DOI:10.13203/j.whugis20140443　　文章编号:1671-8860(2016)06-0832-06

城市密集点云的区域生长表面构网改进算法

卢学良[1,2,3]　童晓冲[1]　张永生[1]　谢金华[4]　于　英[1]

1　信息工程大学,河南 郑州,450001
2　西安测绘研究所,陕西 西安,710054
3　地理信息工程国家重点实验室,陕西 西安,710054
4　国家测绘地理信息局卫星测绘应用中心,北京,100000

摘　要:随着飞行平台与传感器技术的发展,空间信息的获取能力逐渐增强,特别是低空倾斜摄影及其数据处理方法的出现,使得城市真三维信息的获取手段变得更加丰富。针对现有区域生长算法在城市三维密集点云拐角处未能完全生长的问题,设计了点到边(point to edge,PTE)数据结构,提出了遍历 PTE 数据结构以完善构网的改进策略。实验结果表明,改进算法能对城市真三维密集点云进行较好的表面构网,并且能够解决现有方法在拐角处不能完全生长的问题。

关键词:倾斜摄影;真三维点云;表面构网;区域生长

中图法分类号:P237　　**文献标志码**:A

传统城市三维数据生产主要是以航空摄影测量技术和激光探测与测量(light detection and ranging,LiDAR)技术为主[1-3]。近年来,大倾角航空摄影技术方法被广泛使用,该方法与传统方法最大的不同是在获取地物顶部信息的同时还获取了大量地物的侧面信息,利用丰富的点云数据构建地面的表面模型,再通过纹理映射,可以为智慧城市的三维模型构建提供高质量的基础数据[4,5]。而这其中,点云数据的快速表面构网方法是研究的重点之一。国外的一些商业软件(如 Smart3D)已经可以基本做到从数据输入到三维表面模型建立的自动化,但是其技术细节并没有完全公开。国外的一些文献[6-9]主要侧重于整体技术流程的概述,没有对真三维数据表面构网的具体技术细节和问题进行深入分析。

现有针对三维点云表面构网的方法可以分为隐式曲面算法[10-12]、三维 Delaunay 三角化算法[13-15]以及区域生长算法[16-18]。由于城市真三维密集点云的突出特点是数据量大,在对构网效率充分考量的基础上,结合算法的复杂度和应用性,本文选取了区域生长算法作为基础算法,并针对数据特点对算法进行了改进,取得了较好的效果,初步解决了传统区域生长算法在诸多拐角处不能完全生长的局限性。

1　现有区域生长算法与存在的问题

区域生长算法首先由 Bernardi 提出[16],在后续的发展中 Lin[17]和 Angelo[18]也做出了贡献。以上学者研究的不同之处主要在于区域生长的策略不同。下面就文献[18]中的方法进行简要介绍。

1) 点云数据空间索引建立。采用 k-d 树或 hash 表的方式构建空间索引,高效索引的建立有利于后续生长过程中邻近点的查找,能加快重建速度。

2) 种子三角形的构建。随机选取一个起始点,查找起始点的最邻近点,连接起始点与其最邻近点,作为待选种子三角形的一条边。以该边的中点为球心,边长的 k 倍为半径,建立搜索球,遍历搜索球内的每一个点,将其与已知边组成三角形,验证该三角形的最小外接球内是否存在其它点,如果不存在,则该三角形为待选种子三角形;验证该三角形是否在点云数据表面,如果满足要求,则种子三角形构建完成;如果不在点云数据表面,则重复以上步骤,直到种子三角形构建完成。

3) 区域生长策略。首先定义两个概念:未完

收稿日期:2014-11-20
项目资助:国家自然科学基金(41201392,41401534);航空遥感技术国家测绘地理信息局重点实验室经费 (2014B02)。
第一作者:卢学良,硕士,主要研究方向为城市三维建模及点云处理。lxl_rs@163.com

成构网的边(只隶属于一个三角形的边,边界边除外)称为活动边,未完成构网的三角形(含有活动边的三角形)称为活动三角形。

首先搜索活动边指定范围内的候选点。在活动三角形所在的平面上作活动边的中垂线,在其中垂线上距活动边一定距离处确定搜索球的球心。根据球心和活动边的端点确定搜索球的半径,根据球心和半径计算球内所包含的点,即待选点。根据待选点确定最佳候选三角形的方法已在种子三角形的构建中说明,这里不再赘述。

确定最佳候选三角形以后,对新增的两条活动边进行验证,判断其是否属于其他三角形,根据判断情况,修改活动边列表,递归完成构网。

由于城市密集点云的主体为建筑物,而建筑的拓扑形态中不可避免地包含很多拐角,特别是直角拐角,然而候选点的搜索策略(如图 1 所示)却难以保证拐角处的生长点被搜索到,进而难以保证在拐角处进行了正确生长。即便生长点能被搜索到,也往往因为拐角的原因难以满足空球性质(如图 2 所示)而被舍弃。针对该情况,本文将给出算法的改进思路。

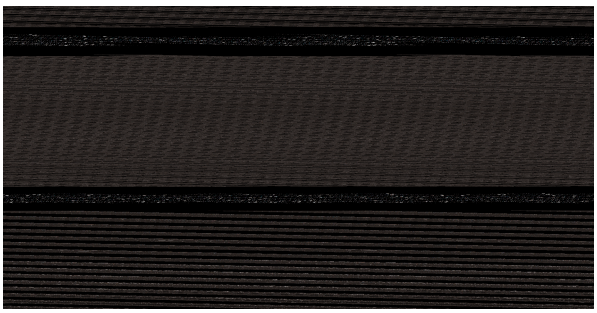

图 1　拐角处候选点的搜索

Fig. 1　Search of Candidate Points at Corner

图 2　拐角处空球性质验证

Fig. 2　Validation of Empty Ball Nature at Corner

2　改进算法

2.1　改进思路

直角拐角处表面构网如图 3 所示。其中 v_i($i=1,2,\cdots,12$)表示顶点,这里既代表标示符也代表索引值;e_i($i=1,2,3$)表示活动边;t_i($i=1,2,\cdots,13$)表示三角形,其中 t_3、t_7、t_8 为活动三角形,t_{13} 表示未完成生长的三角形,e_1 为直角拐角处的边。理论上一个确定三角形的构建有 3 种情况,以三角形 t_{13} 为例,可以以 e_1 为活动边向 v_9 方向生长,以 e_2 为活动边向 v_6 方向生长以及以 e_3 为活动边向 v_5 方向生长。然而三角形 t_{13} 在实际构建过程中有如下情况:

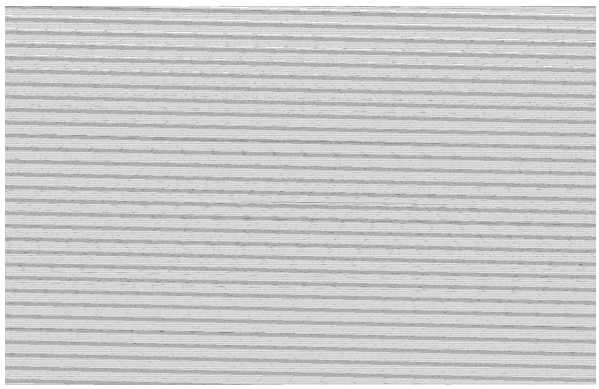

图 3　拐角处的构网

Fig. 3　Triangulation at Corner

(1) 以 e_1 为活动边向 v_9 方向生长时,由于 e_1 为拐角边,导致搜索不到候选点,因此构建失败;

(2) 以 e_2 为活动边向 v_6 方向生长时,在搜索区域内候选点在包含 v_6 的同时往往也包含 v_2,然而三角形 t_{13} 却不满足空球性质,也就是三角形 t_{13} 的最小外接球包含 v_2,因此构建失败;

(3) 以 e_3 为活动边向 v_5 方向生长时,情况与(2)类似。

由于直角拐角是城市密集点云的显著特征,因此在表面重建的过程中会大量出现这种情况,导致拐角处有很多三角形未完成构建,出现很多空洞。针对此类问题,本文设计了一种新的点到边数据结构(point to edge,PTE)来解决该问题。在点云数据生长完成以后,遍历该数据结构对上述情况进行处理。图 4 为 PTE 数据结构处理拐角处三角形的示意图。

为了方便对 PTE 数据结构进行介绍,将拐角处的生长情况简化到平面上描述,需要特别说明的是,在近似平面上一般不会出现此类未完全生长的情况。

首先为点云数据的每个顶点建立一个容器,容器的标示符为 $pV[i]$,其中 i 表示顶点的索引值,容器中的元素为与顶点 i 相连接所组成边的另一个顶点的索引值。容器中元素的增加规则为只要生长出新的三角形,就将三角形的每个顶点

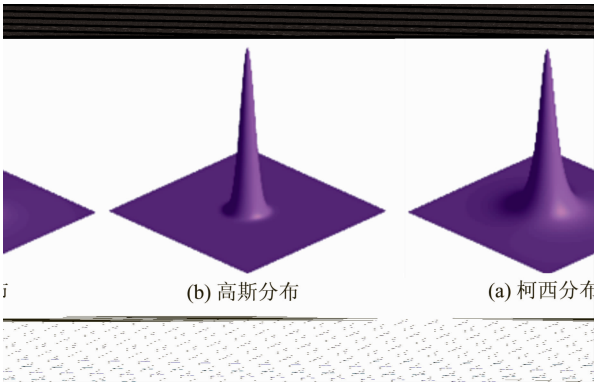

图 4 PTE

Fig. 4 Sketch of PTE Processing

所对应的其余两个顶点的索引值加入该顶点容器。

假设图 4 中所示区域按照三角形 t_1 到 t_{12} 的顺序进行生长，且以三角形 t_{13} 为例，如果该三角形构建完成，则顶点 v_5、v_6、v_9 所对应的容器中的所有元素如表 1 所示。

将表 1 中的重复元素去除，得到只带 * 的实际元素，可得 $pV[v_5]$中的元素为 v_6 和 v_9，由元素 v_6 检查 $pV[v_6]$是否包含元素 v_9，若包含且仅包含一个，则三角形 t_{13} 构建完成，然后补充其顶点所对应的容器的元素；若已经包含两个，则可能是拓扑结构构建错误的地方，暂不作任何处理；若不包含，则可能是多个未完成生长的三角形连在一起，也可能是边界情况，可以根据向量间的夹角作具体处理，这里不作详细介绍。同理可以根据 $pV[v_6]$、$pV[v_9]$中元素的情况，按照上述步骤完成三角形的构建，它们之间处理的先后依据为容器所对应的索引值的大小，将整个 PTE 结构按照上述规则遍历完毕后，拐角处未完成构网的三角形则构建完成。

表 1 容器包含的所有元素

Tab. 1 All Elements in Container

容器	元素									
$pV[v_5]$	v_1	v_4	v_1	v_2	v_2	$\boldsymbol{v}_6^*$	v_4	v_8	v_8	$\boldsymbol{v}_9^*$
$pV[v_6]$	$\boldsymbol{v}_5^*$	v_2	v_2	v_3	v_3	v_7	$\boldsymbol{v}_9^*$	v_{10}	v_{10}	v_7
$pV[v_9]$	v_8	$\boldsymbol{v}_5^*$	$\boldsymbol{v}_6^*$	v_{10}	v_8	v_{11}	v_{11}	v_{12}	v_{12}	v_{10}

2.2 算法步骤

改进算法的基本思路就是在文献[18]的算法基础上增加遍历 PTE 数据结构这一步骤，进而对构网进行完善。由于需要遍历 n(点云数量)个 PTE 数据结构，而对每个 PTE 的遍历在 $O(1)$时间内完成，所以改进算法的时间复杂度相较于原算法只增加了 $O(n)$，基本不会对构网效率有影响。

3 实验与分析

本文在文献[18]的基础上，进行了算法改进，并通过相同的实验数据和实验结果进行原始算法与改进算法这两种算法的比对。实验环境为：Intel Core i5 CPU(2.8 GHz)，3 GB DDR3 内存，操作系统是 Windows7 32 位。实验数据采用某城市两个区域(以下简称 A 区域和 B 区域)的真三维点云数据。其中，A 区域的数据量为 4.78 MB(文本文件)，含有128 160个三维点；B 区域的数据量为 10.3 MB(文本文件)，含有322 319个三维点。A、B 区域的构网概略图如图 5 所示，其中图 5(a)表示 A 区域，图 5(b)表示 B 区域。对 A 区域通过两种算法进行表面构网的局部放大对比图如图 6所示，对 B 区域通过两种算法进行表面构网的局部放大对比图如图 7 所示。对两区域通过两种算法进行构网所生成的三角形个数、消耗时间、漏洞个数等指标进行统计，结果如表 2 所示。

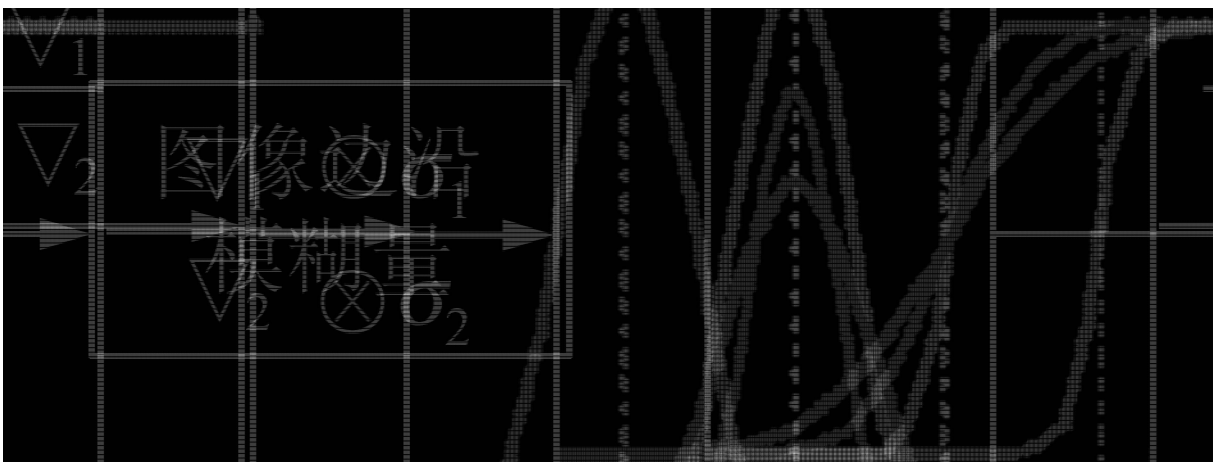

图 5 构网概略

Fig. 5 Triangulation Skeleton

表 2　数据处理结果统计

Tab. 2　Statistics of Data Processing Results

区域	点云数量	算法	三角形个数	构网时间/s	漏洞个数	漏洞比例/‰
A	128 160	原始算法	213 918	10.1	1 115	5.2
		改进算法	215 325	10.5	321	1.5
B	322 319	原始算法	635 943	32.4	1 004	1.6
		改进算法	638 126	33.5	278	0.4

图 6　*A* 区域局部放大对比效果

Fig. 6　Magnifying Comparison of Region *A*

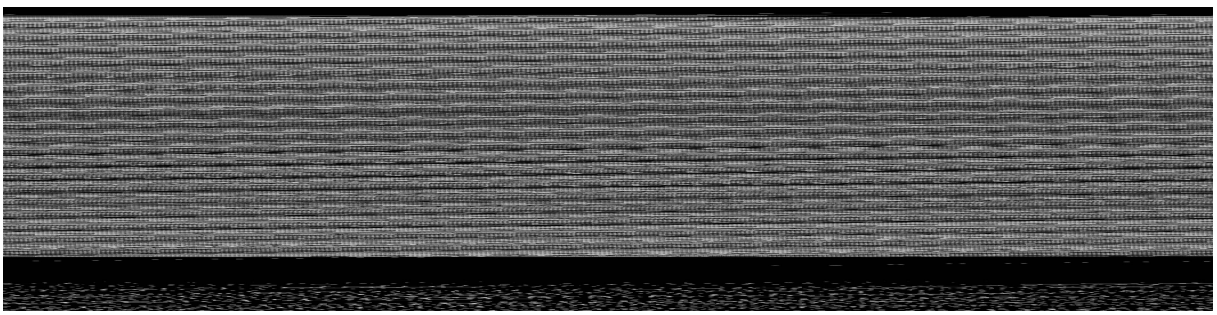

图 7　*B* 区域局部放大对比效果

Fig. 7　Magnifying Comparison of Region *B*

通过两种算法的对比效果图以及相关指标统计可以看出,改进算法对原算法处理城市点云拐角处的不适应性进行了较好的处理,漏洞数量显著降低,并且改进算法相较于原始算法的效率几乎没有任何下降。但该改进算法并不能处理所有漏洞,究其原因,本文算法改进的本质是在点云拓扑结构构建正确的基础上,对网格的进一步完善,但是有些漏洞属于拓扑结构上的构建错误,如图 8所示为拐角与地面连接处的错构情况;而图 9表示的是在某些起伏剧烈且采样稀疏地区。对于

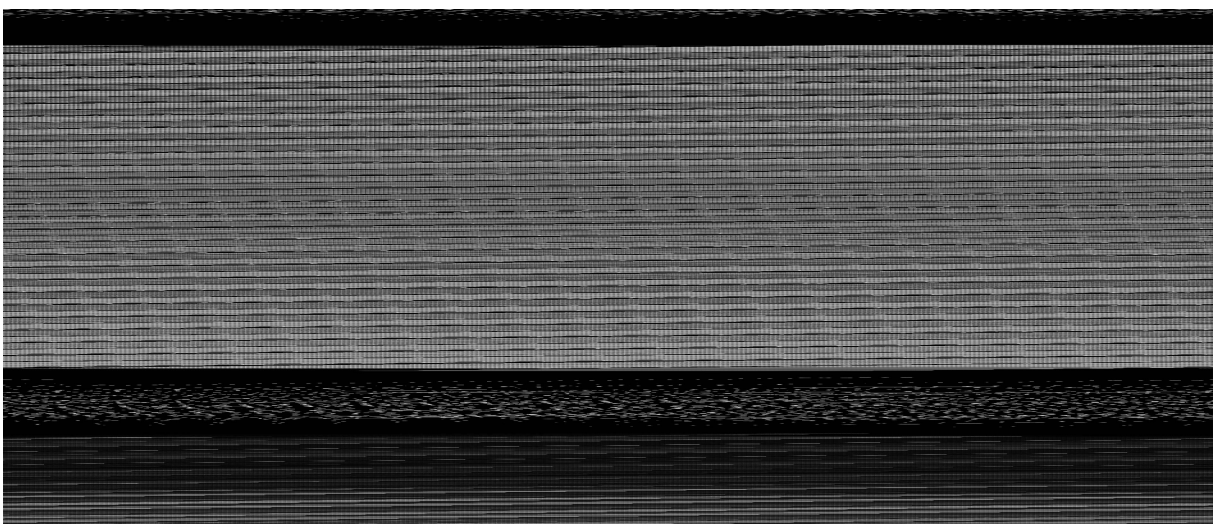

图 8　拐角与地面连接处的错构情况

Fig. 8　Error Structures at the Corner of Ground

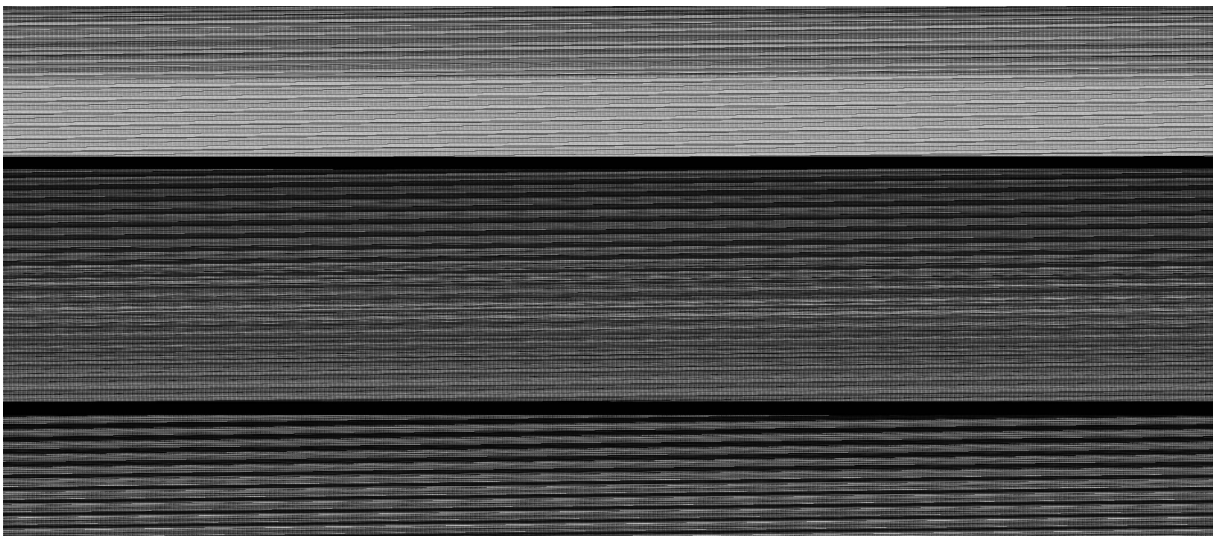

图9 起伏地区的错构情况

Fig. 9 Error Structures in Rolling Region

这些情况,本文算法因不能正确辨别其拓扑结构,会导致错构的发生。

4 结 语

本文对现有三维表面构网算法中的生长算法进行了分析,并针对建筑物数据拐角特征明显的特点对算法进行了改进。实验证明,该改进策略是有效的,能够解决大部分的拐角处不能完全生长问题。

拐角处未完全生长只是区域生长算法处理城市密集点云的一个问题,其他的问题还包括拓扑结构构建错误等,这将是下一步需要研究的方向。

参 考 文 献

[1] Mu Chao, Yu Jie, Xu Lei. Extracting Building Points from the DSM Data Combining the High-Resolution Remote Sensing Image[J]. *Geomatics and Information Science of Wuhan University*, 2009, 34(4): 414-417(穆超,余洁,许磊. 基于高分辨率遥感影像的DSM建筑物点的提取研究[J]. 武汉大学学报·信息科学版,2009, 34(4): 414-417)

[2] Wang Ren, Xu Qing. Zhu Xinhui. Picking up Foot Prints of Building from Airborne LiDAR Data with Multi-strategies[J]. *Geomatics and Information Science of Wuhan University*, 2008, 33(7): 688-691(王刃, 徐青, 朱新慧. 用多种策略从机载LiDAR数据中提取建筑脚点[J]. 武汉大学学报·信息科学版,2008,33(7): 688-691)

[3] Poullis C, You S. Automatic Creation of Massive Virtual Cities[C]. IEEE Virtual Reality Conference, Louisiana, 2009

[4] Li Zhenzhou, Zhang Xuezhi. Research on the Quick Construction of 3D Model of City Based on Oblique Photogrammetric Technique[J]. *Geomatics & Spatial Information Technology*, 2012, 35(4): 117-119(李镇洲,张学之. 基于倾斜摄影测量技术快速建立城市3维模型研究[J]. 测绘与空间地理信息, 2012,35(4): 117-119)

[5] Musialski P, Wonka P, Aliaga D G, et al. A Survey of Urban Reconstruction[J]. *Computer Graphics Forum*, 2013, 32(6): 146-177

[6] Merchan P, Adan A, Salamanca S, et al. Geometric and Colour Data Fusion for Outdoor 3D Models [J]. *Sensors*, 2012, 12(6): 6 893-6 919

[7] Kuschk G. Large Scale Urban Reconstruction from Remote Sensing Imagery[J]. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2013, XL-5/W1: 139-146

[8] Kuschk G. Model-Free Dense Stereo Reconstruction for Creating Realistic 3D City Models[C]. Urban Remote Sensing Event, IEEE Joint, Brazil, 2013

[9] Canciani M, Falcolini C, Saccone M, et al. From Point Clouds to Architectural Models: Algorithms for Shape Reconstruction[J]. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2013, XL-5/W1: 27-34

[10] Boissonnat J D. Geometric Structures for Three Dimensional Shape Representation[J]. *ACM Transactions on Graphics*, 1984, 3(4): 266-286

[11] Kolluri R. Provably Good Moving Least Squares [J]. *ACM Transactions on Algorithms*, 2005, 4(2): 782-795

[12] Kazhdan M, Bolitho M, Hoppe H. Poisson Surface Reconstruction[C]. Eurographics Symposium on Geometry Processing, Switzerland, 2006

[13] Amenta N. A New Voronoi-based Surface Reconstruction Algorithm[C]. ACM Conference on Computer Graphics and Interactive Techniques, Atlanta, 1998

[14] Dey T K, Goswami S. Tight Cocone: A Water-Tight Surface Reconstructor[C]. ACM Symposium on Solid Modeling and Applications, Seattle, 2003

[15] Cohen-Steiner D, Da F. A Greedy Delaunay-based Surface Reconstruction Algorithm[J]. *The Visual Computer*, 2004, 20(1): 4-16

[16] Bernardini F. The Ball-Pivoting Algorithm for Surface Reconstruction[J]. *Visualization and Computer Graphics*, 1999, 5(4): 349-359

[17] Li X, Han C Y, Wee W G. On Surface Reconstruction: A Priority Driven Approach[J]. *Computer-Aided Design*, 2009, 41(9): 626-640

[18] Di Angelo L. A New Mesh-Growing Algorithm for Fast Surface Reconstruction[J]. *Computer-Aided Design*, 2011, 43(6): 639-650

An Improved Region-Growing Surface Triangulation Algorithm for Urban Dense Point Cloud

LU Xueliang[1,2,3]　*TONG Xiaochong*[1]　*ZHANG Yongsheng*[1]　*XIE Jinhua*[4]　*YU Ying*[1]

1　Information Engineering University, Zhengzhou 450001, China

2　Xi'an Research Institute of Surveying and Mapping, Xi'an 710054, China

3　State Key Laboratory of Geo-information Engineering, Xi'an 710054, China

4　Satellite Surveying and Mapping Applications Center, Beijing 100000, China

Abstract: With the development of flying UAV platforms, sensor technologies, and corresponding data processing methods, low-altitude oblique photography becomes a powerful method to obtain true three-dimensional urban spatial information. In existing region-growing methods however, corners within urban dense point clouds are not fully grown. In this paper, we propose an improvement over the previous mesh-growing algorithms for fast surface reconstruction, a PTE (point to edge) data structure and traversing algorithm. Experimental results show that the improved algorithm is able to handle dense true three-dimensional urban point clouds,and can solve most of the incomplete growing problems in corners, with better performance than the exiting methods.

Key words: oblique photography; true three-dimensional point cloud; surface triangulation; region growing

First author: LU Xueliang, master, specializes in the urban 3D reconstruction and point cloud processing. E-mail: lxl_rs@163.com

Foundation support: The National Natural Science Foundation of China, Nos. 41201392, 41401534; the Fund of the Key Laboratory for Aerial Remote Sensing Technology of National Administration of Surveying, Mapping and Geo-information, No. 2014B02.