



JOURNAL OF IMAGE AND GRAPHICS

主办: 中国科学院遥感与数字地球研究所
中国图象图形学学会
北京应用物理与计算数学研究所

中国图象图形学报

2016
01
VOL.21

ISSN1006-8961
CN11-3758/TB



医学图像处理 P69

中国图象图形学报

刊名题字：宋健 | 月刊（1996年创刊）



第21卷第1期（总第237期）
2016年1月16日

中国精品科技期刊
中国国际影响力优秀学术期刊
中国科技核心期刊
中文核心期刊

版权声明

凡向《中国图象图形学报》投稿，均视为同意在本刊网站及CNKI等全文数据库出版，所刊载论文已获得著作权人的授权。本刊所有图片均为非商业目的使用，所有内容，未经许可，不得转载或以其他方式使用。

Copyright

All rights reserved by Journal of Image and Graphics, Institute of Remote Sensing and Digital Earth, CAS. The content (including but not limited text, photo, etc) published in this journal is for non-commercial use.

主管单位 中国科学院
主办单位 中国科学院遥感与数字地球研究所
中国图象图形学会
北京应用物理与计算数学研究所

主 编 李小文
编辑出版 《中国图象图形学报》编辑出版委员会
邮政信箱 北京9718信箱
邮 编 100101
电子信箱 jig@radi.ac.cn
电 话 010-64807995
网 址 www.cjig.cn

广告经营许可证 京朝工商广字第0361号
总 发 行 北京报刊发行局
订 购 全国各地邮局
海外发行 中国国际图书贸易集团有限公司
(邮政信箱: 北京399信箱 邮编: 100048)
印刷装订 北京科信印刷有限公司

Journal of Image and Graphics

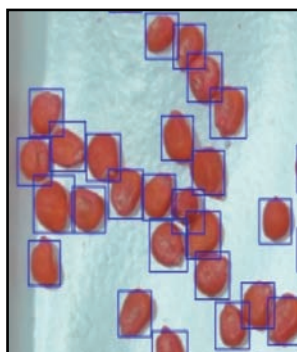
Title inscription: Song Jian | Monthly, Started in 1996

Superintended by Chinese Academy of Sciences
Sponsored by Institute of Remote Sensing and Digital Earth, CAS
China Society of Image and Graphics
Institute of Applied Physics and Computational Mathematics

Editor-in-Chief LI Xiaowen
Editor, Publisher Editorial and Publishing Board of Journal of Image and Graphics
P.O.Box 9718, Beijing, P.R.China
Zip code 100101
E-mail jig@radi.ac.cn
Telephone 010-64807995
Website www.cjig.cn

Distributed by Beijing Bureau for Distribution of Newspapers and Journals
Domestic All Local Post Offices in China
Overseas China International Book Trading Corporation
(P.O.Box 399, Beijing 100048, P.R.China)
Printed by Beijing Kexin Printing Co., Ltd.

CN 11-3758/TB
ISSN 1006-8961
CODEN ZTTXFZ
国外发行代号 M1406
国内邮发代号 82-831
国内定价 60.00元



多尺度特征融合与极限学习机的玉米种子检测
(第0024页)



深度卷积神经网络的显著性检测(第0053页)



双树复小波域的MRI图像去噪(第0104页)

图像处理和编码

空时形状预测与高效编码

朱仲杰, 王玉儿, 蒋刚毅..... 1

第二小方向导数信息熵的兴趣点检测

卢健, 黄杰, 潘峰..... 8

采用面积采样的反走样字符旋转算法

郑凯文, 刘文波..... 17

图像分析和识别

多尺度特征融合与极限学习机的玉米种子检测

柯道, 杜明智..... 24

图像理解和计算机视觉

结合NSCT和压缩感知的红外与可见光图像融合

陈木生..... 39

Plücker直线描述的空间后方交会

盛庆红, 陈姝文, 肖晖, 张斌, 王青, 费利佳..... 45

深度卷积神经网络的显著性检测

李岳云, 许悦雷, 马时平, 史鹤欢..... 53

计算机图形学

以优先点为中心的Delaunay三角网生长算法

尤磊, 唐守正, 宋新宇..... 60

医学图像处理

快速数字影像重建的2维/3维医学图像配准

刘坤, 吕晓琪, 谷宇, 于荷峰, 任国印, 张明..... 69

小波-Lagrange方法进行医学图像层间插值

武士想, 尚鹏, 王立功..... 78

遥感图像处理

多主体框架下结合最大期望值和遗传算法的SAR图像分割

张金静, 李玉, 赵泉华..... 86

第十届图像图形技术与应用学术会议

视差互信息引导下的立体航空影像与LiDAR点云自动配准

吴军, 胡彦君, 饶云, 彭智勇..... 95

双树复小波域的MRI图像去噪

黄学优, 张长江..... 104

快速定位图像尺度和区域的3维跟踪算法

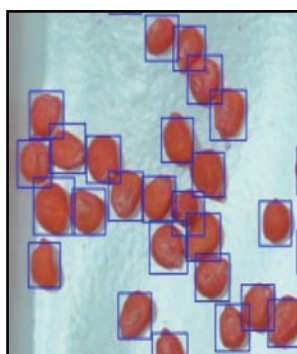
赵奇可, 孙延奎..... 114

局部不变特征点的精度指标

滕日, 周进, 蒋平, 崔雄文..... 122

CONTENTS

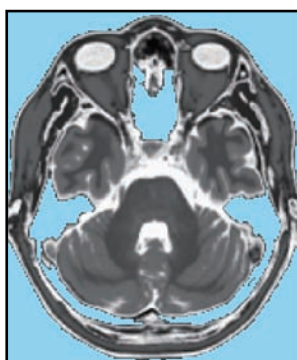
JOURNAL OF IMAGE AND GRAPHICS



Detection of maize seeds based on multi-scale feature fusion and extreme learning machine(P0024)



Saliency detection based on deep convolutional neural network(P0053)



MRI denoising based on dual-tree complex wavelet transform(P0104)

Image Processing and Coding

- Spatio-temporal shape prediction and efficient coding
Zhu Zhongjie, Wang Yuer, Jiang Gangyi 1
- Interest point detection by using information entropy of the second small direction derivative
Lu Jian, Huang Jie, Pan Feng 8
- Anti-aliasing algorithm for character rotation based on area sampling
Zheng Kaiwen, Liu Wenbo 17

Image Analysis and Recognition

- Detection of maize seeds based on multi-scale feature fusion and extreme learning machine
Ke Xiao, Du Mingzhi 24

Image Understanding and Computer Vision

- Image fusion of visual and infrared image based on NSCT and compressed sensing
Chen Musheng 39
- Space resection method based on Plücker line
Sheng Qinghong, Chen Shuwen, Xiao Hui, Zhang Bin, Wang Qing, Fei Lijiao 45
- Saliency detection based on deep convolutional neural network
Li Yueyun, Xu Yuelei, Ma Shiping, Shi Hehuan 53

Computer Graphics

- Growth algorithm centered on priority point for constructing the Delaunay triangulation
You Lei, Tang Shouzheng, Song Xinyu 60

Medical Image Processing

- The 2D/3D medical image registration algorithm based on rapid digital image reconstruction
Liu Kun, Lyu Xiaoqi, Gu Yu, Yu Hefeng, Ren Guoyin, Zhang Ming 69
- Inter-slice interpolation for medical images by using the wavelet-lagrange method
Wu Shixiang, Shang Peng, Wang Ligong 78

Remote Sensing Image Processing

- Combining the expectation maximization and genetic algorithms for SAR image segmentation within a MAS framework
Zhang Jinjing, Li Yu, Zhao Quanhua 86

Column of IGIA' 2015

- Automatic registration of aerial stereo imagery to LiDAR data through iterative parallax mutual information computation
Wu Jun, Hu Yanjun, Rao Yun, Peng Zhiyong 95
- MRI denoising based on dual-tree complex wavelet transform
Huang Xueyou, Zhang Changjiang 104
- Three dimensional tracking with fast locating of image scale and area
Zhao Qike, Sun Yankui 114
- Extraction precision of local invariant feature points
Teng Ri, Zhou Jin, Jiang Ping, Cui Xiongwen 122

中图法分类号: TP391 文献标识码: A 文章编号: 1006-8961(2016)01-0060-09

论文引用格式: You L, Tang S Z, Song X Y. Growth algorithm centered on priority point for constructing the Delaunay triangulation[J]. Journal of Image and Graphics, 2016, 21(1): 0060-0068. [尤磊, 唐守正, 宋新宇. 以优先点为中心的 Delaunay 三角网生长算法[J]. 中国图象图形学报, 2016, 21(1): 0060-0068.][DOI:10.11834/jig.20160108]

以优先点为中心的 Delaunay 三角网生长算法

尤磊^{1,2}, 唐守正¹, 宋新宇²

1. 中国林业科学研究院资源信息研究所, 北京 100091; 2. 信阳师范学院计算机与信息技术学院, 信阳 464000

摘要: **目的** Delaunay 三角网具备的优良性质使其得到广泛的应用, 构建 Delaunay 三角网是计算几何的基础问题之一, 为了高效、准确地构建大规模点集的 Delaunay 三角网, 提出一种基于优先点的改进三角网生长算法。**方法** 算法以逆时针次序的一条凸包边为初始基边, 使用基边对角最大化并按照逆时针次序选定第 3 点构建一个 Delaunay 三角形, 通过待扩展边列表中的数据判断新生成的两条边是否需要扩展, 采用先进先出的方式从待扩展边列表中取边作为基边, 以优先点为中心构建局部 Delaunay 三角网使优先点尽快成为封闭点, 再从点集中删除此封闭点。**结果** 对于同一测试点集, 改进算法运行时间与经典算法运行时间的比率不超过 1/3, 且此比率随点集规模增长逐步下降。相比经典算法, 改进算法在时间效率上有较大提升。**结论** 本文改进算法对点集规模具有较好的自适应性及较高的构网效率, 可用于大规模场景下 Delaunay 三角网的构建。

关键词: 计算几何; Delaunay 三角网; 生长算法; 先进先出; 封闭点; 优先点

Growth algorithm centered on priority point for constructing the Delaunay triangulation

You Lei^{1,2}, Tang Shouzheng¹, Song Xinyu²

1. Institute of Forest Resources and Information Techniques, Chinese Academy of Forestry, Beijing 100091, China;

2. College of Computer and Information Technology, Xinyang Normal University, Xinyang 464000, China

Abstract: Objective The Delaunay triangulation is a fundamental problem in the field of computational geometry. It is widely used because of its excellent characteristics. To construct a Delaunay triangulation network efficiently and accurately on a large-scale point set, an improved Delaunay triangulation algorithm based on priority point is presented in this paper.

Method An initial base edge is selected from the convex hull edges in anticlockwise order. A Delaunay triangle is constructed by base edge and the third point in the anticlockwise order, which maximizes the angle opposite the base edge. Whether the generated two edges need to be expanded is determined by the array of the needed expanded edges. The first-in-first-out strategy is used to extract the base edge from the array of the needed expanded edges. The local Delaunay triangulation is constructed around the priority point and accelerates the priority point to become a closed-point. Then, the closed-point is

收稿日期: 2015-08-02; 修回日期: 2015-10-16

基金项目: 国家高技术研究发展计划(863)基金项目(2012AA102002); 国家自然科学基金项目(61202194, 31470641, 61572417, 11501489); 河南省省院科技合作专项资金项目(122106000052)

第一作者简介: 尤磊(1981—), 男, 讲师, 现为中国林业科学研究院资源信息研究所森林经理专业在读博士研究生, 主要研究方向为点云数据处理、数据挖掘。E-mail: yleiou@163.com

通信作者: 唐守正, 中国科学院院士, 研究员, E-mail: stang@caf.ac.cn

Supported by: National High Technology Research and Development Program of China (122106000052); National Natural Science Foundation of China (61202194, 31470641, 61572417, 11501489)

removed. **Result** The running time ratio of improved algorithm to classical algorithm is less than a third when using the same point set, and the ratio gradually decreases with increasing point set scale. Compared with the classical algorithm, the improved algorithm has significant enhancement in time efficiency. **Conclusion** The improved algorithm has better adaptability for point set scale and high efficiency for constructing Delaunay triangulation. It can also be used for large-scale point set Delaunay triangulation.

Key words: computation geometry; Delaunay triangulation; growth algorithm; first in first out; closed-point; priority point

0 引言

平面点集的 Delaunay 三角剖分是计算几何中的一个基本问题,其具有的最小角最大化特性与空外接圆特性保证了2维点集三角剖分的唯一性与合理性。这使得 Delaunay 三角剖分在图像表面重建、数字地形模型、有限元分析等领域得到广泛应用。随着应用领域的延伸与数据规模的增长,高效、准确、稳定地构建大规模点集的 Delaunay 三角网已逐渐成为一个研究热点^[1]。

Delaunay 三角网构建算法可分为3个大类:分治—合并法、逐点插入法、三角网生长法。这3类算法各有其优缺点。文献[1-2]分析比较了3类算法的优缺点,文献[1]评价了几类代表性算法的性能,分治—合并算法效率最高;文献[2]总结了代表性算法的时间复杂度。概括而言,合并—分治算法采用递归的方法将点集逐层划分为多个子集,分别在子集上构建三角网,然后逐层向上合并。此类算法时间效率高,但受系统进程栈空间的限制,进程的递归深度有限,难以将此类算法应用到大规模点集的三角网构建中^[1]。逐点插入法需要在三角网中搜索包含新加入点的三角形,然后生成新的三角形并优化三角网。在大规模点集中使用此类算法需要频繁地遍历三角网中的三角形以确定点的位置,时间复杂度较高^[3]。三角网生长算法以基边为基础,通过在点集中搜索第3点,使其与基边构成一个新的 Delaunay 三角形。此类算法看似复杂度较高,但此类算法确能充分发挥 Delaunay 三角网局部性的特征,对基边而言,基边扩展得到 Delaunay 三角形必定是 Delaunay 三角网中的三角形,在后续构网中无需调整。对点而言,若此点的 Delaunay 三角形构造完毕,则在后续构网中无需考虑此点。

基于此,在分析三角网生长法及其改进算法的基础上,提出了一种简单高效的基于优先点的三角

网生长算法,此生长算法充分发挥 Delaunay 三角网局部性的特征,对大规模点集有较好的自适应性。其优势主要体现在以下4个方面:1)高效判断三角形是否重复;2)先进先出的优化取边策略;3)以优先点为中心的局部构网策略;4)逐渐减小待构网的点集规模。

1 经典 Delaunay 三角网生长算法

三角网生长算法^[4]最早由 Green 和 Sibson 提出,随后众多学者都在此基础上提出了改进算法^[5-10]。三角网生长算法及其改进算法本文称之为经典的三角网生长算法。

对于任意4点不共圆的一个平面点集 P , P_i 表示点集 P 的第 i 个点。构建其 Delaunay 三角网的基本步骤如下^[1,3]:

- 1) 初始化待扩展边列表 E 与三角形列表 T ;
- 2) 任取点集 P 中一点 P_i 与其最近邻点 P_j 构成基边 $\overline{P_i P_j}$;
- 3) 在点集中寻找第3点 P_k , 使得三角形 $\triangle P_i P_j P_k$ 构成一个 Delaunay 三角形;
- 4) 将 $\triangle P_i P_j P_k$ 的另外两条边 $\overline{P_i P_k}$ 与 $\overline{P_j P_k}$ 加入到待扩展边列表 E 中;将三角形 $\triangle P_i P_j P_k$ 加入到三角形列表 T 中;
- 5) 如果待扩展边列表 E 不为空,取其中的一条边为基边转到步骤3)继续执行。如果待扩展边列表 E 为空,算法退出执行。

生长算法的核心思想是根据基边寻找第3点 P_k , 使基边的两个顶点与第3点 P_k 形成一个新的 Delaunay 三角形。目前提高 Delaunay 三角网生长算法的运行效率主要从以下两个方面入手:

- 1) 提高搜索第3点的效率;
- 2) 提高判断第3点是否符合条件的效率。

提高搜索第3点的效率包括:采用正负区^[5]搜索第3点,每次只需要在基边的另一侧搜索第3点。

使用封闭点^[6]的概念在构网过程中及时剔除已经完成 Delaunay 三角形构建的节点。提高判断第 3 点是否符合条件的效率有:采用基边对角最大化^[7]判定第 3 点。

在三角网生长算法中有一个必需的步骤:如何确保三角网中不出现重复的三角形。然而,少有文献讨论此步骤的计算过程及其时间复杂度。本文从分析此步的计算过程入手,逐步深入,对三角网生长算法的几个主要步骤提出改进方法或简化策略。

2 算法分析与改进

2.1 三角形重复性的优化判断

在生长算法的扩展过程中,5)步骤是从待扩展边列表中取其中一条边作为基边进行下一步扩展的步骤,此取边步骤在算法实现上有两种取边策略:

- 1) 后进先出,后进入到待扩展边列表中的边优先作为基边扩展三角形;
- 2) 先进先出,先进入到待扩展边列表中的边优先作为基边扩展三角形。

按照这两种取边策略构建三角网都会产生重复三角形的问题,如图 1 所示。

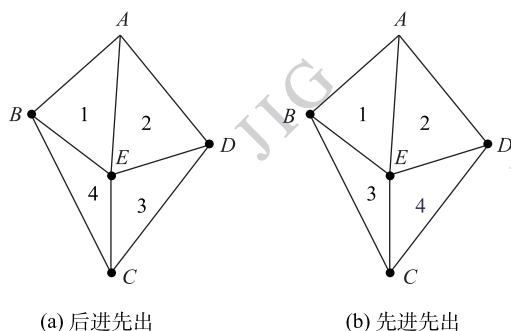


图 1 重复三角形示例

Fig. 1 Repeated triangle figure((a) last in first out;
(b) first in first out)

图 1 中,共有 5 个点,分别用 A 、 B 、 C 、 D 、 E 表示。图 1 中数字是三角形的构建次序, AB 表示节点 A 与 B 之间形成的一条无向边。无向边 AB 是初始基边,扩展 $\triangle ABE$ 后将边 AE 、 BE 插入到待扩展的边列表中。图 1(a)是后进先出策略,扩展边 AE 、 DE 、 EC 分别得到 $\triangle AED$ 、 $\triangle EDC$ 与 $\triangle BEC$,此时待拓展边中包含边 BE ,若继续扩展此边则会再次生成 $\triangle BEC$,若

不加以处理就会在三角网中出现重复的三角形。同理先进先出的取边策略也存在这样的问题。

为避免出现重复三角形,有以下两种较直接的方法:1)遍历当前生成的三角形在三角网中是否存在;2)在三角网中统计边出现的次数以判断边是否需要扩展。这两种方法都是基于搜索的算法,其最优时间复杂度为 $O(n \lg n)$,其中前者的 n 是三角网中三角形的个数。显然,当点集规模不大时,选择这两种方法中的任何一个对算法运行时间的影响都不大。但当点集规模较大时,则必须考虑此步对算法运行时间的影响。

对 Delaunay 三角网分析易知,Delauany 三角网的边界边是点集凸包中相邻的两个凸包点形成的边,且边界边只能形成一个三角形。三角网中除边界边外的边都是内部边,内部边都必须形成两个三角形。考虑到待扩展边列表中的边都是在扩展一个三角形后而生成的,即在待扩展边列表中的边都已经生成过一个三角形。所以,待扩展边列表中的边至多只能被扩展一次再形成一个三角形。显然,若待扩展边是边界边,此边已无法再被扩展;若待扩展边是内部边,则此边只能被扩展一次。

由此,提出一种三角形重复的优化判断方法:算法以点集的某一条凸包边作为起始基边扩展三角网,对于新生成的三角形引出的两条待扩展边的每一条边,若此边是凸包边,则无需扩展;若不是凸包边,且在待扩展边列表中存在,则从待扩展边列表中删除此边,若不在待扩展边列表中存在,则将此边添加到待扩展边列表中。

以点集的凸包边作为起始基边,是确保起始基边不会在待扩展边列表中出现,从而保证不会再次从起始基边扩展三角形。而经典算法中以任意点及其最近点构成起始基边扩展三角网的方法容易造成再次从基边扩展三角形。

2.2 两种不同的取边策略

从待扩展边列表取边有两种方法,取边的方法会影响待扩展边列表的最大长度。图 2 是不同取边策略下三角形的生成次序图。图 2 中数字表示此三角形的生成次序。

从图 2 不难看出,若采用后进先出的方式,三角形扩展是按照外圈三角形向内圈三角形逐圈扩展的次序进行;若采用先进先出的策略,则三角形扩展是以螺旋形式逐步扩大的方式进行。

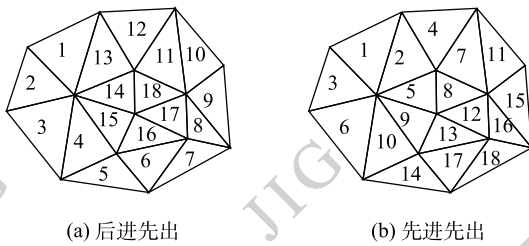


图2 不同取边策略构网次序图

Fig. 2 The order figure of different getting edge strategy

((a) last in first out; (b) first in first out)

显然,采用后进先出的取边策略所产生的待扩展边列表的最大长度大于采用先进先出的取边策略所产生的待扩展边列表的最大长度。这可以从以下两个方面论证:1)从内部边的角度。当采用后进先出的取边策略时,最外圈的三角形都生成完成后才会与第1个生成的三角形相遇,此时才会有内部边形成三角形个数等于2,才开始从待扩展边列表中删除边;而采用先进先出的策略,相互邻近的三角形会先后生成,很快就有多个内部边形成的三角形个数等于2,很快就有边要从待扩展边列表中删除。图2(a)中生成第13个三角形,才有一个内部边满足不需要扩展条件;而图2(b)中生成第5个三角形时就开始有一个内部边满足不需要扩展条件。2)从点的角度。采用先进先出策略能更快地构建点的局部 Delaunay 三角网。

2.3 以优先点为中心的构建策略

结合文献[6]对封闭点的描述,首先给出封闭点的定义。

点夹角:若点集中的一个点 P_j , $\triangle P_i P_j P_k$ 是点 P_j 形成的一个 Delaunay 三角形, $\angle P_i P_j P_k$ 称为点 P_j 在 $\triangle P_i P_j P_k$ 的点夹角。

封闭点:对于点集中的一个非凸包点,若该点在其所形成的 Delaunay 三角形的所有点夹角的和等于 360° ,则点为封闭点。

图3中点 F 是封闭点的一个例子。在三角网生长算法中,若一个是封闭点,则在后续的三角网构建过程中,该点不会也不可能再派生出其他边。这意味着删除此点并不影响后续的构网结果,但却会减少后续构网中需要遍历的节点数量。从而在保证构网无误的情况下,能提高算法的运行效率。

根据2.2节,采用先进先出的取边策略能更快地形成一些封闭点。基于此,在文献[6]的基

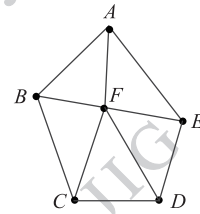


图3 封闭点示例

Fig. 3 Closed-point figure

础上更深一步,提出以优先点为中心的构网策略。

优先点:点集中的一个顶点 P_i ,若在建网过程中,优先扩展以 P_i 为顶点的边,使 P_i 成为封闭点后,再扩展其他顶点不为 P_i 的边,则此点称为优先点。

具体方法是:向三角网添加一个新扩展生成的三角形时,分别计算三角形的3个顶点在此三角形内的内角,并计入到各顶点的夹角和中;分别判断3个点是否是封闭点,若已经是封闭点,则从点集中删除此点;然后判断是否存在优先点,若存在,就将新生成的两条待扩展边中包含优先点的边插入到待扩展边列表的首部,若不存在优先点,则以这两边中的任意一条边的一个顶点为优先点,将此边插入到待扩展边列表的首部,再将待扩展边列表中包含优先点的边前移到首部之后。然后转入下一步,采用先进先出的策略从待扩展边列表中取边继续扩展三角网。

通过以优先点为中心的构网策略,能更快地将优先点变为封闭点,从而能尽快地将此点从点集中删除,减少了后续构网的点集数量。

2.4 正负区的简化处理

文献[5]中使用正负区来搜索第3点是基于以 Delaunay 三角形的一边扩展得到的新的 Delaunay 三角形必然位于原三角形的另外一侧,如图3中,若以 $\triangle ABF$ 的边 BF 扩展,则 $\triangle ABF$ 与扩展边 BF 得到的三角形 BCF 必定位于边 BF 的两侧。即在扩展边 BF 时,只需要在边 BF 的另外一侧搜索第3点。

为简化问题,采用逆时针构网策略:三角网中三角形的3个顶点的排列与待扩展边中两个顶点的排列都按照顶点在坐标平面上的逆时针次序排列。对于按照逆时针次序排列的点 P_i 与 P_j ,其形成有向边用 $\overrightarrow{P_i P_j}$ 表示。

因此,若 $\triangle P_i P_j P_k$ 是扩展边有向边 $\overrightarrow{P_i P_j}$ 得到的三角形,考虑到正负区的选点方法,则应加入到扩展边列表的两条边应该是 $\triangle P_i P_j P_k$ 的两条有向边 $\overrightarrow{P_j P_k}$ 、 $\overrightarrow{P_k P_i}$ 的逆向边 $\overrightarrow{P_k P_j}$ 与 $\overrightarrow{P_i P_k}$ 。如图4所示。

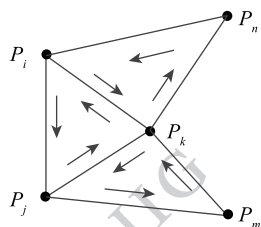


图4 逆时针次序构建 Delaunay 三角网

Fig. 4 The Delaunay triangulation constructed in anticlockwise order

若点 $P_i(x_i, y_i)$ 、 $P_j(x_j, y_j)$ 与点 $P_k(x_k, y_k)$ 3 点不共线, 且是坐标平面上逆时针次序的一个排列, 则 3 点坐标满足不等式

$$f(P_i, P_j, P_k) = \begin{vmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{vmatrix} > 0 \quad (1)$$

显然, 对于给定的有向边 $\overrightarrow{P_i P_j}$, 若有点 $P_k(x_k, y_k)$ 满足式(1), 则点 P_k 就是有向边 $\overrightarrow{P_i P_j}$ 另外一侧的一个点, 即是潜在的第 3 点。

通过此简化过程, 一方面明确了三角网中三角形顶点的次序, 另一方面明确了待扩展边列表中边的方向, 进而明确了待扩展边的扩展方向。

2.5 基边扩展第三点判定的简化准则

在给定基边的情况下, 若采用 Delaunay 三角形的定义, 三角形 3 个顶点的外接圆内无其他点的方法来判断, 需要构建以基边的两个顶点与第 3 点为圆周点的外接圆, 再判断点集中的其他点是否在圆内, 其时间复杂度为 $O(n^3)$ 。文献[7]中使用基边对角最大化作为选定第 3 点的条件。即对于给定的基边 $\overrightarrow{P_i P_j}$, 从待选点集中选定一点 P_k , 使得有向角 $\angle P_i P_k P_j$ 的角度最大。这种判断方法的最差时间复杂度为 $O(n^2)$ 。显然使用基本对角最大化的选定方法时间效率高。

根据 2.4 节的分析, 第 3 点 P_k 应满足式(1), 其夹角 $\angle P_i P_k P_j$ 的取值范围为 $(0, \pi)$, 则有

$$\cos(\angle P_i P_k P_j) \in (-1, 1) \quad (2)$$

因 $\angle P_i P_k P_j$ 角度值越大, $\cos(\angle P_i P_k P_j)$ 值越小。可使用 $\cos(\angle P_i P_k P_j)$ 值作为第 3 点的选定准则。 $\cos(\angle P_i P_k P_j)$ 的值为

$$\cos(\angle P_i P_k P_j) = \frac{((x_i - x_k)(x_j - x_k) + (y_i - y_k)(y_j - y_k))}{\sqrt{((x_i - x_k)^2 + (y_i - y_k)^2)((x_j - x_k)^2 + (y_j - y_k)^2)}}$$

$$\sqrt{((x_j - x_k)^2 + (y_j - y_k)^2)} \quad (3)$$

式中, $\sqrt{x}$ 是 x 的正平方根。据此, 给出基边扩展第 3 点的一个简化准则: 对于给定的基边 $\overrightarrow{P_i P_j}$, 从待选点集中选定一点 P_k , 满足式(1)且满足 $\cos(\angle P_i P_k P_j)$ 的值最小, 则基边 $\overrightarrow{P_i P_j}$ 与点 P_k 组成的 $\triangle P_i P_j P_k$ 就是基边 $\overrightarrow{P_i P_j}$ 扩展的 Delaunay 三角形。

考虑到算法采用逆时针次序构建三角网, 为确保初始基边能扩展成功, 将初始基边选定为逆时针次序的两个相邻凸包点形成的边。显然, 此基边必定能找到一个能与其形成 Delaunay 三角形的第 3 点。

3 基于优先点的三角网生长算法

基于以上分析, 提出一种基于优先点的 Delaunay 三角网生长算法, 首先给出算法的数据结构。

1) 三角形结构体

```
struct Triangle
```

```
{
    int FirstPointIndex;
    int SecondPointIndex;
    int ThirdPointIndex;
};
```

2) 待扩展边结构体

```
struct NeedExpandEdge
```

```
{
    int FirstPointIndex;
    int SecondPointIndex;
};
```

3) 点夹角和结构体

```
struct PointAngleSum
```

```
{
    int PointIndex;
    float AngleSum;
};
```

4) 点结构

```
struct Point
```

```
{
    int GlobalIndex;
    float x;
    float y;
};
```

在构网过程中, 删除封闭点会使点在点集中的

索引 Index 发生变化, 因此加入用于记录点在全局点集中的索引字段 GlobalIndex, 确保从点集中删除封闭点后依然能获取点的全局索引。

为便于算法描述, 将算法中使用的数组定义如下: T 表示三角形数组、 N 表示待扩展边数组、 S 表示点夹角和数组、 H 表示凸包点数组。对于任意 4 点不共圆的一个平面点集 P , P_i 表示点集 P 的第 i 个点, 基于优先点的 Delaunay 三角网生长算法的步骤如下:

- 1) 初始化三角形数组 T 、待扩展边数组 N 、点夹角和数组 S 、凸包点数组 H ;
- 2) 按照逆时针次序计算点集的凸包点(凸包点形成一个闭包集), 并存储到 H 数组中;
- 3) 从 H 中任意取相邻的两点 $P_i (P_i = H_k)$ 与 $P_j (P_j = H_{k+1})$, 构成一条初始基边 $\overrightarrow{P_i P_j}$;
- 4) 使用基边扩展第 3 点判定的简化准则, 以 $\overrightarrow{P_i P_j}$ 为基边, 从点集中找到一点 P_k , 构建 Delaunay 三角形 $\triangle P_i P_j P_k$;
- 5) 将 $\triangle P_i P_j P_k$ 加入到 T ;
- 6) 分别计算 $\triangle P_i P_j P_k$ 中三点的点夹角, 并与 S 数组中与各点对应的角度和相加后存储到各点对应的位置; 判断点 P_i 的点夹角和是否等于 360° , 若等于 360° , 则从点集 P 中删除点 P_i , 同时删除 N 中所有包含点 P_i 的待扩展边, 且边 $\overrightarrow{P_i P_j}$ 无需扩展; 同理, 对点 P_j 与点 P_k 执行同样的操作;
- 7) 根据步骤 6), 若 $\overrightarrow{P_i P_j}$ 需要继续扩展, 则对边 $\overrightarrow{P_i P_j}$ 做如下操作: 根据 H 中元素判断 $\overrightarrow{P_i P_j}$ 是否是凸包边, 若是凸包边则无需扩展; 若不是凸包边则进一步判断其逆边 $\overrightarrow{P_j P_i}$ 是否在 N 中存在, 若存在则从 N 中删除 $\overrightarrow{P_j P_i}$; 若不存在, 转到步骤 8);
- 8) 若存在优先点, 判断边 $\overrightarrow{P_i P_j}$ 的顶点是否包含优先点, 若包含, 将边插入到待扩展边列表的首部, 若不包含, 将边插入到待扩展边列表的尾部, 若不存在优先点, 以 $\overrightarrow{P_i P_j}$ 中的一个点为优先点, 将待扩展边列表中顶点为优先点的边前移至首部。然后将边 $\overrightarrow{P_i P_j}$ 插入到待扩展边列表的首部。同理, 若边 $\overrightarrow{P_i P_k}$ 需要继续扩展, 则按照同样的步骤执行;
- 9) 如果 N 不为空, 采用先进先出的策略从 N 中取第 1 条边, 转向步骤 4); 如果待扩展边列表 N 为空, 算法执行完毕。

算法流程图如图 5 所示。

4 算法复杂度分析

4.1 时间复杂度

以访问点的操作作为算法的基本运算。根据欧拉公式 $V + F - E = 2$ (V 是顶点个数, F 是面(包括无穷面)的个数, E 是边的个数) 与文献[10]可知, Delaunay 三角网中的点集个数、边的个数与三角形的个数三者之间都是线性关系。从而待扩展边列表的长度与点集个数也是线性关系, 事实上, 采用先进先出的取边策略, 待扩展边列表的最大长度远小于点集的个数。点的夹角和数组采用空间换时间的思路, 按照点集的全局索引顺序存储访问。从而算法的时间复杂度主要取决于构造点集的凸包上与使用第 3 点选定准则选定第 3 点上。

文献[11]已证明凸包运算的时间复杂度与排序的时间复杂度等价, 其最优时间复杂度为 $O(n \lg n)$ 。

算法在给定基边搜索第 3 点时, 需遍历点集中所有点。此步的时间复杂度在最差情况下的时间复杂度是 $O(n^2)$ 。但根据第 3 点判定准则, 基边对角的余弦值最小的点就是所选, 最优状态下点集是基边对角余弦的有序排列, 此时选定第 3 点的复杂度是 $O(n)$ 。但此步需先排序, 排序的最优时间复杂度是 $O(n \lg n)$, 即此步的最优时间复杂度是 $O(n \lg n)$ 。

算法在运行过程中, 待构网点集的规模会随着封闭点的删除而逐渐减小, 故基于优先点的改进算法的最优时间复杂度的上界是 $O(n \lg n)$, 最差的时间复杂度的上界是 $O(n^2)$ 。

4.2 空间复杂度

根据算法描述可知, 在算法运行过程中需要 3 个辅助数组: 三角形结构体数组、待扩展边数组及点夹角和数组。根据 4.1 节的分析可知, 上述 3 个数组与点集个数都是线性关系, 故算法的空间复杂度是 $O(n)$ 。

5 实验

在型号 ThinkCenterM8500t、内存 8 GB、处理器 I7-4770 的台式机上, 采用 VC++2010 作为开发环境, 采用地面 3 维激光扫描仪获取的点云数据在地

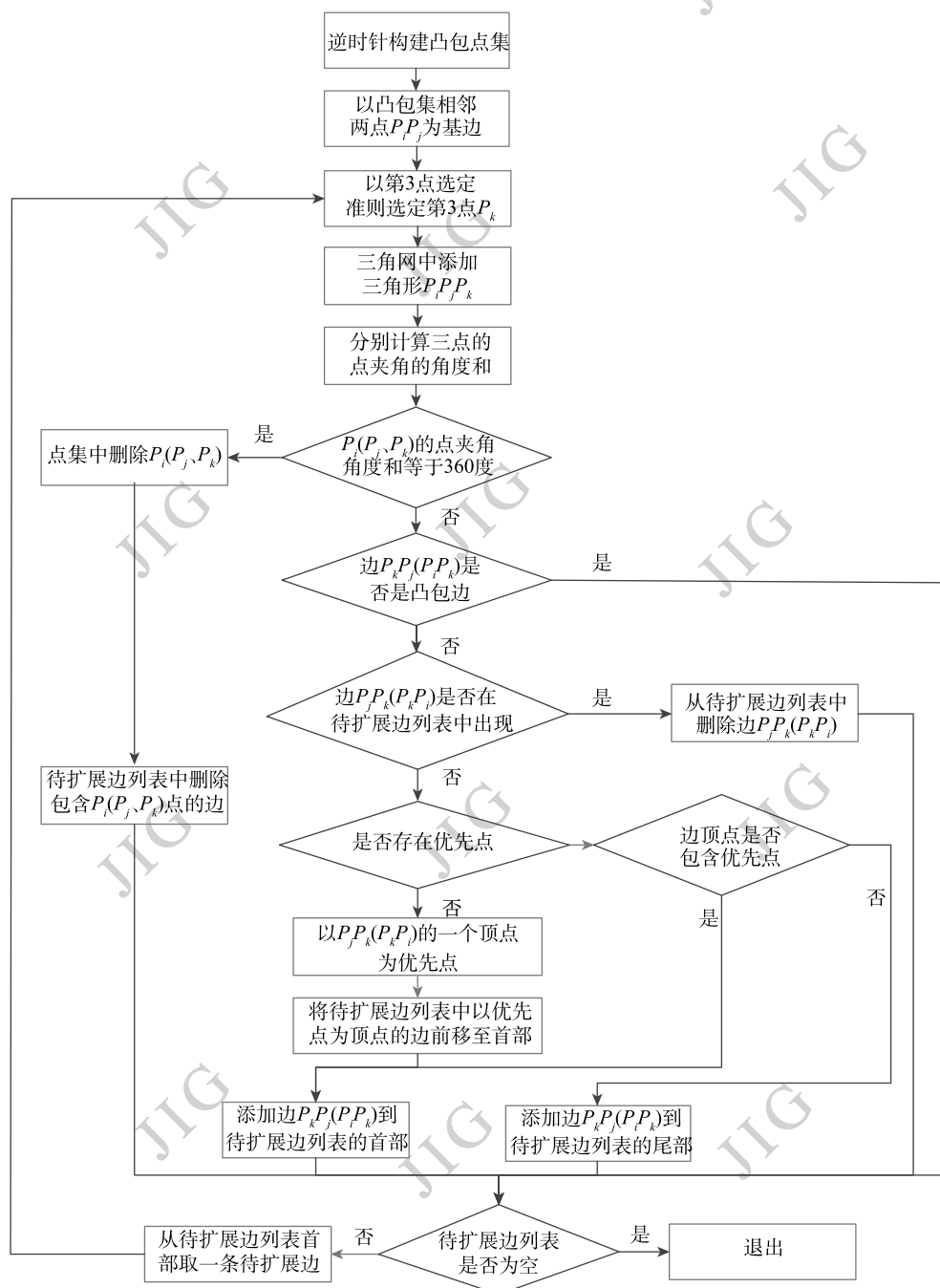


图5 基于优先点的三角网生长算法流程图

Fig. 5 The algorithm flowchart of algorithm based on priority point

平面上的投影为实验点集。分别以点集中的 5 000 个、10 000 个、20 000 个点为测试点集。

采用 3 种算法分别开展实验,算法 A 任取一点及其最近点为基边,从三角网中判断边是否需要扩展以避免出现重复三角形,可认为是经典算法;算法 B 以逆时针的凸包边为基边,使用本文提出的三角形重复性的优化判断方法;算法 C 在算法 B 的基础上再采用优先点为中心的构网策略,即本文提出的基于优先点的生长算法。3 种算法都采用正负区的

简化处理;按照逆时针构造三角网;采用基边扩展第 3 点判定的简化准则扩展第 3 点;采用先进先出的取边策略。

根据计算精度的需要,在 VC++2010 开发环境下,改进算法的数值运算都使用 double 数据类型。分别记录每次实验的待扩展边列表的点集个数 n 、三角网中三角形的个数 n_{tri} 、待扩展边列表的最大长度 k 及运行时间 t 。实验结果如表 1 所示。3 种算法在不同点集规模下的运行时间对照图如图 6 所示。

表 1 算法运行时间比较结果

Table 1 The running time comparison of algorithms

n	n_{tri}	算法 A		算法 B		算法 C		
		k	t/ms	k	t/ms	k	t/ms	Rate
5 000	9 964	483	23 230	274	911 9	321	7 584	0.326
10 000	19 965	629	91 868	308	35 544	489	27 712	0.301
15 000	29 958	788	205 506	594	79 695	631	60 687	0.295
20 000	39 955	837	358 872	539	132 915	545	98 479	0.274

注:Rate 表示在同一测试点集下算法 A 的运行时间与算法 C 的运行时间的比率。

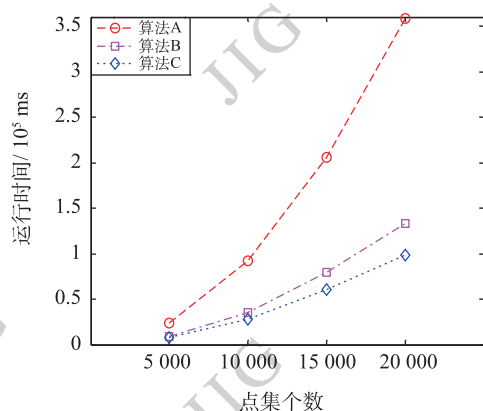


图 6 3 种算法不同点集规模运行时间对比图

Fig. 6 Comparison figure of algorithms of different data scale

从表 1 运行数据可以看出,对于同一测试点集,算法 C 的运行时间与算法 A 运行时间的比率不超过 1/3,且随着点集规模的增长,比率有逐渐下降的趋势。从图 6 容易看出,算法 C 的运行时间随点集规模增长最为缓慢,与算法时间复杂度的分析结果一致,在时间效率上有较好的表现。在待扩展边列表的长度上,算法 B 有较好的表现,这与算法 B 以三角形为中心,能尽快地完成当前选定三角形的与其共边的其他三角形的构建,所有待扩展边列表中的边能尽快完成三角网的构建;而算法 C 以点为中心,优先完成当前选定点的三角形构建,所以待扩展边列表长度稍长,但算法 C 能更快地减少点集的数据规模,从而有了更好的时间效率。

算法 C 构建的三角网部分效果如图 7 所示。

6 结 论

三角网生长算法是构造 Delaunay 三角网的 3

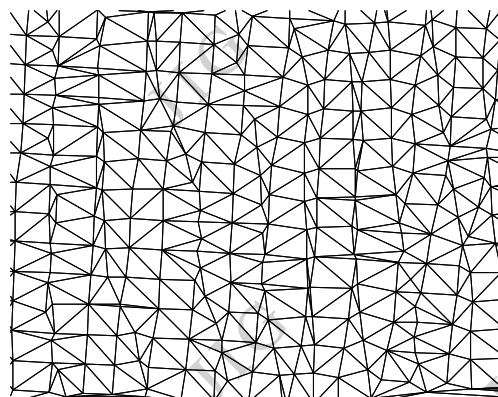


图 7 算法 C 构建的三角网部分效果图

Fig. 7 The partial triangulation constructed by algorithm C

类经典算法之一。本文在研究三角网生长算法的基础上,从分析三角网生长算法中必须解决的问题——判断生成的三角形是否有重复入手,提出了一种简单有效的判断三角形是否重复的方法;继而分析了从待扩展边列表中取边方法对构网顺序及时间复杂度的影响;再从点的角度进一步分析先进先出的取边策略对完成局部 Delaunay 三角网的促进作用,提出了以优先点为中心的构建策略;并对生长算法在第 3 点的选择与判定上给出了简化准则。在综合以上分析的基础上,提出了基于优先点的 Delaunay 三角网生长算法;并分析了算法的时间与空间复杂度;理论分析与实验,结果表明了算法对点集规模的自适应性及构网的高效性,适用于大规模 2 维点集下 Delaunay 三角网的构建。

改进算法在构网过程中需一次性加载点集中所有点。然而,当点集规模大到计算机内存无法一次性加载所有点时,如大数据场景下的 Delaunay 三角网构建,结合改进算法优化构建以优先点为顶点的 Delaunay 三角形的特点,及 Delaunay 三角网全局唯

一性与区域性的特点,研究在大数据场景下改进算法的并行算法或以优先点为中心边加载边构网的改进算法是下一步的研究内容。

参考文献 (References)

- [1] Yu J, Lyu P, Zheng C W. A comparative research on methods of Delaunay triangulation [J]. Journal of Image and Graphics, 2010, 15(8): 1158-1167. [余杰, 吕品, 郑昌文. Delaunay 三角网构建方法比较研究[J]. 中国图象图形学报, 2010, 15(8): 1158-1167.] [DOI:10.11834/jig.20100804]
- [2] Peter S P, Drysdale R L S. A comparison of sequential Delaunay triangulation algorithms [C]//The Eleventh Annual Symposium on Computational Geometry. Vancouver, British Columbia, Canada: ACM, 1996:61-70. [DOI:10.1145/220279.220286]
- [3] Wu X B, Wang H X, Xiao C S. A new study of delaunay triangulation creation [J]. Acta Geodaetica et Cartographica Sinica. 1999, 28(1):28-35. [武晓波, 王世新, 肖春生. Delaunay 三角网的生成算法研究[J]. 测绘学报, 1999, 28(1):28-35.]
- [4] Green P J, Sibson R. Computing dirichlet tessellations in the plane [J]. The Computer Journal, 1978, 1(2):168-173. [DOI: 10.1093/comjnl/21.2.168]
- [5] He Q B, Li G Y, Wen J, et al. An improved algorithm for building Delaunay triangulation [J]. Computer & Digital Engineering, 2006, 34(5):50-20. [贺全兵, 黎贵友, 文进, 等. 生成 Delaunay 三角网的改进算法[J]. 计算机与数字工程, 2006, 34(5): 50-52]
- [6] Ling H B, Wu B, et al. An improved algorithm for auto-connected Delaunay triangulation [J]. Computer Applications, 1999, 19(12):10-12. [凌海滨, 吴兵. 改进的自连接 Delaunay 三角网生成算法[J]. 计算机应用, 1999, 19(12):10-12.]
- [7] Maus A. Delaunay triangulation and the convex hull of n points in expected linear time [J]. Bit Numerical Mathematics, 1984, 24(2):151-163. [DOI:10.1007/BF01937482]
- [8] Brassel K E, Reif D. A procedure to generate thiesse polygons [J]. Geographical Analysis, 1979, 11(3):289-303. [DOI: 10.1111/j.1538-4632.1979.tb00695.x]
- [9] Zhang X C, Huang Z C, Dai Q C, et al. An algorithm of speedily building tin based on topological structure and convex shell in GIS [J]. Chinese Journal of Computers, 2002, 25(11): 1212-1218. [章孝灿, 黄智才, 潘云鹤, 等. GIS 中基于拓扑结构和凸壳技术的快速 TIN 生成算法[J]. 计算机学报, 2002, 25(11):1212-1218]
- [10] LEE D T, Schachter B J. Two algorithms for constructing a Delaunay triangulation [J]. International Journal of Computer & Information Sciences, 1980, 9(3):219-242. [DOI:10.1007/BF00977785]
- [11] De Berg M, Van Kreveld M, Overmars M, et al Computational Geometry: Algorithms And Applications [M]. 3rd ed. New York:Springer-Cerlag, 2009:2-10. [De Berg M, Van Kreveld M, Overmars M, et al. 计算几何:算法与应用 [M]. 邓俊辉, 译. 第3版. 北京:清华大学出版社, 2009:2-10.]
- [12] Chen Z G, Cao J, Yang C H. Topology improvement for constructing optimal Delaunay triangulation [J]. Journal of Computer-Aided Design & Computer Graphics, 2011, 23(12): 1967-1974. [陈中贵, 曹娟, 杨晨晖. 构造最优 Delaunay 三角剖分的拓扑优化方法[J]. 计算机辅助设计与图形学学报, 2011, 23(12):1967-1974.]
- [13] Kolingerová I, žalik B. Improvements to randomized incremental Delaunay insertion [J]. Computers & Graphics, 2002, 26(3): 477-490. [DOI:10.1016/S0097-8493(02)00090-0]
- [14] Sohler C. Fast reconstruction of Delaunay triangulations [J]. Computational Geometry, 2005, 31(3): 166-178. [DOI: 10.1016/j.comgeo.2004.12.006]
- [15] Yang S W, Choi Y, Jung C k. A divide-and-conquer Delaunay triangulation algorithm with a vertex array and flip operations in two-dimensional space [J]. International Journal of Precision Engineering and Manufacturing, 2011, 12(3):435-442. [DOI: 10.1007/s12541-011-0056-1]
- [16] Pinto F, Freitas C. Dynamic Voronoi diagram of complex sites [J]. The Visual Computer, 2011, (27): 463-472. [DOI: 10.1007/s00371-011-0581-z]
- [17] Qi M, Cao T T, Tan T S. Computing 2D constrained Delaunay triangulation using the GPU [J]. IEEE Transactions on Visualization and Computer Graphics, 2013, 19(5):736-748. [DOI: 10.1109/TVCG.2012.307]