

一种方向链码扫描与跟踪的图像细化后期处理算法

瞿 中^{1,2} 蒋玉萍¹ 文倩云¹

(重庆邮电大学计算机科学与技术学院 重庆 400065)¹ (重庆邮电大学移通学院 重庆 401520)²

摘 要 目标图像骨架的提取是智能分析中的重要组成部分,利用 Zhang 并行细化算法提取的目标骨架不是单一像素且极易产生毛刺。提出一种获取单一像素并消除毛刺的快速目标图像骨架提取算法。该算法首先对提取得到的目标二值图像进行形态学预处理,然后结合 8 邻域方向链码扫描编码原理对细化后的图像进行单一像素处理,最后采用优化的 8 邻域方向链码来消除毛刺。实验结果表明,提出的算法不仅效率高,而且能够很好地获得单一像素宽度、无毛刺的骨架。

关键词 骨架,单一像素宽度,毛刺,方向链码

中图法分类号 TP391.41 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.2.062

Algorithm of Image Thinning Post-processing Based on Direction Chain Code Scanning and Tracking

QU Zhong^{1,2} JIANG Yu-ping¹ WEN Qian-yun¹

(College of Computer Science & Technology, Chongqing University of Posts & Telecommunications, Chongqing 400065, China)¹

(College of Mobile Telecommunications, Chongqing University of Posts & Telecommunications, Chongqing 401520, China)²

Abstract The extraction of the target image skeleton is an important part of the intelligent analysis. Some flaws of Zhang parallel thinning algorithm are that the skeleton is non-single-pixel and also easily produces burr. This article proposed a fast image skeleton extraction algorithm to obtain a single pixel and eliminate burr. Firstly, the binary target image needs to be morphologically preprocessed to fill the tiny holes and smooth the boundary. Secondly, this article used 8 direction chain code scanning and coding principle to achieve a single pixel. Finally, this article adopted the 8 direction chain code to remove the burr. The experiments show that the algorithm can rapidly and effectively obtain a single pixel width and remove the burrs.

Keywords Skeleton, Single pixel width, Burr, Direction chain code

1 引言

裂缝自动分析是实现智能交通维护的重要参数计算方法,作为裂缝自动分析^[1]的重要内容,裂缝骨架提取在裂缝的分类和裂缝长度^[2]、宽度特征参数的计算中都有重要的应用。

近年来,研究提出了几种新的细化算法^[3,4]。这些算法针对性强,但不适合裂缝图像的骨架的提取。当前一些经典细化算法,例如基于模板^[5]、索引表^[6]的形态学细化算法和 Zhang 并行快速细化算法^[7]都有被应用于裂缝骨架的提取,通过形态学腐蚀和开操作得到的骨架会产生较多的毛边,有损裂缝骨架的连通性。Zhang 并行细化算法设计两种子迭代模板,通过多次迭代实现裂缝骨架的获取,它具有速度快、保持细化后的骨架曲线为中心线和连通性等优点。但是裂缝骨架的单一像素宽度仍无法满足应用,同时细化后骨架还存在毛刺,这为后续裂缝交叉点检测和裂缝特征参数计算带来误差。现有的图像细化后期处理算法基于模板来获取单一像素宽度和消除毛刺^[8],在单一像素化时会产生断裂,处理复杂,同时消除毛刺的算法^[9]适应性不强。

本文针对裂缝二值图进行裂缝骨架提取,提出一种结合方向链码和 Zhang 并行细化算法的优化裂缝骨架提取算法,

该算法主要分为预处理、裂缝细化、单一像素化和毛刺消除 4 个模块。预处理阶段利用形态学的膨胀腐蚀完成裂缝中孔洞的消除和边缘的圆滑;裂缝细化采用 Zhang 快速并行细化算法获得具有很好连通性的裂缝骨架图;骨架的单一像素化引用 8 邻域 Freeman 链码扫描的思想,并在此基础上加以改进,很好地保证了骨架 8 邻域单一连接像素点的特征;毛刺消除采用一种优化的基于方向链码的毛刺消除算法^[10]。以骨架的端点作为起点,计算链码长度并判定是否为毛刺。实验结果表明,本文算法不仅效率高,而且能够很好地获得单一像素宽度、无毛刺的裂缝骨架。

2 裂缝细化

Zhang 并行快速细化算法由 Zhang 在 1984 年提出。设定已知图像目标点像素为 1,背景点像素为 0,根据细化的要求设定判定条件来消除非骨架上的点。其基本思想就是层层剥离,即从边界开始一层一层向里剥离,直到最后骨架为止。

记边界点 P_1 中心点。其 8 邻域像素点按照顺时针方向分别记为 $P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9$, 其中 P_2 在 P_1 上方,如图 1 所示。

到稿日期:2014-02-18 返修日期:2014-05-23 本文受重庆市科委自然科学基金计划资助项目(2010BB2399)资助。

瞿 中(1972-),男,博士,教授,主要研究方向为数字图像处理、普适计算、物联网技术等,E-mail:quzhong@cqupt.edu.cn;蒋玉萍(1988-),女,硕士生,主要研究方向为数字图像处理;文倩云(1990-),女,硕士生,主要研究方向为数字图像处理。

P_9	P_2	P_3
P_8	P_1	P_4
P_7	P_6	P_5

图1 点 P_1 的 8 邻域图

具体的算法步骤如下:

子迭代 A: 将同时满足下面 4 个条件的像素点消除, 变为背景点。

(1) P_1 的 8 邻域中, 值为 1 的点的个数大于等于 2 并且小于等于 6;

(2) P_2, P_3, \dots, P_9 的排列顺序中, 01 模式数量为 1;

(3) 点 P_2, P_4, P_6 中至少有一个为 0;

(4) 点 P_4, P_6, P_8 中至少有一个为 0。

子迭代 B: 将同样满足以下 4 个条件的像素点消除变为背景点。

(1) 与子迭代 A 中(1)、(2)条件相同;

(2) P_1, P_2, P_8 中至少有一个为 0;

(3) P_2, P_8, P_6 中至少有一个为 0。

子迭代 A 和 B 构成一次迭代, 对整幅图像进行循环迭代处理, 直到没有满足上述条件的点则终止循环, 剩下的像素点构成目标区域的中心骨架。

图 2(b) 即为上述细化算法作用于图 2(a) 的细化结果, 图 2(c) 为局部区域 Label1 细化放大, 图 2(d) 为局部区域 Label2 细化放大。通过图 2(c) 和图 2(d) 可以清楚地得出, 该算法得到的裂缝骨架不能保证单一像素宽度, 还会存在毛刺。

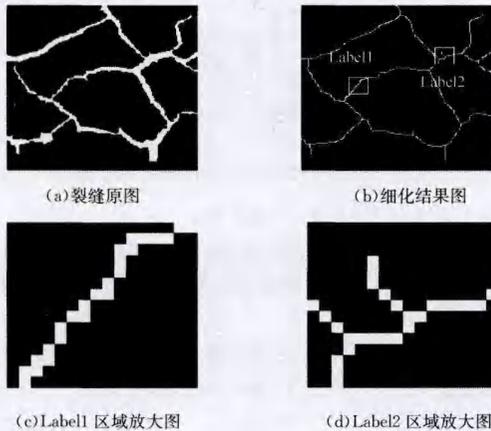


图2 Zhang 细化算法结果

3 细化骨架的单一像素化

理想细化效果下的骨架为单一像素宽度且无毛刺, 相邻骨架点之间不会形成 90° 的夹角。实验结果显示, 经过 Zhang 并行快速细化算法之后获得的骨架 (如图 3 所示) 存在导致 90° 夹角形成的骨架点 P , 称为突起点。裂缝骨架存在 4 个方向的突起点, 它们分布在前一编码方向的两侧, 本文提出利用 8 邻域方向链码编码的思想来消除突起点 P 。

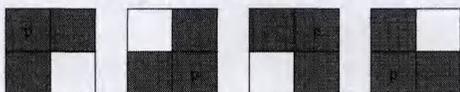


图3 非单一像素宽度点

3.1 8 方向链码编码

8 方向链码的概念由 Freeman 于 1961 年提出, 最初是用来表示图像边界轮廓。如图 4 所示, 考虑像素点的 8 个邻域对应着 8 个连接方向, 这 8 个方向用 0、1、2、3、4、5、6、7 分别来表示。其最优跟踪编码过程如下: 首先扫描确定图像的某一边界点为起点, 初始化定义扫描方向, 直到扫描到下一边界点, 确定其编码方向, 接下来每次扫描寻找的边界点都是在前一编码方向上相应旋转 90° 再反向开始扫描连接点, 如此循环从而完成整个图像边界 8 邻域链码的跟踪和编码。

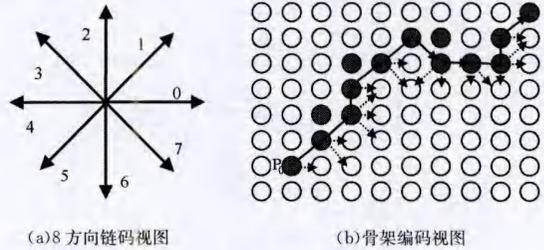


图4 8 方向链码编码示意图

如图 4(b) 所示, 一段经过细化后的骨架分支, 以 P_0 为起点对该分支进行 8 邻域方向链码扫描和跟踪编码, 定义初始编码方向为 0, 再每次逆时针旋转 45° 扫描 8 邻域像素, 确定下一连接的分支点进行跟踪编码, 接下来每次都在前一编码方向上顺时针旋转 90° 开始继续扫描下一连接点。图 4(b) 中实线箭头的指向即为该分支的链码方向, 图中分支的编码为: 1121170021。实线箭头连接的像素点即为经过 8 方向链码编码遍历过的分支点。从图中可以看出, 每次顺时针旋转 90° 后再反向逆时针扫描遍历的分支会消除左上方和右上方的突起点, 同理, 每次先逆时针旋转 90° 后再顺时针扫描遍历分支便会消除左下方和右下方的突起点。

3.2 单一像素骨架提取算法

本文基于 8 方向链码的编码原理, 利用相反方向的两次编码过程来消除细化后所得骨架图像的突起点, 实现细化骨架的单一像素宽度。对细化后图像, 利用广度优先遍历整个图像的目标骨架。

算法的实现步骤如下 (见图 5):

(1) 对细化后得到的 0-255 二值图像 Ske_Image 进行扫描, 利用链表 $cross_p$ 和 $node_p$ 分别存贮其细化后图像的端点和节点;

(2) 对图像中的端点、节点以及节点周围 8 邻域分支点的像素进行标记, 像素值分别置为 1、3 和 8;

(3) 依次以 Ske_Image 图像的节点和端点为扫描起点, 判定是否已扫描该分支, 针对未扫描的每一分支, 确立初始编码方向, 接下来每次扫描初始方向都在前一编码方向先顺时针旋转 90° 再逆时针每一次旋转 45° 扫描该分支的下一个连接点, 利用广度优先遍历所有图像分支, 对分支加以跟踪并标记, 将链码上的图像骨架点标记为 128;

(4) 经过步骤(2)处理后图像 Ske_Image 的像素值有 0、1、3、128 和 255 这 5 种取值, 保留图像骨架所需要的像素为 1、3 和 128 的像素点, 像素值设为 255, 余下的归为背景点, 像素值为 0;

(5) 同步步骤(2)、(3)、(4), 再一次采用广度优先对图像上所有分支进行一次遍历, 分支遍历方向反向, 每次扫描方向都

在前一编码方向上逆时针旋转 90°再顺时针每一次旋转 45°扫描分支连接点,最后同样对遍历后并加以标记的图像进行恢复,得到 0-255 的单一像素骨架宽度的二值图,如图 6 所示。

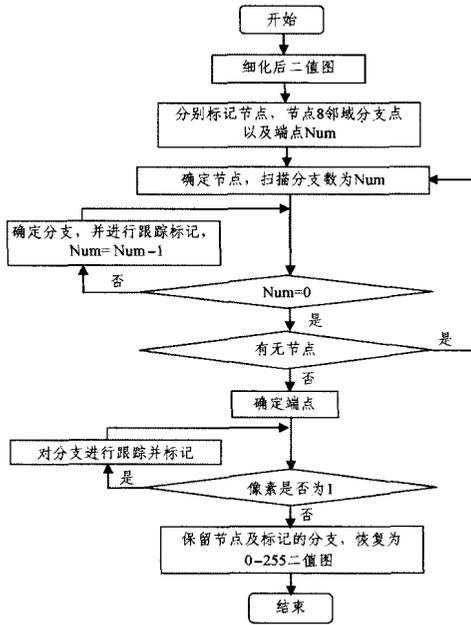


图 5 目标骨架遍历算法流程



(a) 细化后局部放大图 (b) 单一像素化后局部放大图

图 6 方向链码扫描跟踪获得单一像素骨架效果图

如图 6 所示,利用方向链码跟踪扫描分支的方法不仅可以很好地消除细化后图像中存在的突起点,而且对于类似图 6(a)中 4 条分支交点的非单一像素情况,也能很好地消除多余像素,最终实现整幅图像目标骨架的单一像素化。

4 结合方向链码的毛刺消除算法

经过单一像素化后的细化图像,虽然很好地获得了单一像素宽度,但细化后的图像毛刺仍然存在,还得再作处理。本文优化单一像素骨架毛刺消除算法,结合方向链码编码原理很好地消除了多余的毛刺。

4.1 毛刺消除分析

设计毛刺消除算法,首先要理解毛刺的定义,图像中毛刺是指目标图像边缘的一些刺状物和毛边,细化后的图像骨架中一些很短的分支也被定义为毛刺,它是一个感性的定义,没有一个标准和方法来检测,需要根据实际的应用来确定。本文研究的裂缝图像,经过 Zhang 快速并行细化算法处理后得到的图像会存在一些要比骨架整体长度短的很多的分支,这些分支是因为裂缝宽度不一样或者图像边缘存在的一些毛边,在细化后将宽度突变大的部分突显出来而产生的,我们称它为裂缝毛刺。

对于同一形状的目标图像,由于最大宽度不一样,细化迭代的次数也不同,产生的毛刺数量以及长度也会不一样。本文结合细化迭代的次数来设定裂缝骨架毛刺定义的阈值,以像素为单位,这里也称之为步长 L 。

$$L = \text{ceil}(2\sqrt{2} \times \text{times}) \quad (1)$$

times 为目标图像细化迭代次数, L 为判定目标图像毛刺的步长, ceil 表示取大于或等于括号内数的整数。

4.2 优化的细化图像毛刺消除算法

骨架节点为可能的毛刺生长点,利用方向链码从节点出发对分支进行跟踪,将未超过设定阈值的分支判定为毛刺并将其删除。该方法能够将单一像素宽度的骨架图像的毛刺很好地删除,但是对于复杂的网状图像,存在很多内部节点之间连接的分支,这些分支不可能为多余的毛刺,所以本文改进该算法,即利用目标图像骨架的所有端点为起点扫描分支,忽略图像上节点之间连接的分支,以提高实现速率。同时本文改进的细化图像消除毛刺的算法不限定节点的可能情况,适应包含各类分支的节点。其流程如图 7 所示。

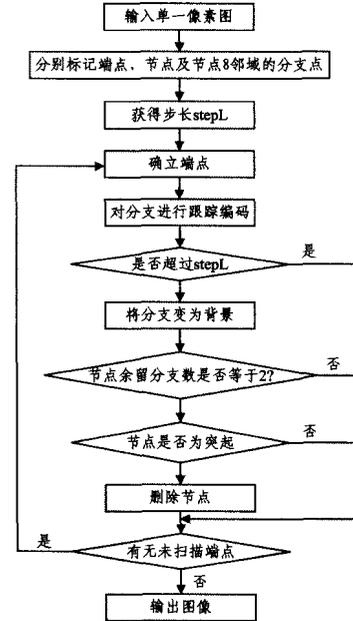


图 7 方向链码编码毛刺消除算法流程

本文改进的毛刺消除算法步骤:

- (1)对单一像素化后的图像进行标记,将节点和节点 8 邻域点以及端点加以标记,像素分别置为 3、8 和 1,存储图像所有端点;
- (2)若存在节点,说明图像可能存在毛刺,则以节点为起点,扫描图像分支进行 8 方向链码编码;
- (3)设定步长 stepL ,当扫描步长大于 stepL 时,退出该分支的扫描,反之在步长内扫描到分支的另一端点,结束扫描;
- (4)若分支的另一端为节点,忽略当前分支,对节点的余留的分支数进行判定,若分支数大于等于 2,则为毛刺;
- (5)若为毛刺,对节点的 8 邻域进行分析,若删除节点不影响余下分支的连通性,则连带节点删除该分支,否则保留节点,删除分支;
- (6)完成所有端点对应分支的跟踪编码,最终实现整幅图像毛刺的消除。

4.3 毛刺消除结果分析

实验结果显示,该毛刺消除算法能够很好地判定并消除裂缝毛刺,如图 8 所示,图 8(a)为 400×300 的裂缝图像,经过细化后如图 8(b)所示,其产生了很多细微的毛刺,图 8(c)显示了经过毛刺消除算法后多余的毛刺被很好地消除了。判定毛刺的步长为细化迭代次数的一个线性相关动态值,可以很

好地消除裂缝的毛刺而不受裂缝的宽度影响。图9为测试数据中的3个不同类别图像A、B和C,大小都为 400×300 。将其放大4倍后得图像A'、B'和C'。通过表1得到的测试数据可以看出,设定的步长可以很好地消除细化算法产生的毛刺,保证细化后图像的骨架与原图架构一致。

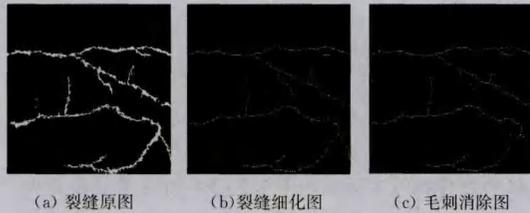


图8 图像毛刺消除效果

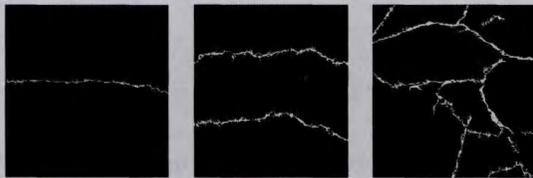


图9 实验测试图像

表1 毛刺消除数据分析

图像	细化迭代次数	步长	分支个数		
			原图	细化后骨架图	消除毛刺后骨架图
A	4	10	0	7	0
A'	11	27	0	22	0
B	5	13	0	9	0
B'	17	42	0	62	0
C	6	15	9	35	9
C'	21	51	9	101	9

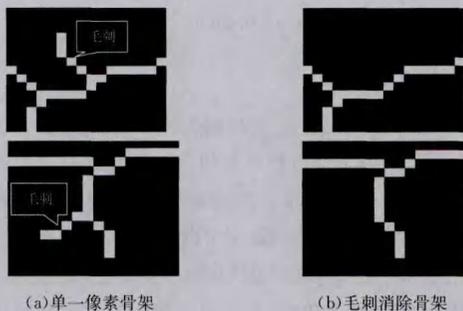


图10 局部毛刺消除效果

如图10所示,该算法在消除毛刺的同时,能够判定消除

遗留节点造成的突起,保证最终骨架的单一像素宽度。另外,在同一节点引生出的两条分支,当分支长度都在毛刺判定的阈值范围内时,本文算法根据起始扫描方向来保留一条分支而不同时消除两条分支,保留骨架的真实长度。

结束语 本文利用8方向链码扫描跟踪原理,实现了细化图像的单一像素化。与现有的利用模板来消除多余像素点实现单一像素宽度相比,本方法能够很好地保持细化图像的连通性,不需要再进行断点的连接,适应性更强。针对细化毛刺的消除,本文提出以可能为毛刺的终端(分支端点)为起点,以方向链码编码来判断分支,确定并消除毛刺,该算法能够忽略两节点之间的分支,优化速度,提高效率。同时该算法不限定毛刺生长点(节点)引出的分支数及其在8邻域分布情况。实验结果表明,利用本文算法对Zhang并行快速细化图像做后期处理,可以有效地实现单一像素宽度且无毛刺的目标骨架图像。

参考文献

- [1] 孙姜燕. 基于图像分析的桥梁下部结构裂缝检测算法研究[D]. 西安:西安电子科技大学,2011
- [2] 马鑫,魏鹏旭,岳康. 裂缝图像识别与特征参数算法的研究[J]. 工程技术,2011,(11):47-48
- [3] Bag Sou-men, Harit G. A Medial Axis Based Thinning Strategy And Structural Feature Extraction of Character Images[C]// Proceedings of 2010 IEEE 17th International Conference on Image Processing. 2010:26-19
- [4] Padole G V, Pokle S B. New Iterative Algorithms For Thinning Binary Images[C]// Third International Conference on Emerging Trends in Engineering and Technology. 2010:166-171
- [5] 陈艳君. 基于特征空间的路面裂缝检测与识别算法研究[D]. 武汉:武汉工程大学,2012
- [6] 王龙云. 路面裂缝检测算法研究[D]. 南京:南京邮电大学,2012
- [7] Zhang T Y, Suen C Y. A Fast Parallel Algorithm for Thinning Digital Patterns[J]. Image Processing and Computer Vision, 1984,27(3):236-239
- [8] 宁亚辉,雷小奇,王功孝,等. 改进的基于模板消除骨架毛刺的方法[J]. 计算机应用,2011,31(1):58-63
- [9] 郭斯羽,董红霞,张翌. 一种用于植物叶片图像骨架提取的去毛刺方法[J]. 电子测量与仪器学报,2013,27(1):52-56
- [10] 王要峰,崔艳. 基于方向链码消除骨架图像毛刺算法[J]. 计算机应用,2013,33(1):193-194
- [11] 祝强,徐臻. 采用小波构造的图像阈值去噪算法[J]. 重庆理工大学学报,2013,27(6):61-67

(上接第259页)

- [11] Nguyen, Hien V. Sparse embedding: A framework for sparsity promoting dimensionality reduction[J]. Lecture Notes in Computer Science, 2012,7577(6):414-427
- [12] 冷亦琴,张莉,杨季文. 一种基于局部稀疏线性嵌入的降维方法及其应用[J]. 南京大学学报,2013,49(4):403-410
- [13] Sun Yang, Ye Qing-wei, Wang Xiao-dong, et al. Improved LLE

Algorithm Based on Sparse Constraint[J]. Computer Engineering, 2013,39(5):53-56,60

- [14] 陈才扣,喻以明,史俊. 一种快速的基于稀疏表示分类器[J]. 南京大学学报,2012,48(1):71-76
- [15] Pan Y, Ge S S, Mamun A A. Weighted Locally Linear Embedding for Dimension Reduction[J]. Pattern Recognition, 2009,42(5):798-811