

改进的多边形扫描转换方法

张志刚¹, 刘小冬¹, 张志强², 张曰贤¹

ZHANG Zhi-gang¹, LIU Xiao-dong¹, ZHANG Zhi-qiang², ZHANG Yue-xian¹

1. 西安财经学院 信息学院, 西安 710061

2. 西安广播电视大学, 西安 710002

1. Department of Information, Xi'an University of Finance and Economics, Xi'an 710061, China

2. Xi'an Radio and Television University, Xi'an 710002, China

E-mail: zzg.gg@163.com

ZHANG Zhi-gang, LIU Xiao-dong, ZHANG Zhi-qiang et al. Improved method for polygon scan conversion. Computer Engineering and Applications 2009, 45(4): 193-195.

Abstract: Aiming at shortcoming of traditional edge-labeled algorithm, an improved method for polygon scan conversion is proposed through analyzing the solutions processing horizontal edge and the reason causing horizontal edge. The information of adjacent edges is introduced in this algorithm and the polygon can be filled correctly including horizontal and near horizontal edge and it is easy to be realized and applicable to complicated shape polygons.

Key words: edge-labeled scan conversion; polygon

摘要: 针对传统的边标志算法的不足, 分析了目前对于水平边填充的解决方法, 研究了水平边出现的原因, 在此基础上引入了水平边的邻边特征等信息, 提出了一种改进的方法。它能正确地填充含有水平边或近似水平边的多边形, 且简单易实现, 适用于复杂形状的多边形。

关键词: 边标志; 扫描转换; 多边形

DOI: 10.3778/j.issn.1002-8331.2009.04.055 文章编号: 1002-8331(2009)04-0193-03 文献标识码: A 中图分类号: TP391

1 引言

多边形扫描转换是一种重要的区域填充方式, 也是计算机图形学的基本问题之一。通常可以分为几类^[1]。x-扫描线算法是按扫描线顺序求出扫描线与多边形的相交区域, 再进行填充, 这种方法的主要不足是每条扫描线都要和多边形各边求交, 且需排序; 有效边表算法则考虑了扫描线及多边形边的连贯性, 提高了算法的效率, 但增加了对各种表的维持开销。边缘填充算法将扫描线与多边形边交点的右方像素取补, 算法简单, 但像素会被多次访问。栅栏填充算法通过设置栅栏减少了被重复访问的像素数目, 但未从根本上解决问题。边标志算法扫描出边界后再进行填充, 使每个像素的访问次数不超过 2 次, 这种方法简单高效, 易于实现, 在实际中得到了广泛应用, 但传统的边标志算法在处理水平边时还存在着不足。本文对现有的方法进行了分析, 研究了水平边出现的原因, 在此基础上提出了一种改进算法, 既保留了边标志算法的优势, 又使之可以处理水平边, 最后通过实例进行了分析。

2 边标志算法分析

边标志算法的步骤主要分为两步:

(1) 对多边形的每条边进行直线扫描转换, 即在多边形边界所经过的像素处打上边标记;

(2) 对每条与多边形相交的扫描线, 按从左到右的顺序逐个访问扫描线上的像素, 并设置一个布尔变量 *inside*, 在每行开始时令 *inside* 为 F, 每当访问到边标记像素时, 将 *inside* 取反, 否则 *inside* 不变。经过如此处理后若 *inside* 为 T, 就将该像素置为填充色, 如图 1 所示。

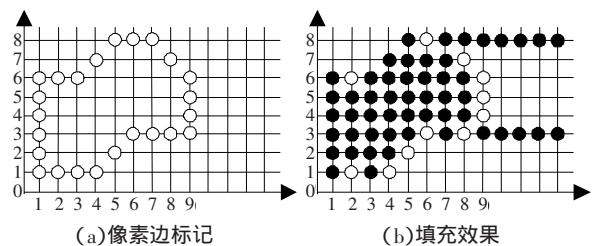


图 1 边标志算法示意图

基金项目: 陕西省自然科学基金(the Natural Science Foundation of Shaanxi Province of China under Grant No.SJ08F32), 陕西省教育厅自然科学研究计划(the Natural Science Research Plan of Education Department of Shaanxi Province of China under Grant No.08JK290), 国家统计局全国统计科学研究项目(the Key Program of National Bureau of Statistics of China under Grant No.LX2005-20)。

作者简介: 张志刚(1970-), 男, 博士, 讲师, 主要研究方向: 图形图像处理; 刘小冬(1963-), 男, 博士, 教授, 主要研究方向: 人工智能; 张志强(1975-), 男, 讲师, 主要研究方向: 现代教育技术; 张曰贤(1966-), 男, 讲师, 主要研究方向: 信息技术与管理。

收稿日期: 2008-08-21 修回日期: 2008-11-13

由图 1 可得:在没有水平边出现的扫描线上能得到正确的填充效果,如图 1(b)的第 2、4、5、7 行;在出现水平边的第 3 行,像素 $P(9,3)$ 右方的非区域内像素被错误地填充,其主要原因就是水平边上像素数目是偶数,使得 *inside* 在最后的边标记像素处为 T,但同样在出现水平边的第 1 行,尽管水平边上像素数也是偶数,但像素 $P(4,1)$ 右方的非区域内像素未被填充。对于具有奇数个像素的水平边而言,也有着类似的结果,如第 6 行和第 8 行。

对此,若不考虑水平边^[2],也可能导致误填,仍以图 1(a)为例,填充结果如图 2 所示。

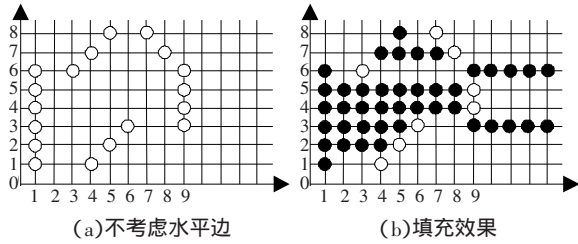


图 2 不考虑水平边的填充效果

文献[3]也提出了类似的改进思路,认为应该忽略水平边上除两个端点以外的其他边界点,即若当前被处理像素与其相邻的两个像素都被打上了边标志,则不做处理,否则布尔变量取反。这种方法只考虑了两个端点而未考虑与端点关联的两条邻边的位置关系。

3 改进算法

从上述分析可得,影响水平边填充效果的主要原因在于与水平边邻接的另两条边的相对位置关系,而不仅仅取决于水平边本身的两个端点和边上像素数目的奇偶性,如图 3 所示。

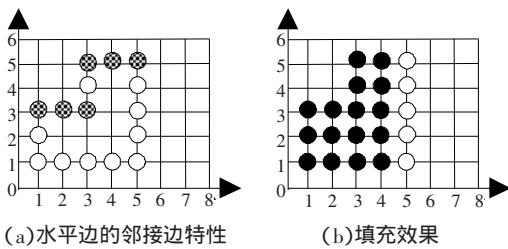


图 3 填充效果示意图

图 3(a)的多边形其第 3 行和第 5 行都有像素数目为 3 的水平边(阴影像素),但对于第 3 行的水平边,与其邻接的两条边分列在水平边两侧,故水平边的两个端点中只能对一个点做边标记;而对于第 5 行的水平边,由于其邻接的两条边都在水平边一侧,因此水平边的两个端点均应做边标记,这样才能得到正确的结果,如图 3(b)所示。

此外,对于多边形中出现的水平边,不仅有两端点同处于一行的真正的水平边,如图 4 中第 6 行的边,而且也存在因直线扫描转换而产生的“水平直线段”,如图 4 中第 1、2、3 行所示的阴影像素组成的 3 段水平线段,它们实际是以 $(1,1)$ 和 $(8,3)$ 为端点的直线上的像素,而这类因直线斜率介于 $[-1,1]$ 之间使得数字化后导致的“水平直线段”不在少数,它们也存在着和第一种水平边类似的填充问题。对于这类水平线段往往无法事前判定其端点位置,所以处理起来较第一种水平边更复杂。

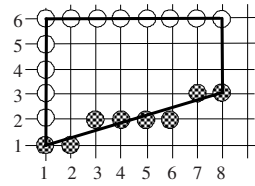


图 4 水平边的类型

综合上述分析,要正确地对含水平边、或接近水平的边的多边形进行填充,不仅要有多边形各边像素进行标记,以得到边界像素的位置,而且要保存被标记的像素间的相对位置关系,以便于对水平边进行正确的填充。对此,参考文献[4-5]等对线段端点的定义及区分的基础上,设 $P_1(x_1, y_1)$ 、 $P_2(x_2, y_2)$ 、 $P_3(x_3, y_3)$ 是某直线扫描转换后所得点集中连续的 3 个像素,以此为例,可将已经过边标记处理后的像素进一步分为几类:

- (1)水平边起点:若 $y_2 \neq y_1$ & $y_2 = y_3$, 则 P_2 即为水平边的起始端点,记为 a 型。
- (2)水平边终点:若 $y_2 \neq y_3$ & $y_2 = y_1$, 则 P_2 即为水平边的终止端点,记为 b 型。
- (3)水平边内部点:若 $y_2 = y_3$ & $y_2 = y_1$, 则 P_2 即为水平边的内部点,记为 c 型。
- (4)一般边界点:若 $y_2 \neq y_1$ & $y_2 \neq y_3$, 则 P_2 即为一般的边界点,记为 d 型。

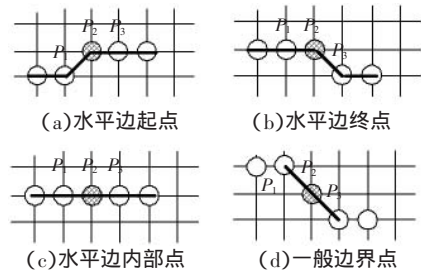


图 5 标志点的类型

对于一般边界点和水平边起点,可按传统的边标志算法进行处理;对于水平边内部点则应予以忽略;对于水平边终点,其处理过程还应取决于水平边起点的邻边、和水平边终点的邻边的相对位置。为此再设置一个记录邻边位置的标志 *direction*。整个改进算法的步骤如下:

(1)利用直线扫描转换算法对给定的多边形的每对相邻顶点进行直线扫描转换,得到一有序的点列,然后再剔除重复的像素点,如此将形成连续闭合的有序边标志点列 P ;

(2)确定多边形占有的扫描线数,对于每一条扫描线 $y=i$ 上的像素进行如下过程的填充:

```

inside=F direction=0 %分别设置两个变量的初值;
for j=1 maxcolumn %扫描线逐条处理;
    if pixels(i,j)=d inside=! inside %若当前点是 d 型点 inside 取反;
    elseif pixels(i,j)=a %若当前点是 a 型点 inside 取反,同时记录其 direction;
        inside=! inside;
        direction=P(index).y-P(index-1).y; % index 表示当前点在点列 P 中的序号;
    elseif pixels(i,j)=b & P(index).y-P(index+1).y=direction
        %若当前点是 b 型点,且其邻边方向与起点的 direction 一致,则 inside 取反,同时 direction 置为 0;
    
```

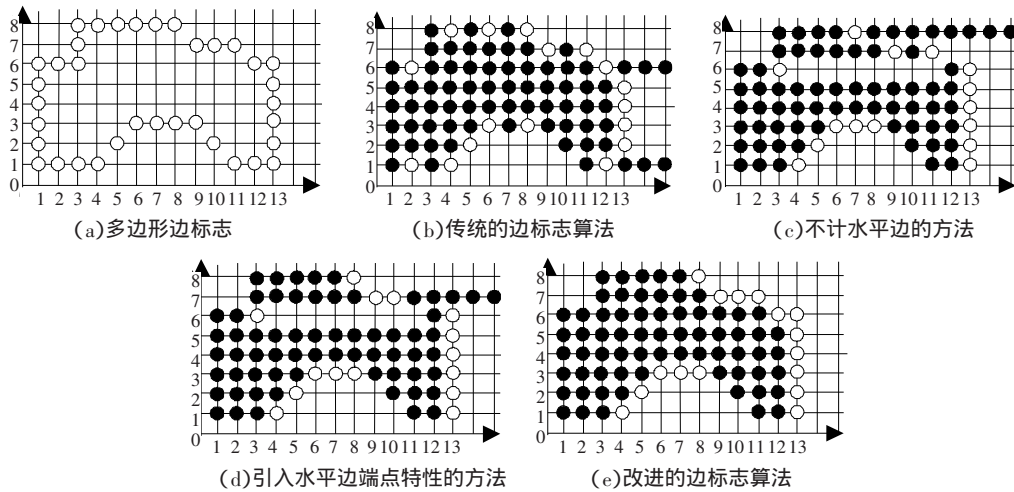


图6 实验结果

```

inside=! inside ;
direction=0 ;
end %end of if
if inside=T %经过上述的处理,如果 inside 为真,将该像素置
成填充色 ;
    putpixel(i,j,'color') ;
end %end of if
end %end of for

```

上述算法第一步中的剔除重复像素点主要是考虑数字化处理后的多条边有可能共享公共的点,因此要予以剔除,但对于多边形两相邻边共享的顶点,其顶点数目要根据传统 x 扫描线算法中扫描线与顶点相交的顶点计数策略进行统计,即为共享顶点的两条边另两个端点,其 y 值中大于共享顶点 y 值的个数,故此类情况下出现的重复像素点是不能剔除的。

4 实验与分析

图6所示是以一个比较复杂的多边形为对象,用改进的算法进行多边形扫描转换,并与其他多边形标志算法进行了对比。其中(a)为多边形的经过标记的边,(b)所示的方法在第1、6行由于水平边像素数目性质不同(奇数或偶数)而导致误填;(c)为应用文献[2]的方法进行填充的效果,因为不计水平边,所以在第6行出现了漏填、第8行出现了误填,(d)所示是采用文

献[3]的方法,尽管考虑了两个端点的特性,但忽略了与端点关联的两条邻边的位置关系,因而在第6行出现了漏填,第7行出现了误填。(e)为本文改进的边标志算法的填充结果,对于第1行的两段水平边,和第3、8行的水平边,与其相邻的边都在水平边一侧,而对于第6行的两段水平边则邻边处于异侧,由于进行了不同的处理,故能得到正确的填充效果。通过实验表明,利用改进的填充算法能处理各种情况的水平边,包括与扫描线相交的多边形顶点形成的带重复顶点的水平边等情况,但本算法的主要不足是增加了对边标志点列数组的维持开销,以及新的变量 $direction$;另外,对于水平边本身的填充也不具有统一性,如在(e)中第1、8行对水平边进行了填充,而第3行的水平边则没有,这也将是后续研究的重点。

参考文献:

- [1] 孙家广,杨长贵.计算机图形学[M].北京:清华大学出版社,1997.
- [2] 陈传波,陆枫.计算机图形学基础[M].北京:电子工业出版社,2002.
- [3] 王秀华,严兵.一种边标志填充的改进算法[J].计算机应用,2004,24(6):182-183.
- [4] 吴章文,杨代伦,勾成俊,等.区域填充极点判别算法[J].计算机辅助设计与图形学学报,2003,15(8):979-983.
- [5] 李波,王刚,刘东华,等.基于边界标注的单连通区域扫描线填充新方法[J].空军工程大学学报:自然科学版,2003,4(4):65-68.

(上接 182 页)

- [2] Wu S H, Xie X D, Liew A W, et al. Eukaryotic promoter prediction based on relative entropy and positional information [J]. Physical Review E, 2007, 75(04):1908:1-7.
- [3] Knudsen S. Promoter2.0: For the recognition of Po promoter sequences [J]. Bioinformatics, 1999, 15: 356-361.
- [4] Down T A, Hubbard T J. Computational detection and location of transcription start sites in mammalian genomic DNA [J]. Genome Res, 2002, 12: 458-461.
- [5] Bajic V B, Seah S H, Chong A, et al. Computer model for recognition of functional transcription start sites in polymerase promoters of vertebrates [J]. Journal of Molecular Graphics & Modeling, 2003, 21: 323-332.
- [6] Prestridge D S, Burks C. The density of transcription elements in promoter and non-promoter sequences [J]. Hum Mol Genet, 1993, 2:

1449-1453.

- [7] Cross S H, Clark V H, Bird A P. Isolation of CpG islands from large genomic clones [J]. Nucleic Acids Res, 1999, 27: 2099-2107.
- [8] Davuluri R V, Grosse I, Zhang M Q. Computational identification of promoters and first exons in the human genome [J]. Nat Genet, 2001, 29: 412-417.
- [9] Scherf M, Klingenhoff A, Werner T. Highly specific localization of promoter regions in large genomic sequence by PromoterInspector: A novel context analysis approach [J]. J Mol Biol, 2000, 297: 599-606.
- [10] Sergios T, Konstantines K. Pattern recognition [M]. San Diego, USA: Academic Press, 2003.
- [11] Schapire R, Singer Y. Improved boosting algorithms using confidence-rated predictions [J]. Computational Learning, 1998, 37: 297-336.