>* * * *</td></tr></table> (16)	1 1 1 *	0 1 1 0	0 0 0 0	* * * *	<table border="1"><tr><td>1 0 0</td></tr><tr><td>1 1 0</td></tr><tr><td>1 1 0</td></tr><tr><td>* 0 0</td></tr></table> (17)	1 0 0	1 1 0	1 1 0	* 0 0	<table border="1"><tr><td>0 0 1 *</td></tr><tr><td>0 1 1 *</td></tr><tr><td>0 1 1 *</td></tr><tr><td>0 0 * *</td></tr></table> (18)	0 0 1 *	0 1 1 *	0 1 1 *	0 0 * *												
0 0 0 0																															
0 1 1 0																															
1 1 1 *																															
* * * *																															
1 1 1 *																															
0 1 1 0																															
0 0 0 0																															
* * * *																															
1 0 0																															
1 1 0																															
1 1 0																															
* 0 0																															
0 0 1 *																															
0 1 1 *																															
0 1 1 *																															
0 0 * *																															
以上是保留模板要去掉的情形																															

图五

OPTA 以及改进的 OPTA 算法和新的细化算法的效果图如图六所示。

示。



图六

从效果图中可以看出,OPTA 算法在线段粗细均匀的部分,细化效果是比较理想,但对毛刺较敏感,线段不均匀时效果不太好。改进的 OPTA 算法细化较彻底,但在纹线的水平和垂直方向会产生较多毛刺,在分叉点处同样存在不是单像素宽度的现象。采用新的细化算法之后,线条平滑,图像规整,连通性、细化性较为理想,为下一步的处理打下了一个良好基础。

5.结论

本文的实验是在 Windows Xp +.NET 的平台下,使用 C# 语言编程实现。利用本文所提供的模板组对指纹图像进行细化,实验结果表明该方法能比较好地保持图像原有的骨架结构,纹线的细节特征没有丢失,没有毛刺的出现。因此该算法是一种较好的,并且实用的指纹图像细化算法。

参考文献

[1] 李四明. 工程图纸输入与自动识别的改进细化算法[J]. 计算机工程. 2003, 29(16): 37-38.
 [2] 王业琳, 宁新宝, 尹义龙. 指纹图像细化算法的研究[J]. 南京大学学报(自然科学). 2003, 39(4): 468-475.
 [3] 冈萨雷斯. 数字图像处理[M]. 阮秋琦, 阮宇智, 译. 北京. 电子工业出版社. 2003: 438-441.
 [4] 夏良正. 数字图像处理[M]. 南京. 东南大学出版社. 1999: 255-258.

作者简介: 王蒙, 男(1972-), 河南鹤壁人, 郑州市职教教研室教研员, 在读研究生, 主要从事数字图像处理。

(上接第 353 页) 八、一手抓成本, 一手抓资金

困扰施工企业的有两个问题, 一是任务, 二是资金。造成施工企业资金紧张的原因是多方面的, 客观因素很多, 有标价低(收入减少)、各种保函、质何金(资金抵押)、完工工程价款结算滞后(通常达二-三个月以上)、建设单位拖欠工程款等。对资金问题稍加放松, 就影响施工, 一味从银行贷款, 就会加大财务费用支出。解决这个问题, 应从两方面采取措施, 一方面, 该要的钱, 特别是建设单位的工程款, 要及时要。另一方面, 应付出去的钱, 严格按照规定支付, 不到万不得已, 不得银行借款。

九、项目主管会计(财务负责人)实行委派制

单位成本管理的好坏, 项目成本管理是基础、是关键。管理项目成本, 固然要项目部全体人员按照成本分解的责任目标认真管好, 但是

项目主管会计有着不可低估的作用。成本的分解、计划、控制、核算、分析、报表、考核等, 有的要会计人员参与, 有的要会计人员完成。项目主管会计如果受项目经理领导和聘用, 不执行财务制度, 不及时记帐、报帐、造假帐、编假表的现象时有发生, 为改变这种状况, 对项目主管会计实行由公司委派。委派的主管会计, 一方面对派出单位负责, 完成派出单位交待的各项任务, 包括执行财务制度, 保证会计数字的真实性等。另一方面, 又要为派驻单位服务, 及时记帐、算帐, 提供会计信息, 参与和做好成本管理的各项工作。派出单位对主管会计的各项工作, 明确考核标准, 每年由财务部门和人事部门共同考核, 按考核分数给予奖惩。

施工成本管理是全企业的事, 涉及到各项管理工作, 按照以上思路去做, 是会取得一定效果的。