

一种改进的二值图像边界跟踪与边界链码获取算法

邓仕超, 李伟明, 龙芋宏, 高兴宇*

桂林电子科技大学广西制造系统与先进制造技术重点实验室, 广西 桂林 541004

摘要 为了克服传统二值图像边界跟踪速度慢和边界漏跟踪等问题, 基于 Freeman 链码提出一种通用性强的多起点分段二值图像边界跟踪与边界链码获取算法。在借鉴前人对传统 Freeman 链码搜索算法的改进, 将平均搜索方向减少为 2.5 就可以找到新边界点的基础上, 进一步将复杂的边界分段处理, 令前一边界链的起点作为待跟踪边界链搜索的起点, 同时把跟踪后的边界点填充, 这样自上而下自左而右的对图像扫描, 只需一次即可跟踪完所有的边界, 并通过设定阈值消除了单像素冗余点。实验论证结果表明, 本文算法能够对复杂网状边界、内部孔洞进行有效地跟踪, 同时具有抗噪去除冗余点的能力和较快的速度。

关键词 图像处理; 边界跟踪; 分段; 二值图像; 链码

中图分类号 TP391.4

文献标识码 A

doi: 10.3788/LOP55.061007

Improved Algorithm for Boundary Tracking and Boundary Chain Code Acquisition of Binary Image

Deng Shichao, Li Weiming, Long Yuhong, Gao Xingyu

Guangxi Key Laboratory of Manufacturing Systems and Advanced Manufacturing Technology,

Guilin University of Electronic Technology, Guilin, Guangxi 541004, China

Abstract In order to overcome the shortcomings of the traditional binary image boundary tracking, including slow speed and leakage tracking, an algorithm of binary image boundary tracking and boundary chain code acquisition with strong generality is proposed based on the Freeman chain code. Based on the improvement of traditional Freeman chain code search algorithm, the average search direction is reduced to 2.5 when a new boundary point is found. Further the complex boundary segment is processed, and the starting point of the previous boundary chain is used as the starting point for the following boundary chain search, and fills the following boundary points, so that the image scanning from top to down and left to right is only one time to track all the boundaries, and eliminates the single pixel redundancy by setting the thresholds. The experimental results show that the proposed algorithm can effectively track the complex mesh boundary and internal holes, and it has the ability to remove the redundant points with anti-noise and faster speed.

Key words image processing; boundary tracking; segmentation; binary image; chain code

OCIS codes 100.2000; 100.3008; 100.2960; 100.2960

1 引言

边界跟踪与链码获取是图像处理中一项重要技术, 在目标分析、图像压缩、图像分割以及计算机图像学中广泛应用。边界跟踪是从一个起始点开始, 按照四连通或八连通关系, 按同一方向沿边界行走,

直到到达某一点为止的过程^[1]。通过边界跟踪, 可以获得区域边界点的坐标序列和链码表示。分析数字图像中物体的轮廓是现代数字图像处理的重要手段, 属于图像分割技术的范畴^[2]。

在边界跟踪算法中, 确定下一个边界点的跟踪方向和寻找不同边界链的起始点是该算法的关键,

收稿日期: 2017-11-14; 收到修改稿日期: 2017-12-11

基金项目: 广西自然科学基金(2015jjBA70017)

作者简介: 邓仕超(1973—), 男, 博士, 副研究员, 硕士生导师, 主要从事图像处理和图像识别方面的研究。

E-mail: dsc@guet.edu.cn

* 通信联系人。E-mail: gxy1981@guet.edu.cn

经典的算法有“虫随法”“光栅法”等。近年来, 研究者们提出了多种改进算法, 但重复跟踪、漏跟踪、速度慢、死循环^[3]以及无法克服噪声点冗余点干扰^[4-5]等问题依然存在, 这些问题在存在岛、内部孔洞和边界多交叉的图像表现中尤为突出^[6]。崔凤魁等^[7]提出目标邻域点的边界点搜索方法, 该算法从上一边界点的下一邻域点开始搜索, 将搜索方向减至 7 个, 但仍有大量搜索冗余没有被去除。王珏等^[8]提出一种带记忆的变窗口的边界跟踪算法, 即先用 Roberts 算子对原始灰度图像进行边界提取, 然后通过人工选择一个或多个起始种子点对边界进行跟踪。此算法最大的弊端是需要人工选择一个或多个起始种子点, 容易造成漏跟踪。王福生等^[9]将搜索方向减至最多 5 次, 比崔凤魁等^[7]提出的目标邻域点的边界点搜索算法的搜索冗余少, 但仍有部分冗余未去除; 此外, 该算法还存在重复跟踪、目标区域填充效率低和内部孔洞边界漏跟踪等问题。周丰乐等^[10]提出的动态权值跟踪算法与陈旺等^[6]提出的一种四邻域边界跟踪算法都只能跟踪闭合边界, 对于非闭合边界没有给出解决方法。李贞培等^[11]提出的分段算法仅基于 GIS 栅格地图, GIS 栅格地图的轮廓为不同的单一颜色, 边界搜索时只需搜索单一颜色, 从而限制了该算法的应用范围。周秀芝等^[3]提出基于交叉点树的边界跟踪算法, 即首先利用八邻域寻找到交叉点和边界, 再把交叉点与其他边界段连起来, 该算法可以很好地跟踪复杂交叉的边界, 但是算法复杂、效率低。Kuagoolkijgarn 等^[12]提出一种新的互连轮廓跟踪算法, 即应用形态学和匹配的方法把不同的轮廓分离出来分别跟踪, 该算法对于互连轮廓跟踪效果良好, 但算法效率低、且未考虑噪声点和冗余点。

针对一直以来存在的这些问题, 本文提出一种改进算法, 既能兼顾边界跟踪速度, 又能解决重复跟踪和传统“虫随法”的死循环问题, 同时对孔洞、多交叉的复杂网络边界有很好的跟踪效果, 并具有抗噪去除冗余点的能力。

2 边界跟踪与边界链码创建改进

基本原理

2.1 新算法的思路

首先进行二值图像边界提取, 即遍历二值图像的每一个像素, 以 0 值像素为参考点, 搜索参考点十字星(“+”)上下左右 4 邻域方向上的像素, 确定 4 个方向是否存在至少一个像素为 1 的点, 若存在, 则

参考点为边界点, 否则为非边界点。边界提取后进行边界分段跟踪: 1) 从边界图左上角点开始, 自上而下自左而右逐像素点扫描, 当遇到边界点时开始边界跟踪, 边界跟踪中, 根据每一条边界链上前一边界点链码值的奇偶性选择对后一边界点的搜索方向, 每一条边界链跟踪回到起始点(对于封闭轮廓)或末尾点搜索不到新的后续点(对于非封闭或者岔路线段)即完成一个边界链的跟踪; 2) 每一条边界链, 从起始点之后, 即从第二点起, 每搜索完成一次边界点, 将该边界点置为背景色 1, 最后回到起始点或最后一个点, 再把起始点置背景色 1; 3) 通过设定阈值判断噪声或者冗余孤立点, 对噪声或者冗余孤立点直接置背景色 1, 同时删除链码记录; 4) 每下一个边界链起始点的扫描寻找, 都从刚被跟踪完的边界链(即上一个边界链)的起始点开始, 这样可以实现扫描一次就完成所有边界链的搜索和跟踪。

2.2 链码和链码表原理

链码通过给定方向和单位长度的线段(单位长度的线段是指单个像素)来表示轮廓或者边界, 链码的建立是一个轮廓(边界)跟踪的过程^[13]。若当前点 (x, y) 为一个边界点, 如果相邻的位置还存在边界点, 则下一个边界点必定在当前点 (x, y) 的 8 邻域内^[14], 该邻域内 8 像素位置的链码表示和坐标表示如图 1 所示, 其中图 1(a)为链码, 链码值为 0~7, 按逆时针旋转递增; 图 1(b)为链码的坐标表示。图 2 为边界按逆时针旋转的链码表示图。

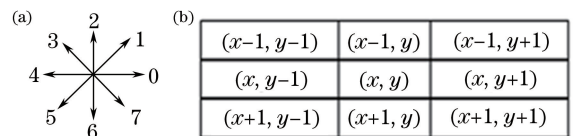


图 1 (a)链码表示; (b)坐标表示

Fig. 1 (a) Chain code representation; (b) coordinate representation

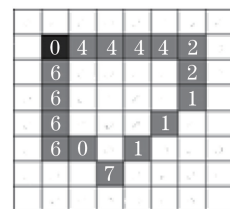


图 2 边界表示

Fig. 2 Border representation

2.3 搜索冗余

通过对二值图像边界研究分析, 如图 3 所示, 假设 A、B 分别表示相邻的两个边界点(A 为前一点,

B 为后一点,另外 6 种情况同理),01234567 表示 A 的搜索范围,0①②③④⑤⑥⑦表示 B 的搜索范围,图中加粗的搜索数字即为重复搜索的冗余位置。如果前一个边界点搜索这些位置没有发现边界点,那么后一边界点重复搜索也不会发现,因此前后已知的两个边界点和将要搜索确认的第 3 个边界像素位置有 8 种规律,如图 4 所示。

3	2③	1②	①			③	②	①	
4	A④	B0	0		3	2④	B1	0	
5	6⑤	7⑥	⑦		4	A⑤	0⑥	⑦	
					5	6	7		

图 3 重复搜索区域

Fig. 3 Repeated search area

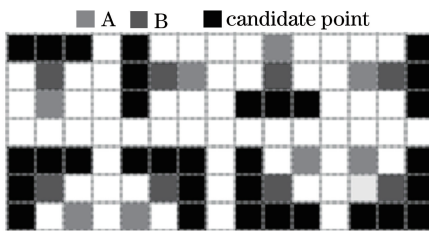


图 4 边界点位置

Fig. 4 Boundary point location

此外,由于对边界起始点的搜索是按自上而下自左到右的顺序,因此起始点的左方、左上方、上方、右上方等 4 个方向不会出现边界点,因此只需搜索起始点的其余 4 邻域即可,如图 5 所示。

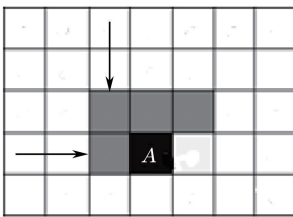


图 5 灰色冗余区域

Fig. 5 Gray as a redundant area

2.4 单个边界链跟踪规则

在链码严格按照逆时针旋转搜索时,通过 2.3 节的冗余分析可得:1) 起始点对后一个边界点的搜索从链码值为 5 的方向开始;2) 起始点之后的边界点,当边界点链码值为偶数时,对下一个边界点的搜索,从当前链码值顺时针旋转 45° (即链码值加 1) 处开始;3) 当边界点链码值为奇数时,对下一个边界点的搜索,从当前链码值顺时针旋转 90° (即链码值加 2) 处开始,如果旋转 45° 或者 90° 后得到的像素

是 0 则为新的边界点,否则按逆时针方向顺序搜索,遇到的第一个 0 值像素即为边界点。本文方法的平均搜索次数为 $[(1+2+3)/3+(1+2+3+4+5)/5]/2=2.5$ 次,而文献[7]算法的平均搜索次数为 $(1+2+3+4+5+6+7)/7=4$ 次,文献[9]算法的平均搜索次数为 $(1+2+3+4+5)/5=3$ 次。

2.5 边界跟踪边界链码创建算法流程

设二值图像中背景像素值为 1,前景像素值为 0; n 为链码值, P_{tk} 为第 t 个边界链编码系的起始点后的第 k 个边界点, t, k 的初值均为 0。首先对二值图像进行边界提取,然后再对边界图进行边界跟踪,算法流程如图 6 所示。本文算法有如下几点需注意:

1) 第一个边界链的起始点为 P_{00} (其他边界链起始点为 P_{t0} ,每跟踪完一条边界链 $t+1$),在每一条边界链上每搜索到一个新的边界点 $k+1$,把每一个边界链起始坐标 $P_{t0}(x, y)$ 和每一个边界点的链码值存入各自的边界链序列表中。

2) 从 P_{t1} 开始,即每一条边界链起始点后第二个边界点起,每完成一次新的边界点搜索后,将该点填充成背景色,边界跟踪回到起始点或者末尾点搜索不到新的边界点时再把起始点填充成背景色。

3) 由于从图像的左上角点开始,自上而下自左到右逐像素点扫描,每跟踪完一条边界链,该边界链被填充,因此下一个边界链的起始点 $P_{(t+1)0}$ 不会出现在上一边界链起始点 $P_{t0}(x, y)$ x 行左边和上方区域,如图 7 所示,因此新的边界链起始点的搜索扫描从上一个跟踪完成的边界链的起始点开始。

4) 如果边界链为单个像素的冗余点,把该单个像素置背景色并删除记录, t 保持初值 0 不变,继续从该点开始扫描寻找新的边界链,如果扫描所有像素没有找到边界点或者都是单个冗余像素点, $t=0$ 保持不变,算法结束。

2.6 算法时间复杂度分析

假设图像大小为 $n \times n$,文献[8]首先采用 Roberts 算子进行边缘提取,Roberts 算子为局部差分梯度算子,利用 2×2 的模板再开方,时间复杂度为 $O(8n^2)$;然后对边界图 8 邻域边界跟踪与交叉点搜索记忆,时间复杂度为 $O(8\rho n^2)$,其中 ρ 为边界像素与图像像素的比值(即周长与面积之比),同时,由于搜索到边界链末尾处依次变搜索窗口为 5×5 和 7×7 继续搜索,如果搜索到边界点则填充断点继续按 3×3 搜索,如果搜索不到则原路返回,所以本部分时间复杂度 $\geq O(8\rho n^2)$,为 $O(8\mu\rho n^2)$,其中 $\mu \geq 1$,因此文

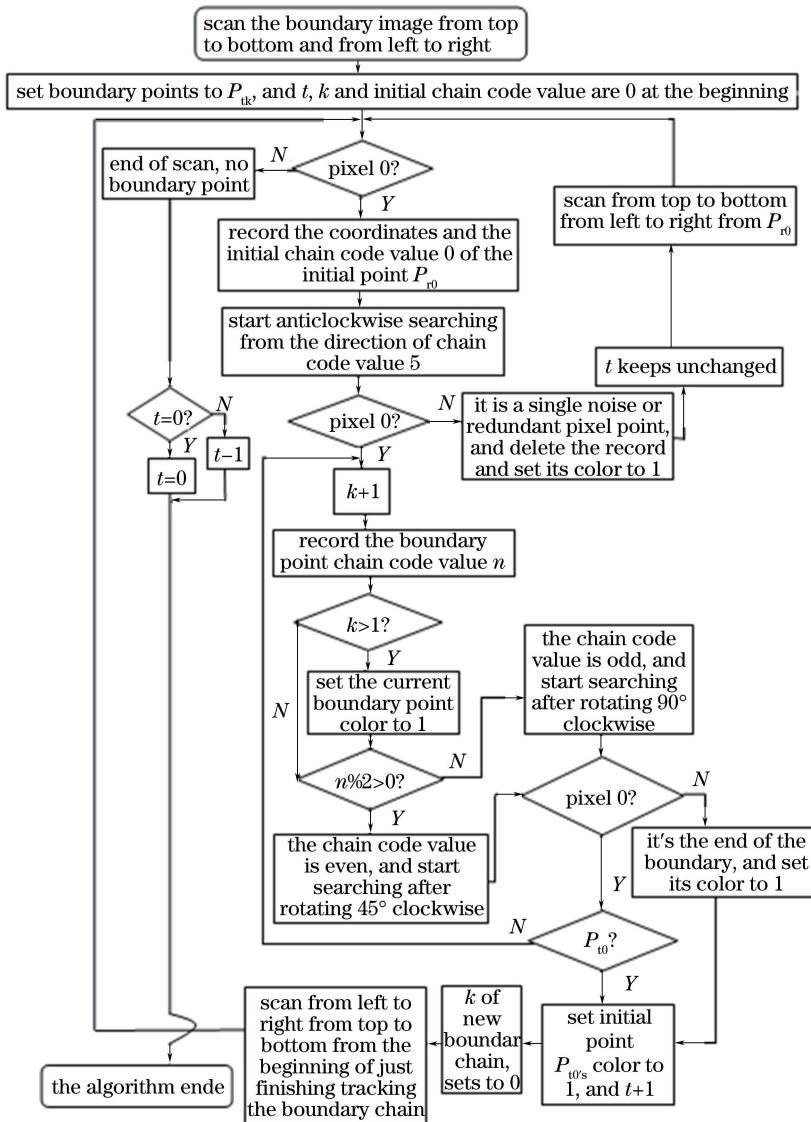


图 6 算法流程图

Fig. 6 Algorithm flow chart

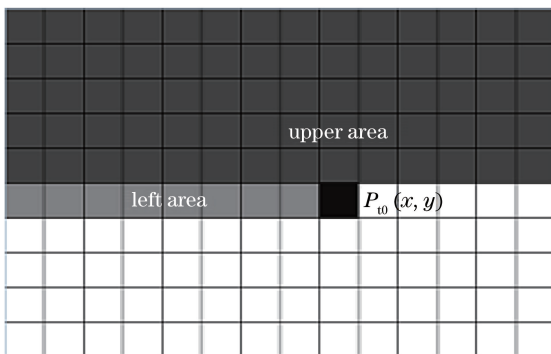


图 7 边界链搜索冗余区

Fig. 7 Boundary chain search for redundancy

献[8]算法整体时间复杂度为 $O(8n^2 + 8\mu\phi n^2)$ 。

文献[9]首先利用 8 邻域两遍扫描法分割出每

一个目标区域,时间复杂度为 $O(2n^2)$;然后对目标区域进行逐个边界跟踪,由于此过程非闭合区域原路返回,因此时间复杂度为 $O(3\mu\phi n^2)$,其中 $\mu \geq 1$;对每一个目标区域跟踪完后进行填充,时间复杂度为 $O(8an^2)$,其中 a 为所有目标区域面积与图像面积之比, $0 < a < 1$;此外,对下一个目标区域边界起始点的搜索是返回图像原点重新扫描,此过程中存在大量的重复搜索,因此寻找起始点的时间复杂度为 $O(An^2)$,其中 $A > 1$,因此文献[9]算法整体时间复杂度为 $O(An^2 + 2n^2 + 8an^2 + 3\mu\phi n^2)$ 。

本文算法首先用四连通器提取边界,时间复杂度为 $O(4n^2)$,然后自上而下自左而右扫描寻找每一段边界的起始点,时间复杂度为 $O(n^2)$,跟踪和填充边界时间复杂度为 $O(2.5\phi n^2)$,因此本文算法整体

时间复杂度为 $O(5n^2 + 2.5pn^2)$ 。

从以上算法时间复杂度分析可见,本文算法的效率明显高于文献[8]和文献[9]算法的效率真。

3 实验与分析

3.1 实验说明

硬件条件为 Intel © Pentium © CPU J2900@2.41 GHz; 软件环境为 Visual Studio2013 × 64Debug 模式 OpenCV3.1。

图 8 为手持棋盘图像的二值图像,该图具有交叉边界、孔洞和噪声冗余点,图片大小为 2050×1540 。图 9 为文献[8]算法选择 30 个种子点的边界跟踪结果图。图 10 为文献[9]算法的边界跟踪结果图。图 11 为采用本文算法的边界跟踪结果图。效率实验采用 4 组每组 10 张大小相同的二值图像,1、2、3、4 组分别对应的图片大小依次为 $2050 \text{ pixel} \times 1540 \text{ pixel}$ 、 $4216 \text{ pixel} \times 3484 \text{ pixel}$ 、 $1200 \text{ pixel} \times 790 \text{ pixel}$ 、 $548 \text{ pixel} \times 312 \text{ pixel}$,运用 3 种算法分别计算每张图片的时间,取每组的平均时间。对于文献[8]的实验,在微软基础类库(MFC)框架下先提取二值图像边界,再利用鼠标在边界图上点击选择 30 个种子点进行边界跟踪,两部分时间之和即为一次算法的总时间,文献[9]与本文算法的时间计算均为一次完成。

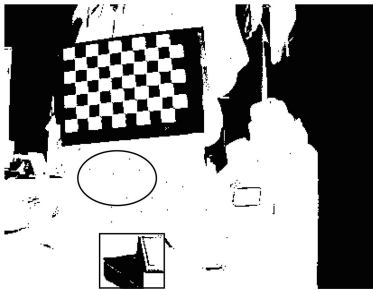


图 8 原二值图像图

Fig. 8 Original binary image



图 9 文献[8]算法的边界跟踪结果图

Fig. 9 Boundary tracking result of Ref. [8]

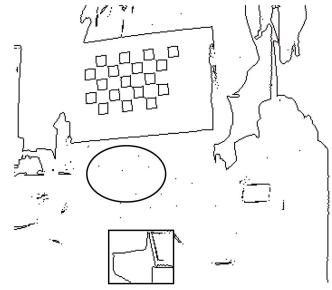


图 10 文献[9]算法的边界跟踪结果图

Fig. 10 Boundary tracking result of Ref. [9]



图 11 本文算法的边界跟踪结果

Fig. 11 Boundary tracking result of proposed algorithm

3.2 结果与分析

3.2.1 交叉边界无漏跟踪

对比图 8、图 9 和图 10 中棋盘格部分可见,对于交叉边界,文献[8]算法即使选择了 30 个种子点,依然存在漏跟踪情况,可见人工选取种子点难以以保证精度,而本文算法边界跟踪完整,无漏跟踪问题。

3.2.2 内部孔洞跟踪

文献[9]同样也无法保证边界跟踪精度,存在内部孔洞边界漏跟踪问题,而本文算法对内部孔洞边界跟踪效果良好。对比图 8、图 10 和图 11 方框区域,对应的方框区域放大图如图 12 所示。

对比图 12 可见,文献[9]算法对原图内部的孔洞边界没有跟踪到。按照文献[9]算法,在跟踪完一个目标区域的外部边界后,若还需要跟踪其他目标区域边界,需要返回图像原点重新自上到下自左到右扫描,因此,只能跟踪到目标区域的外部边界而无法跟踪到内部孔洞的边界,并会导致大量的重复扫描。

3.2.3 噪声冗余点去除

对比图 8、图 10 和图 11 中的椭圆框区域,图 8 内的噪声冗余点在文献[9]的实验中依然存在,而在本文改进算法中则被全部去除。根据本文改进算法,当边界链起始点搜索不到新的后续边界点时,判断为长度为 1 的单像素,即视为噪声冗余点直接置背景色并删除链码记录。

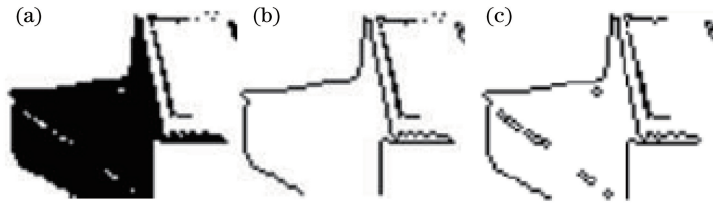


图 12 内部孔洞。(a)原图;(b)文献[9]算法跟踪;(c)本文算法跟踪

Fig. 12 Inner hole image. (a) Original image; (b) tracking results of Ref. [9]; (c) tracking results of proposed algorithm

3.2.4 跟踪效率

表 1 为算法运行时间的比较。由表 1 可知,本文算法的运行时间效率相比文献[8]算法的运行时间效率平均提高了 31.68%,最大提高了 32.14%,最少提高了 31.17%;比文献[9]算法的运行时间效率平均提高了 19.47%,最大提高了 19.8%,最少提高了 19.12%。可见,本文算法可以显著提高运行速度。

表 1 算法运行时间的比较

Table 1 Comparison of the calculation time

Picture number	Ref.[8] algorithm	Ref.[9] algorithm	Proposed algorithm
1	0.7209	0.6114	0.4923
2	1.4270	1.2156	0.9749
3	0.4174	0.3568	0.2873
4	0.1932	0.1621	0.1311

4 结 论

阐述了一种二值图像边界跟踪的新算法,通过对边界分段和跟踪填充实现对二值边界图扫描一次即可完成对所有边界的跟踪,可避免重复扫描,实现对岛、洞、多交叉的复杂网络边界进行跟踪,同时对于每一条边界链上边界点的搜索跟踪,都根据前后边界点的相对链码值的奇偶性去掉冗余,使平均搜索次数为 2.5,并通过设置阈值实现抗噪去除冗余点。实验证明,本文算法具有较高的效率、良好的跟踪效果和抗噪去冗余点的性能,但分段可能破坏轮廓完整性,以及设置阈值去噪可能存在误删等问题有待进一步改进。

致谢 感谢广西高等学校高水平创新团队及卓越学者计划资助。

参 考 文 献

[1] Seo J, Chae S, Shim J, *et al.* Fast contour-tracing algorithm based on a pixel-following method for image sensors[J]. *Sensors*, 2016, 16(3): 353.
 [2] Jordan H, Dyck W V, Smodić R. A co-processed contour tracing algorithm for a smart camera [J].

Journal of Real-Time Image Processing, 2011, 6(1): 23-31.

- [3] Zhou X Z, Chen Y, Hu W T. Tree traversal binary image boundary tracking algorithm based on cross-point [J]. *Computer Applications and Software*, 2014, 31(2): 230-232.
 周秀芝, 陈洋, 胡文婷. 基于交叉点的树遍历二值图像边界跟踪算法 [J]. *计算机应用与软件*, 2014, 31(2): 230-232.
 [4] Hua W P, Zhao J F, Li M, *et al.* Single image noise estimation based on image segmentation and scatter statistics of noise [J]. *Laser & Optoelectronics Progress*, 2016, 53(4): 041006.
 华玮平, 赵巨峰, 李梦, 等. 基于分块与噪声散点统计的单图噪声估计 [J]. *激光与光电子学进展*, 2016, 53(4): 041006.
 [5] Lu M T, Su X Y, Cao Y P, *et al.* 3D shape reconstruction algorithms for modulation measuring profilometry with synchronous scanning [J]. *Chinese Journal of Laser*, 2016, 43(3): 0308006.
 卢明腾, 苏显渝, 曹益平, 等. 同步扫描的调制度测量轮廓术三维面形重建算法 [J]. *中国激光*, 2016, 43(3): 0308006.
 [6] Chen W, Guo Q S, Fan W, *et al.* A rapid algorithm on tracing edge of binary image and its application [J]. *Engineering of Surveying and Mapping*, 2014, 23(12): 63-66.
 陈旺, 郭庆胜, 范伟, 等. 二值图像中目标物体边界跟踪的一种快速算法及应用 [J]. *测绘工程*, 2014, 23(12): 63-66.
 [7] Cui F K, Zhang F S, Bai L, *et al.* Using neighborhood points to trace edge of object [J]. *Journal of Luoyang Institute of Technology*, 2001, 22(1): 28-30.
 崔凤魁, 张丰收, 白露, 等. 二值图像目标邻域点法边界跟踪算法 [J]. *洛阳工学院学报*, 2001, 22(1): 28-30.
 [8] Wang J, Lu Y P, Zhang Z H, *et al.* A reptile method with memory and alterable window for image boundary tracking [J]. *Chinese Journal of Scientific*

- Instrument, 2004, 25(4): 483-486.
- 王珏, 卢艳平, 张泽宏, 等. 一种有记忆的变窗“爬虫”图像边界跟踪方法[J]. 仪器仪表学报, 2004, 25(4): 483-486.
- [9] Wang F S, Qi G Q. Boundary tracking algorithm of objects in binary images [J]. Journal of Dalian Maritime University, 2006, 32(1): 62-64.
- 王福生, 齐国清. 二值图像中目标物体轮廓的边界跟踪算法[J]. 大连海事大学学报, 2006, 32(1): 62-64.
- [10] Zhou F L, Xu X M, Xiao Y, *et al.* A new algorithm on objects boundary tracking of binary images [J]. Microcomputer Information, 2007, 23(6): 259-261.
- 周丰乐, 徐向民, 肖跃, 等. 一种新的二值图像目标轮廓跟踪算法[J]. 微计算机信息, 2007, 23(6): 259-261.
- [11] Li Z P, Li P. Image segment contour tracking algorithm based on chain code and multithreading[J]. Microcomputer Applications, 2011, 27(12): 30-32.
- 李贞培, 李平. 基于链码和多线程的图像分段轮廓跟踪[J]. 微型电脑应用, 2011, 27(12): 30-32.
- [12] Kuagoolkijgarn P, Koomsap P, Chansri N. A new algorithm for tracing nests of interconnected contours [J]. International Journal of Advanced Manufacturing Technology, 2010, 50 (5-8): 717-727.
- [13] Milan S, Vaclav H, Roger B. Image processing, analysis, and machine vision[M]. Fourth edition. Xu J L, Ai H Z, Transl. Beijing: Tsinghua University Press, 2015: 241.
- 米兰, 瓦茨拉夫, 罗杰. 图像处理, 分析与机器视觉[M]. 4版. 徐军亮, 艾海舟, 译. 北京: 清华大学出版社, 2015: 241.
- [14] Su L, Wu J J, Pang D. Panoramic sea-sky-line detection based on improved active contour model[J]. Acta Optica Sinica, 2016, 36(11): 1115003.
- 苏丽, 吴俊杰, 庞迪. 基于改进主动轮廓模型的全景海天检测[J]. 光学学报, 2016, 36(11): 1115003.