

基于平面扫描的线状缓冲区生成的改进算法*

许允波¹, 张建兵², 谭宁生¹

(1. 中国科学院遥感应用研究所 遥感科学国家重点实验室, 北京 100101; 2. 中国石油大学(北京) 信息学院, 北京 102249)

摘要: 针对传统双线圆弧算法存在失真现象和效率低下问题, 提出了一种改进的基于线段/弧段的平面扫描算法。首先通过在凹点记录线段来避免失真问题; 然后利用线段/弧段之间距离远近和相互连接的性质, 采用改进的平面扫描算法大幅度地提高求交效率; 同时设计一种算法快速剔除非边界点; 最后利用线段/弧段具有前进方向的性质, 快速构造最终缓冲区边界。该算法能避免失真问题, 精确高效, 具有较高的空间和时间利用率。

关键词: 缓冲区; 线状; 平面扫描; 地理信息系统

中图分类号: TP399; TP301.6

文献标志码: A

文章编号: 1001-3695(2012)11-4364-03

doi:10.3969/j.issn.1001-3695.2012.11.094

Improved algorithm for line buffering based on plane sweep technique

XU Yun-bo¹, ZHANG Jian-bing², TAN Ning-sheng¹

(1. State Key Laboratory of Remote Sensing Information Sciences, Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing 100101, China; 2. Information Institute, China University of Petroleum-Beijing, Beijing 102249, China)

Abstract: This paper presented an improved algorithm, which could resolve distortion and inefficient problems existing in traditional double parallel lines and circular arcs method. First, it recorded segments in concave point to avoid distortion. Then by using distance and interconnected nature, using improved plane sweep algorithm could substantially improve the efficiency of intersect. At the same time, it designed an efficient algorithm to remove the non-boundary points. Lastly, it built the final buffer boundary by using the forward direction nature of line/arc segment. This algorithm can avoid distortion, and is accurate and efficient, with high spatial and time utilization.

Key words: buffer; linear; plane sweep; GIS

缓冲区分析是地理信息系统中使用非常频繁的一种空间分析, 是对空间特征进行度量的一种重要方法, 在交通、林业、资源管理、城市规划、环境与生态保护等领域都有着广泛的应用。GIS 中缓冲区生成包括点、线、面三种目标类型的缓冲区的生成, 其中线状目标缓冲区的生成是关键。

现有的缓冲区生成算法主要有栅格和矢量两种方法。栅格方法原理简单, 容易实现, 但是效率和内存开销取决于栅格的密度。栅格方法最初的研究集中在栅格距离变换方法^[1,2], 这类算法时间取决于图幅; 最近的研究将目标对象的缓冲区栅格化, 然后利用矢量追踪提取缓冲区的边界^[3,4], 算法时间取决于对象本身, 效率虽有所提升, 但还是难以突破精度和效率的瓶颈。矢量算法原理复杂, 不易实现, 但是效率和精度较高, 内存占用少。矢量算法主要有基于双线圆弧法的跟踪算法和基于单个缓冲区单元的叠加算法^[5-7]。双线圆弧算法存在失真问题, 大部分文献对此没有处理或对失真情况的处理十分复杂, 难以完备实现^[6,8-10]。基于单个缓冲区的叠加算法因存在大量冗余数据, 造成大量不必要的求交运算, 效率大受影响^[7]。此外, 对于比较耗时的求交计算, 大部分文献也较少涉及。

文献[7,9]中提出的平面扫描算法均能避免失真问题, 求

交效率显著提高, 但是两者都存在着大量的冗余数据。文献[7]没有顾及到相邻两个线段的缓冲区有很大一部分是公共缓冲区边界。文献[11]对于失真问题的简单处理会造成明显在缓冲区内部的线段却参与求交, 另外求交和生成最终缓冲区边界时没有利用缓冲区边界本身的连接和方向性质, 在求交、剔除缓冲区内部的交点和跟踪生成缓冲区边界的问题上仍需改进。

本文在文献[7,11]的基础上对基于线段/弧段的平面扫描算法加以改进。首先介绍双线圆弧算法造成失真问题的原因及改进方法; 然后通过剔除/更新大部分不必要的缓冲区边界, 减少参与求交的线段/弧段单元, 并利用线段/弧段距离远近和相互之间连接的性质, 采用改进的平面扫描算法提高求交的效率, 同时设计了一种高效的非边界交点的剔除算法; 最后采用两多边形求并的思路, 完成缓冲区的快速构建。

1 失真问题处理及算法流程

1.1 失真问题及处理策略

传统的双线圆弧算法的基本思想是: 在轴线的两端用半径为缓冲距的圆弧弥合; 在轴线的各个转折点, 首先判断该点的凹凸性, 在凸侧用半径为缓冲距的圆弧弥合, 在凹侧用与该点

收稿日期: 2012-03-12; 修回日期: 2012-04-24 基金项目: 国家“863”计划资助项目(2012AA12A401)

作者简介: 许允波(1987-), 男, 山东菏泽人, 硕士研究生, 主要研究方向为网络空间信息系统(870923xuyunbo@163.com); 张建兵(1974-), 男, 湖北洪湖人, 讲师, 博士, 主要研究方向为网络空间信息系统、网格GIS; 谭宁生(1983-), 男, 安徽宁国人, 博士研究生, 主要研究方向为网络空间信息系统。

关联的前后两相邻线段的偏移量为缓冲距的两平行线的交点作为对应顶点^[8]。该算法最大的不足在于会产生尖角或凹陷等失真现象。虽然龚洁晖等人将缓冲区边界失真现象归纳为三类共十六种^[8],并一一给出了修正处理方法。但实际上缺少凹侧两侧均为凹侧这一类情况的处理,且修正处理较为复杂,难以完备实现。

折线的缓冲区可以看做是各个组成线段的缓冲区合并的结果^[4],因此在凹侧的缓冲区边界的处理上应视为两条线段而非平行线。与平行线不同,线段可能没有交点,此时如果仍用平行线的交点作为对应顶点,便会造成失真问题,如图 1 所示。因此,在凹侧的处理上,如果两条线段有交点,则仍采用传统的处理方式;如果没有交点,则记录凹侧的两条线段,不作处理。这样便可避免失真问题,由此产生的多余线段可和自相交问题在生成最终缓冲区边界时统一处理。

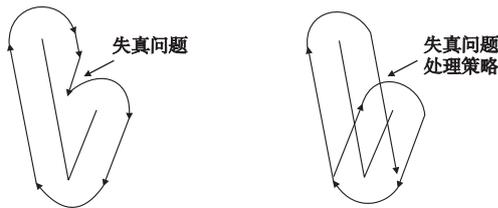


图1 失真问题及处理策略

1.2 算法流程

与传统的双线圆弧算法不同,本文借鉴文献[11]的思想,以线段/弧段作为基本处理单元,通过在凹侧记录两条线段来避免失真问题。图 2 展示了算法的整个处理流程,由生成初始缓冲区边界、计算交点、剔除非边界点和追踪生成缓冲区边界四大部分依次组成。之后将对每个步骤、关键问题以及解决方案作详细介绍。

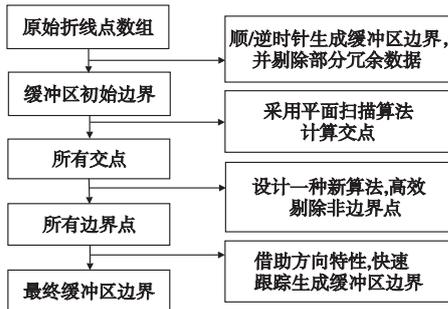


图2 缓冲区算法处理流程

2 高效精确的线状缓冲区生成算法

2.1 构造缓冲区初始边界

和传统的双线圆弧算法相同^[6],在凸侧加入圆弧,在凹侧加入线段,同时记录线段/弧段的前进方向,按顺/逆时针顺序最终生成一个线段/弧段混合数组。其中有所不同的是当凹侧的两条线段不相交时,通过直接记录两条线段处理失真问题,生成的初始缓冲区边界如图 3 所示。

对于失真问题的简单处理会带来大量的冗余数据,不少线段位于缓冲区的内部,参与求交没有任何意义。适当地剔除缓冲区内部的线段以减少参与求交的线段对于提高整个算法效率大有裨益。本文参考文献[8]中提出的三类十六种情况,根据凹点两侧类型的分布情况,细分为两侧均为凹点、两侧均为凸点、前凸后凹和前凹后凸四种情况予以处理。在构造缓冲区

初始边界的过程中,当凹点的两条线段不相交时,较短的线段一定位于缓冲区的内部,可以直接删除;对于较长的线段,需要根据凹点两侧类型的分布适当处理。在构造初始边界过程中更新/删除部分不必要的缓冲区单元后的边界如图 4 所示。

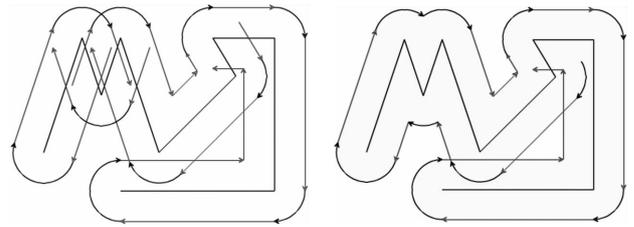


图3 初始缓冲区边界

图4 去除部分冗余数据的初始缓冲区边界

2.2 计算交点

交点的计算是整个缓冲区算法中最为耗时的操作。最为简单的是暴力求交,时间复杂度为 $O(n^2)$ 。当折线很长时,这种方法十分耗时,效率低下。为了提高求交效率,需要充分利用线段/弧段的几何性质:

- a) 只有那些相互靠近的线段/弧段才可能会相交,而相距甚远的线段/弧段则不可能相交^[13]。
- b) 线段/弧段之间具有连接性质。

针对第一个几何特性,可使用计算几何中的平面扫描算法^[12-15],但是该算法针对的对象是随机分布的线段集,没有考虑到线段之间的连接性质,而且需要处理的对象是线段/弧段这两种不同类型的单元,因此需要对平面扫描算法作出改进。

改进 1 需要把非单调的弧段截断成两部分,使截断后的弧段具有单调性质。因为构造初始缓冲区边界时生成的弧段角度跨度不会超过 180° ,所以弧段只有三种情况:a)沿 X 轴单调的弧段;b)有最左点的弧段;c)有最右点的弧段。对于后两种都需要进行截断处理,将弧段分成两部分。

改进 2 初始化生成事件点队列时,事件点分成 left、right、left&right、right&left、left&left、right&right 六种,如图 5 所示。对于类型为 left 的事件点,需加入其所隶属的线段/弧段到扫描线列表中。对于 right 类型,则从扫描线列表中删除其所隶属的线段/弧段,这些和基于线段的平面扫描算法一致。对于 left&right、right&left 两种类型的事件点,首先删除 right 所属的线段/弧段,此时不对新生成的上下两个单元进行相交判断,而是在 right 的位置直接加入 left 所属的线段/弧段,然后进行相交判断,这样可以节省一次相交判断和一次位置查找过程。在生成的初始缓冲区边界中,有很多这样的连接点,通过这种处理可以节省大部分时间。另外对于 left&left 类型处理时,需要判断两个 left 的上下顺序,以便能够插入到扫描线列表中的正确位置。本文的处理方法为对扫描线向右进行一个微小的偏移,以确定其上下顺序。

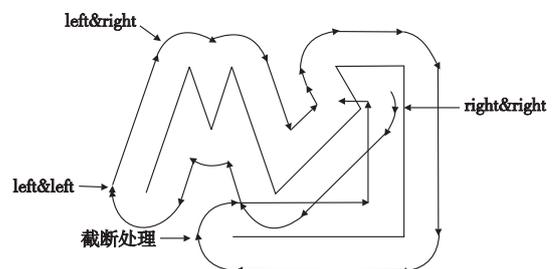


图5 截断处理以及事件点类型

2.3 剔除非边界点

经过求交运算计算出来的交点有些可能位于整个缓冲区的内部,需要予以剔除。当折线的点很多(n 个)、交点也很多(m 个)时,通过计算每个交点到折线的各个线段的最短距离的方法进行判断,时间复杂度为 $O(mn)$,效率十分低下,因此需要一种高效的剔除算法。

实际上,折线上的很多线段离交点很远,根本无须参与剔除计算,因此快速找到那些需要判断的线段是提高剔除算法的关键。本文设计了一个算法,能够利用前一个交点进行剔除判断后的中间结果,从而十分迅速地找到当前交点对应的需要判断的多个线段,效率大为提升。算法的思路如下:

a) 对折线点数组生成线段数组,每个线段包含两个点 $start, end$, 并且满足 $start.x \leq end.x$, 然后对线段数组按 $start.x$ 递增进行归并排序;同时也对交点数组按 x 递增进行归并排序。如图 6 所示,排序后,线段数组为 01、12、54、56、34、32、67、78;交点数组为交点 1、2。

b) 每个交点在 x 方向上有一个缓冲区间 $[x - radius, x + radius]$, 即图 6 中对应的 $[leftX, rightX]$ 。凡是全部或者部分位于这个区间的线段都是需要判断的线段,取舍条件是 $start.x < rightX \ \&\& \ end.x > leftX$ 。用一个临时线段数组 $segments$ 存放前一交点记录下来的多个线段,一个标记 $startIndex$ 记录前一交点判断完毕后线段数组的索引位置。

依次从交点数组中取出一个交点。对于每个交点,首先更新 $segments$, 剔除里面不满足取舍条件的线段,然后依次判断该交点到 $segments$ 中各个线段的距离是否小于缓冲半径。如果小于,则剔除该交点, $startIndex$ 保持不变;否则,从 $startIndex$ 开始,从线段数组中依次取线段。如果线段满足取舍条件,则加入 $segments$ 中,并且判断该交点到线段的距离是否小于缓冲半径,如果小于,则剔除该交点;否则判断下一条线段,直到有一条线段满足点到该线段的距离小于缓冲半径(点在缓冲区内)或者线段的 $start.x \geq rightX$ (点在边界上)终止。 $startIndex$ 的值取剔除成功时的位置加 1 或剔除失败时的索引。

如图 6 所示,对于交点 1, $startIndex$ 为 0, $segments$ 没有元素。线段 01 不满足取舍条件,可以跳过。线段 12 满足取舍条件,加入 $segments$ 中,点到该线段的距离大于缓冲半径,则判断下一线段 54。线段 54 满足取舍条件,加入到 $segments$ 中,点到 54 的距离小于缓冲半径,则可以剔除该交点,更新 $startIndex$ 为 3,指向线段 56。

对于交点 2, 首先判断 $segments$ 中 12、54 是否满足取舍条件,然后计算点到 12、56 的距离。因为交点到 56 的距离小于缓冲半径,所以可以剔除该交点。假设 $segments$ 中没有线段满足交点到线段的距离小于缓冲半径,则从 $startIndex$ 开始,依次判断 56、34、32、67。如果其中一个满足判断条件,则剔除该点,如果都不满足,因为 78 的 $start.x$ 已经大于 $rightX$, 所以此时交点不在缓冲区内。 $startIndex$ 指向线段 78。如此重复,便可以剔除所有的交点。

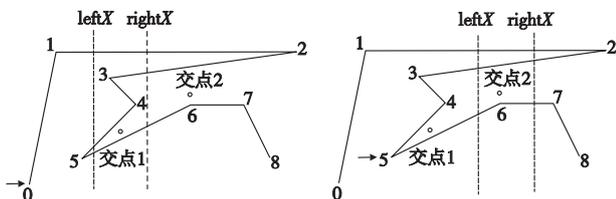


图6 剔除算法示例

2.4 跟踪生成缓冲区边界

经过计算交点并剔除在缓冲区内部的交点后。剩下的交点均位于缓冲区的边界上,为最终缓冲区上的点。此时构造缓冲区的策略借鉴采用两多边形求并的双线算法^[12]。其具体思想是:从交点集合中选择一点,沿着前进方向跟踪,遇到交点则跳转到交点所在的另外一个单元上,如此重复,直到回到该交点;如果还有剩余交点,则重复上述过程,直到所有的交点都参与构造缓冲区边界。此处的一个难点是,确定跟踪时选择的第一个交点是出点还是入点,可以根据叉积进行判断。这个过程中也一并处理了自相交问题。最终生成的缓冲区边界如图 7 所示。

3 实验结果与分析

本算法已用 ActionScript3.0 编程实现,在 Chrome 浏览器下运行测试。该算法能够避免失真问题,同时可自动处理自相交问题。图 8 展示了对 100 条随机线段生成的缓冲区效果。

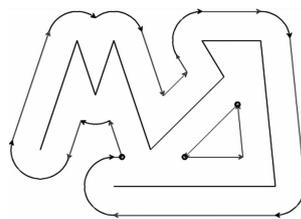


图7 最终缓冲区边界

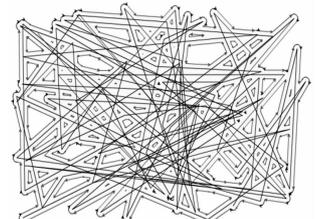


图8 最终缓冲区结果

本文对随机分布的 100 条线段 ($x[0 \sim 800], y[0 \sim 600]$, 缓冲半径为 10) 和沿 X 递增轴分布的 1000 条线段 (缓冲半径为 50) 进行测试,各个阶段的结果如表 1 和 2 所示。

表 1 折线线段集测试结果

各阶段结果	随机线段集	递增线段集
总计耗时/ms	137	49
构造初始边界耗时/ms	7	9
求交耗时/ms	58	32
交点总数/个	3 929	314
剔除非边界点耗时/ms	63	4
边界上的交点数/个	399	310
跟踪边界耗时/ms	7	3

表 2 不同缓冲半径计算耗时(100 条随机线段)

结果	缓冲半径					
	10	14	18	22	26	32
计算时间/ms	138	104	96	88	88	79

分析表 1 发现,整个缓冲区的计算效率和交点数有很大关系。当交点数较多时,不仅求交算法比较耗时,剔除非边界点同样耗时。因此对缓冲区效率的影响主要集中在求交算法和剔除算法上面。

分析表 2 发现,本算法具有一定的缓冲区自适应性。随着缓冲区的扩大,相交的可能性增大,但是会有不少线段/弧段位于缓冲区的内部,通过删除/更新这些单元以减少参与求交运算的线段/弧段,从而使缓冲区生成效率保持相对稳定。

4 结束语

本算法在传统的双线圆弧算法的基础上,通过减少参与求交的线段/弧段,改进平面扫描算法,设计高效的非边界点剔除算法和利用方向性质追踪生成缓冲区边界四个部分对文献^[7,11]提出的基于线段/弧段的平面扫描算法加以改进。(下转第 4389 页)

表 1 两种边缘函数下完成分割的迭代次数和运行时间统计

序列号	使用 g'_{Seq}		使用 g	
	平均迭代次数	总时间/s	平均迭代次数	总时间/s
1	850	283.54	1 100	363.28
2	1 000	330.36	1 450	528.10

1) 横向对比分析

a) 对于第一图像序列,使用本文边缘函数 g'_{Seq} ,单幅图像完成分割平均需要迭代 850 次,分割完成整个序列总用时为 283.54 s;而使用边缘函数 g 需要 1 100 次才能完成分割任务,序列总用时为 363.28 s。经计算,使用本文的边缘函数 g'_{Seq} 平均使每幅图像分割时间节省了 3.987 s。

b) 对于第二图像序列(含有较大噪声),使用本文边缘函数 g'_{Seq} 和原边缘函数 g ,单幅图像完成分割平均分别需要迭代 1 000 和 1 450 次,总耗时分别为 330.36 s 和 528.10 s,使用本文方法平均每幅图像分割时间节省了 9.887 s。

通过横向对比分析可以看出,使用本文的边缘函数 g'_{Seq} 进行医学图像序列分割时,分割效率明显高于边缘函数 g ,并且随着图像序列中噪声的增大,这种优势体现得更加明显。

2) 纵向对比分析

a) 使用本文边缘函数 g'_{Seq} 对两个序列的图像进行分割时,所需要的平均迭代次数分别为 850 和 1 000 次,分割总时间分别为 283.54 s 和 330.36 s,经计算可知噪声使分割总时间增加了 16.5%。

b) 使用函数 g 时,两个图像序列完成分割所需的平均迭代次数和时间分别为 1 100 次、363.28 s 和 1 450 次、528.10 s,计算得到总时间增加约 45%。

通过纵向对比分析可以看出,虽然噪声对两种边缘函数下的模型分割效率都产生了一定程度的影响,但相对而言,噪声对使用本文边缘函数 g'_{Seq} 模型的影响(16.5%)远小于传统的边缘函数 g (45%),说明本文用于医学图像序列分割的边缘函数能有效增强模型的抗噪声性能。

(上接第 4366 页)该算法能够避免失真问题、自动处理自相交问题,精确高效,具有较高的空间和时间利用率。若稍加改进,同样适用于构建单侧缓冲区、面状目标缓冲区和进行多目标缓冲区叠加。

理论和实验结果都表明,计算交点和剔除非边界点是最为耗时的两个步骤,后续研究将探求更为高效的算法,同时对基于椭球体的缓冲区构建算法作进一步探索。

参考文献:

- [1] 胡鹏,游连,杨传勇,等.地图代数[M].武汉:武汉大学出版社,2002:138-146.
- [2] 王结臣,沈定涛,陈焱明.基于栅格距离法的缓冲区生成与实现[J].科技通报,2009,25(5):556-561.
- [3] 王结臣,沈定涛,崔璨.缓冲区生成的游程刷叠置算法[J].武汉大学学报:信息科学版,2010,35(9):1121-1124.
- [4] 蓝秋萍,费立凡,李嘉.基于矢—栅混合数据的线目标缓冲区构建[J].测绘通报,2009(10):21-24.
- [5] 毋河海.关于 GIS 缓冲区的建立问题[J].武汉测绘科技大学学报,1997,22(4):358-366.
- [6] 董鹏,毛东军,李军,等.一种有效的 GIS 缓冲区生成算法[J].计算机工程与应用,2004,40(16):4-8,12.

综合以上纵向和横向两方面的对比分析可以看出,使用本文的边缘函数 g'_{Seq} ,无论是在分割的效率上还是在抗噪声性能上,较边缘函数 g 相比都有了很大的提高。

4 结束语

医学图像分割的准确性,直接影响到医护人员对病人病情的了解和判断。本文从边缘停止函数入手进行改进,将模型应用于医学图像序列分割,取得了较为理想的分割效果。下一步将主要从模型本身入手,进一步对 GAC 模型性能进行改进。

参考文献:

- [1] 李登旺,彭玉华,王洪君.医学图像配准和分割技术研究及在引导放射治疗系统中的应用[D].济南:山东大学,2011:18-19.
- [2] FARZINRAR M, ZHONG Xue, TEOH E K. A novel approach for curve evolution in segmentation of medical images[J]. Computerized Medical Imaging and Graphics, 2010, 34(5):354-361.
- [3] ZHANG Bo, SU Yong, XU Yong-fen, et al. An adaptive geodesic active contour model[C]//Proc of the 6th International Conference on Natural Computation. 2010.
- [4] CASELLES V, MOREL J M, SAPIRO G. Geodesic active contours[J]. International Computer Vision, 1997, 22(1):61-79.
- [5] 段俊杰,郭鹏江,佟建锋,等.新的噪声污染灰度图像边缘检测统计方法[J].计算机应用研究,2011,28(12):4780-4781.
- [6] STARING M, Van der HEIDE U S, KLEIN S, et al. Registration of cervical MRI using multifeature mutual information[J]. IEEE Trans on Medical Imaging, 2009, 28(9):1412-1421.
- [7] HAHN J, LEE C. Geometric attraction-driven flow for image segmentation and boundary detection[J]. Journal of Visual Communication and Image Representation, 2010, 21(1):55-66.
- [8] HE Ning, ZHANG Peng. Variational level set image segmentation method based on boundary and region information[J]. Acta Electronica Sinica, 2009, 37(10):2215-2219.
- [7] 朱焜,艾廷华,王洪.基于条带扫描思想的线目标缓冲区快速构建[J].测绘学报,2006,35(2):171-176.
- [8] 张宏,温永宁,刘爱丽,等.地理信息系统算法基础[M].北京:科学出版社,2006:185-200.
- [9] 白帆,张福浩.多目标缓冲区生成算法[J].全球定位系统,2011(1):38-41.
- [10] 彭李,何贞铭,刘少华,等.一种有效的线状缓冲区生成算法[J].测绘与空间地理信息,2008(5):60-62.
- [11] 李金山,方金云.基于平面扫描的双线圆弧缓冲区生成算法[J].计算机工程与应用,2007,43(23):28-31.
- [12] 周培德.计算几何—算法设计与应用[M].2版.北京:清华大学出版社,2005:217-223,239-241.
- [13] 邓俊辉.计算几何—算法与应用[M].北京:清华大学出版社,2008:23-37.
- [14] CHAZELLE B, EDELSBRUNNER H. An optimal algorithm for intersecting line segments in the plane[J]. Journal of the Association for Computing Machinery, 1992, 39(1):1-54.
- [15] SUNDAY D. The intersections for a set of 2D segments, and testing simple polygons[EB/OL]. http://softsurfer.com/Archive/algorithm_0108/algorithm_0108.htm.