

文章编号: 1001-3806(2018)03-0410-07

基于 LiDAR 数据的布料模拟滤波算法的适用性分析

张昌赛^{1,2,3}, 刘正军^{2*}, 杨树文¹, 左志权²

(1. 兰州交通大学 测绘与地理信息学院, 兰州 730070; 2. 中国测绘科学研究院 摄影测量与遥感所, 北京 100830; 3. 甘肃省地理国情监测工程实验室, 兰州 730070)

摘要: 为了解决现有点云滤波算法设置参量多、滤波效果不理想、难以操作等问题, 采用最新的布料模拟滤波算法, 根据简单的物理过程构建虚拟格网模拟地形表面, 并针对复杂地形的点云数据进行定性和定量滤波验证。结果表明, 其 I 类误差在 5.7%、II 类误差在 3.4% 以内, 但针对部分混合有平坦和陡坡的局部区域滤波效果并非理想; 在满足滤波精度的同时, 可在 30s 内完成数百万个点的滤波, 甚至在数秒内完成数十万个点的滤波。该算法所需参量很少、效率非常高, 能满足绝大多数复杂地形数据的滤波要求。

关键词: 遥感; 点云滤波; 机载激光探测与测量; 布料模拟滤波

中图分类号: P237

文献标志码: A

doi: 10.7510/jjgs.issn.1001-3806.2018.03.023

Applicability analysis of cloth simulation filtering algorithm based on LiDAR data

ZHANG Changsai^{1,2,3}, LIU Zhengjun², YANG Shuwen¹, ZUO Zhiquan²

(1. Faculty of Geomatics, Lanzhou Jiaotong University, Lanzhou 730070, China; 2. Institute of Photogrammetry and Remote Sensing, Chinese Academy of Surveying and Mapping, Beijing 100830, China; 3. Gansu Provincial Engineering National Laboratory for Geographic State Monitoring, Lanzhou 730070, China)

Abstract: In order to solve the problems in the existing point cloud filtering algorithm, such as too many parameters, without ideal filtering effect and inconvenient operation, the cloth simulation filtering (CSF) algorithm was chosen to construct virtual grid to simulate terrain surface. The algorithm was verified using the qualitative and quantitative experiment and analysis in complex terrain point cloud data. The results show that type I error is less than 5.7%, type II error less than 3.4%, but for the local area with some mixed flat and steep slope the filtering effect is not ideal; The algorithm can achieve the filtering of millions of points in 30s while satisfying the filtering precision, and even hundreds of thousands of points in several seconds. The algorithm requires few parameters and is very efficient. It can satisfy the filtering requirements of most complex terrain data.

Key words: remote sensing; point cloud filtering; airborne laser detection and measurement; cloth simulation filtering

引言

机载激光探测与测量(light detection and ranging, LiDAR)技术最初是作为一种获取数字地面模型等地形信息的新技术, 即获取并生成数字高程模型(digital elevation model, DEM)或者数字地形模型(digital terrain model, DTM), 这也是 LiDAR 数据主要用途之一。DEM, DTM 以及数字表面模型(digital surface model,

DSM)的生成算法已经较为成熟, 但是其所需要的点必须是地面点, 所以在得到地形信息之前需要先从原始点云数据中分离出地物点, 如建筑物和植被, 即滤波^[1-4]。点云数据滤波 DEM 生产技术是最基础的数据处理工作, 它对后续的数字产品生产应用保障都起到关键作用^[5]。因此研究合理、快速、准确的自动滤波算法具有重要意义。

机载激光雷达测量技术经过数十年的发展, 技术已经成熟, 硬件方面的关键问题基本都已经得到解决^[6]。在 LiDAR 的数据滤波方面, 虽然已经有很多国内外的学者研究并提出了滤波的算法, 但是纵观已有算法各有其弊端, 尚且不存在一种适合于各种地形等自然条件的滤波方法。滤波算法的性能与地形的复杂程度最为相关, 理想的滤波算法应该能够因地形的变

基金项目: 国家自然科学基金资助项目(41371406); 国家重点研发计划资助项目(2017YFB0504201)

作者简介: 张昌赛(1990-), 男, 硕士研究生, 现主要从事机载 LiDAR 点云数据后处理的研究。

* 通讯联系人。E-mail: zjliu@casm.ac.cn

收稿日期: 2017-06-16; 收到修改稿日期: 2017-07-17

化而做到自适应的调整^[7]。由于 LiDAR 数据的采集带有一定的盲目性,不能针对地形特征点进行采集,因而 LiDAR 数据的滤波中地形的起伏引起的地表不连续与地物原因引起的地表不连续很难区分。滤波分类算法设计时应尽量保留测区的地形特征点信息(如山脊线、陡坎等),并尽量减少分类误差^[8]。分类误差包括两种:第Ⅰ类误差,拒绝本属于地面的激光脚点而将其分类为地物点;第Ⅱ类误差,错误地接收了不属于地面的激光脚点^[9]。

目前国内外学者提出的滤波算法^[10-15]大都仅仅适用于处理城区、山区和林区等地形区域中的某一种或两种区域^[16],不能适用于综合多种地形特征的区域,即普适性较差。点云滤波目前存在两个主要问题:首先多嵌入在商品化软件中,免费软件少,而且参量多普适性差,选择合适的参量得到满意的滤波结果需要较为丰富的专业知识和经验,从而限制了点云数据的使用范围和人群;其次,点云滤波的主要原理已经多年没有突破,研究多集中在基于不规则三角网、高度、坡度、表面拟合以及形态学算法的改进。

针对该现状,现急需寻找一种滤波算法能够有效解决以上问题。本文中利用 ZHANG 首次提出的点云滤波算法——布料滤波算法^[17]对 LiDAR 点云滤波进行了适用性分析。该算法主要思想是首先将 3 维点云翻转,然后在翻转后点云表面上构建一个虚拟的格网,结合一种物理过程的弹簧模型^[19-20],分析格网上节点和对应激光脚点之间的相互作用,可以形成一个近似的地形表面,以此来分类地面点和非地面点。本文中通过具体实例对算法进行分析验证,检验算法滤波性能。

1 CSF 算法概述

1.1 算法原理

布料模拟滤波(cloth simulation filtering, CSF)算法是基于一种简单的物理过程模拟。该算法假设一块虚拟的布料受重力作用落在地形表面上,如果这块布足够软,则会贴附与地形上,而布料的形状就是 DSM。当地形被翻转过来时,则落在表面上的布料形状就是 DEM,如图 1a 所示。该算法使用了布料模拟^[18]方式来提取激光点云数据的地面点。

布料是由大量相互联系的节点构成的格网,如图 1b 所示,这种形式被称为质点弹簧模型^[19-20],在这个模型中,布料点通过虚拟弹簧相互联系,点与点之间的相互作用遵循弹性定律,即布料点受力之后,其应力和

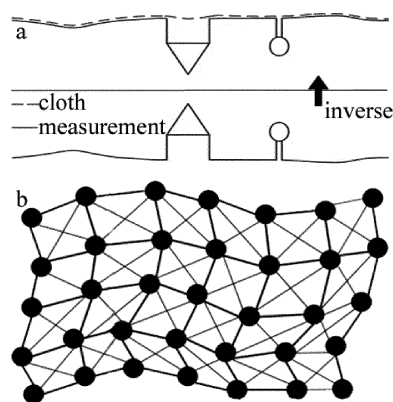


Fig. 1 Overview of CSF algorithm

a—schematic illustration of CSF b—schematic illustration of the grid of CSF

应变之间呈线性关系,点受力的作用后,会产生位移。为模拟某一时刻布料的形状,需要计算布料点在 3 维空间里的位置。结合牛顿第二定律,布料点的位置和相互作用力关系遵照下式:

$$X(t + \Delta t) = 2X(t) - X(t - \Delta t) + \frac{G}{m}\Delta t^2 \quad (1)$$

式中, m 是布料点的质量,被设为常数 1; X 是某个时刻的节点位置; Δt 是时间步长; G 是重力常数。给出时间步长和节点初始位置,就会得到当前点的位置。

需要注意的是,布料上的点只能在垂直方向上移动,初始状态的布料会附着于翻转后激光点云地形表面上且与其还有一定的距离;布料点的初始位置位于翻转后有最大高程值的激光点位置上与其保持一定距离,通常是个固定值;布料节点只受到两种力,重力和邻近节点相互之间的作用力;由于受到邻近节点的牵拉力,布料点有返回到与其邻近节点同一水平面位置的趋势;当布料上的点移动到地面激光点云的位置时,其被标记为不可移动点,否则,被标记为可动点。

布料模拟过程中,首先计算布料点受重力作用而产生位移后的位置,通过(1)式计算。另外,在翻转表面的空白区域,通常为建筑物或深坑等微地形,为限制布料点的移动,需要修正布料点受邻近节点之间的作用力而移动后的位置,为此需要计算相邻布料点的高程差。如果两个相邻节点都是可动点,有不同的高程值,则它们在垂直方向上以相反方向移动相同的距离;如果两者有一个不可动点,则只有另外一个点移动;如果两者位于同一高程,则都不移动。每个布料点的修正位移按照下式进行计算:

$$d = \frac{1}{2}b(p_i - p_0) \cdot n \quad (2)$$

式中, d 为节点的移动向量; p_0 为待移动节点的当前位

置; p_i 为 p_0 的邻近节点位置; n 为垂直方向上的标准向量, $n = (0, 0, 1)^T$; b 是用于判定节点是否移动的参量,当节点为可动点时,设 $b = 1$,否则 $b = 0$ 。

1.2 滤波步骤

首先将布料点和激光点云投影到同一水平面,然后找到每个布料节点的对应的激光点。比较每个节点和其对应激光点的高程。如果节点高程小于或等于激光点高程,则将该节点移动到对应激光点的位置,并将其标记为不可动点。通过布料模拟可获得一个近似真实的地形。使用点与点距离估算算法^[21] 计算激光点和模拟布料点之间的距离,小于阈值 h 的激光点被分为地面点,否则被分为非地面点。具体的滤波算法步骤如下:(1)转换点云几何坐标,使原始的激光点云翻转;(2)初始化布料格网,通过格网分辨率确定格网节点数量;(3)将激光点和格网点投影到同一水平面,寻找每个格网点对应的激光点,并记录对应激光点的高程值;(4)计算格网节点受重力作用移动的位置,并比较该位置高程与其对应激光点高程,如果节点高程小于或等于激光点高程,则将该节点位置替换到对应激光点的位置,并将其标记为不可动点;(5)计算每个格网点受邻近节点影响而移动的位置;(6)重复步骤(4)和步骤(5),当所有节点的最大高程变化足够小或者超出最大迭代次数时,模拟过程终止;(7)分类地面点和非地面点,计算格网节点和相应激光点之间的距离。对于激光点云,如果该距离小于阈值 h ,被分为地面点,否则被分为非地面点。

1.3 滤波参量

CSF 算法主要包括 3 个用户自定义的参量:格网分辨率表示两个邻近节点之间的距离;硬度控制布料的松紧程度,硬度越高,布料越紧致,共分 3 个级别,平坦地形硬度为 1,略微起伏地形硬度为 2,陡峭地形硬度为 3;陡坡拟合因子是一个可选参量,用于边坡后处理。另外还有距离阈值,用于地面点和非地面点的最终分类,通常设置为 0.5m 即可满足绝大部分数据的滤波要求;迭代次数,通常设为 100 ~ 150 以内即可满足滤波要求,可默认 100。

2 滤波实验

2.1 定性分析

本文中选用不同地形的高密度点云数据进行滤波实验,生成数字地形模型,首先通过比较滤波前后的点云断面,来直观判断滤波效果。本文中选用的 4 组数据信息如表 1 所示。

Table 1 Characteristics of test data set of data 1 ~ data 4 (point)

dataset	point number	scope	density	features
1 [#]	6185384	0.7km × 0.6km	11.81 point/m ²	flat terrain, large buildings, potholes
2 [#]	4978327	2.0km × 1.5km	1.61 point/m ²	urban, buildings, vegetation
3 [#]	23097984	0.6km × 0.9km	42.77 point/m ²	mountain, steep slopes, dense vegetation
4 [#]	19659989	1.2km × 0.5km	32.77 point/m ²	mixture of flat terrain and hillside

2.1.1 实例 1 下面给出滤波实例,所选区域是一个较为平坦的区域,该区域有树木、低矮建筑物、高大建筑物以及深坑。滤波前后的点云数据如图 2a 和图 2b 所示。从图 2b 可看出,建筑物、树木均被有效滤除,而微地形(如深坑)得到有效保留。图 2c 是根据保留的地面点生成的 DTM。图 2d 为滤波前后的地形剖面对比,其中 A 为包含地物点的地形剖面,B 为地面点,可以看出地面点和地物点被很好地分离。

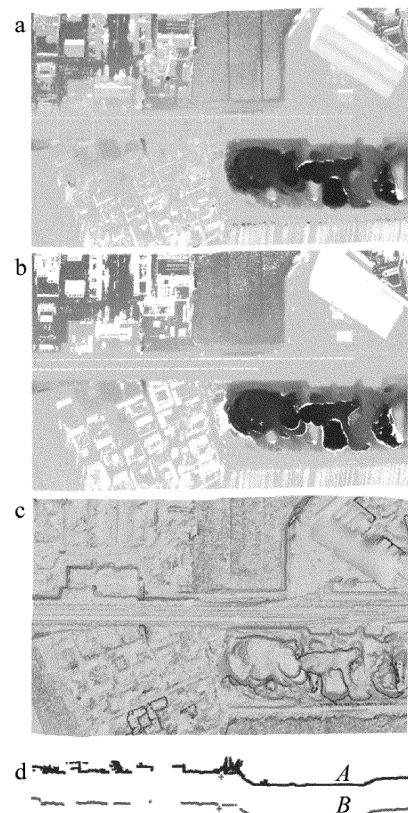


Fig. 2 Filtering results of data 1

a—original LiDAR point clouds b—LiDAR ground point after filtering
c—digital terrain model d—terrain profile before and after filtering

2.1.2 实例 2 所选区域为城市区域,该区域包括建筑物、植被、河流等。原始点云和滤波后地面点云如图 3a 和图 3b 所示。图 3c 中插值生成的 DTM,地面细节信息得到保留,植被被有效识别。图 3d 为滤波前后

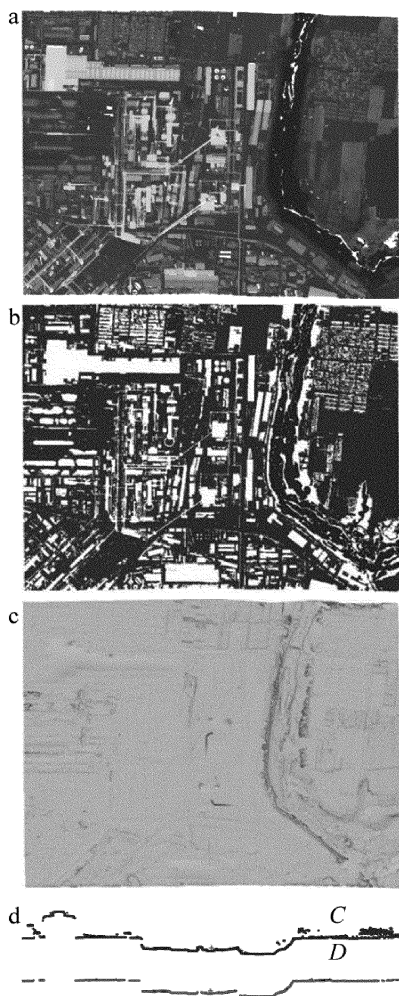


Fig. 3 Filtering results of data 2

a—original LiDAR point clouds b—LiDAR ground point after filtering
c—digital terrain model d— terrain profile before and after filtering

的地形剖面,位于建筑物包围下的地面及处于建筑物和树木下的地面都得到正确识别。

2.1.3 实例3 所选区域为山地,地形起伏较大,地面多被植被覆盖。图4a和图4b分别为滤波前后的点云数据。利用CSF滤波后生成的DTM见图4c。图4d为滤波前后的地形剖面,其中E为包含地物点的地形剖面,F为地面点。通过对比可以看出,数据中的地物部分被过滤,同时还保留了重要的地形特征,如沟壑、陡坡和道路,滤波效果非常好。

2.1.4 实例4 所选区域为混合区域,既有高大陡峭的山坡,又有平坦区域,其中还包括建筑物和植被。由图5b中滤波后地面点和图5a中原始点云对比发现,山坡部分地面点丢失。图5d为滤波前后的地形剖面,从中也可看出明显的错分,陡坡位置的部分地面点丢失,部分建筑物也没有被剔除。该算法针对混合区域滤波效果并非理想。

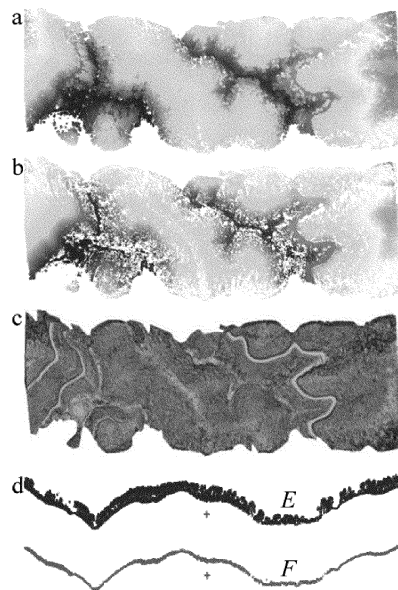


Fig. 4 Filtering results from data 3

a—original LiDAR point clouds b—LiDAR ground point after filtering
c—digital terrain model d—point cloud filtering section

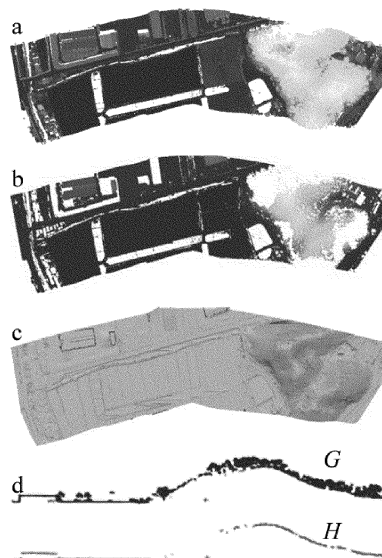


Fig. 5 Filtering results of data 4

a—original LiDAR point clouds b—LiDAR ground point after filtering
c—digital terrain model d— terrain profile before and after filtering

2.2 定量分析

2.2.1 误差分析 因为缺少可已分类好的参考点云数据,本文中选用两组数据实例数据5和实例数据6,其数据信息如表2所示。分别进行人工判读后,将地面点和地物点分离,并依据这两组分类好的数据作为参考数据进行实验分析。

实例数据5和数据6分别为相对平坦区域和相对陡峭区域。图6a和图7a是原始点云,按照高程进行赋色后,地形由高到低以红黄绿蓝渐变。图6b和图

Table 2 Characteristics of test data set of data 5 and data 6

dataset	point number	density	features	ground point number	not-ground point number
5 [#]	418021	5.79point/m ²	flat terrain, vegetation, building	252936	165085
6 [#]	505619	6.57point/m ²	vegetation on hillside, power line, low shrubs	439421	66198

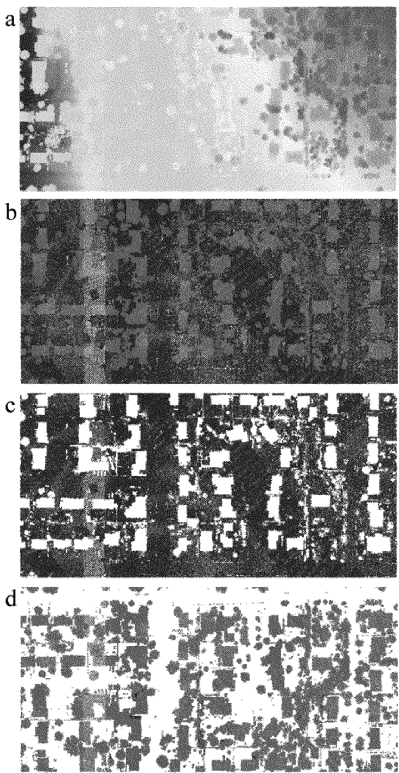


Fig. 6 Filtering results of data 5

a—original LiDAR point clouds with elevation color b—reference point cloud after manual classification c—ground point clouds after filtering
d—nonground point clouds after filtering

7b 是依据人工判读进行手工分类后的点云数据,其中蓝色为地面点,红色为地物点。实例数据 5 滤波后地面点包含很少一部分非地面点(见图 6c 中红色部分),滤波后非地面点几乎没有掺杂地面点(蓝色部分),即 I 类误差和 II 类误差都非常小,如表 3 所示,滤波效果非常理想。实例数据 6 包含低矮灌木、树木和电力线。其中灌木因比较矮小且接近地面,对于算法本身来讲,其属于陡峭地物,算法边坡后处理具有一定局限性,因此极易被误分为地面点(见图 7c 中红色

Table 3 Two types of error statistics

category	data 5		data 6	
	ground point number	non-ground number	ground point number	non-ground number
ground point number	252936	1658	439421	2251
non-ground number	4047	165085	25047	66198
type I	1.60%		5.7%	
type II	1.05%		3.4%	

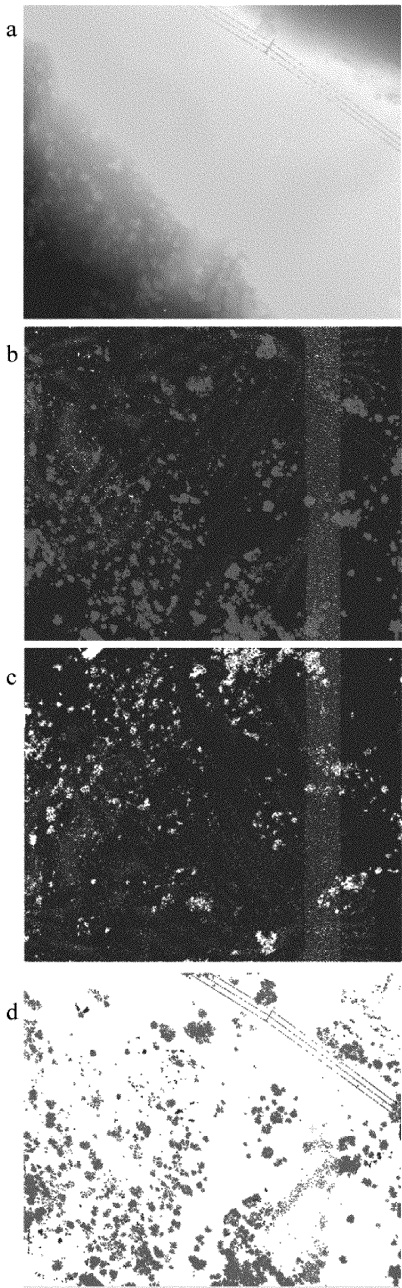


Fig. 7 Filtering results of data 6

a—original LiDAR point clouds with elevation color b—reference point cloud after manual classification c—ground point clouds after filtering
d—nonground point clouds after filtering

部分)。相对来说地物点被很好地分离,如图 7d 中蓝色部分所示,只有少部分地面点错分。因此 I 类误差相对较高,II 类误差相对较小。

2.2.2 时间效率分析 布料滤波算法执行步骤主要包括格网创建、栅格化、初始格网点位移、边坡后处理

和距离估算分类等。本文中针对以上实验数据集将算法各步运行时间进行统计分析,结果见图8。如图8a所示,算法步骤中格网点牵扯移动比其它步骤花费更多的时间,该步将分别计算格网点受重力作用产生的位移和受格网点相互作用力产生的位移,需要遍历所有的格网点,迭代计算位移,因此消耗了更多的时间。

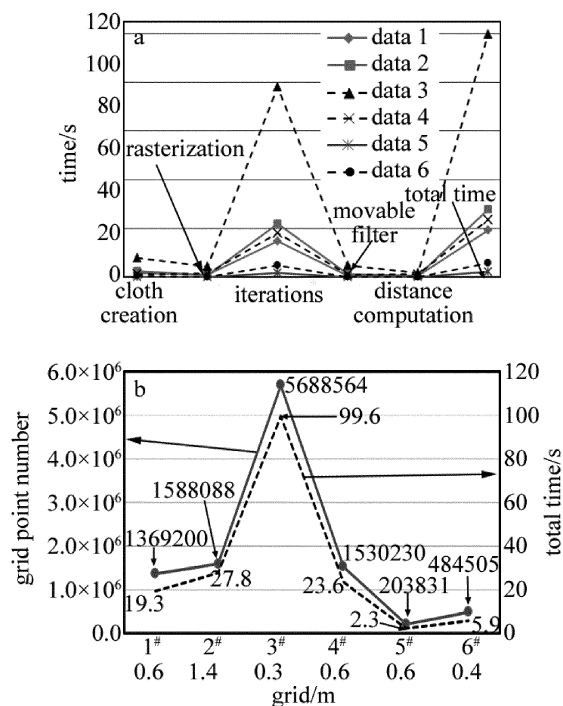


Fig. 8 Statistics of time distribution

a—the running time of each step of the algorithm b—relationship between the running time of the algorithm and the numbers of the constructed grid nodes

针对以上不同的实例数据,本文中滤波的格网分辨率范围设置在0.3m~1.4m之间,格网分辨率越小,布料格网越密集,格网点越多,算法执行耗时越多。如图8b所示,算法执行总时间与格网点数目呈现很强的相关性,完全成正比关系。通过图8也可以得出布料滤波算法执行时间只与格网分辨率和格网点数目有关,不受地形复杂度的影响。另外因为布料格网是被投影到2维平面,格网点数目和点云数量无直接联系,与所设定的格网分辨率、点云密度和点云数据范围大小有关。结合表1和表2所示数据集特性以及图8b,点云数据越密集,滤波所需的格网分辨率越小;格网分辨率增加,运行时间也会大幅度增加。

随着格网点数目增加,算法耗时也不断递增。针对几平方公里范围内的点云数据,实例3数据属于超高密度点云,点云平均密度达到42个/m²,点与点间隔在0.02m以内,激光脚点个数超过两千万个,滤波

所设定的格网分辨率为0.3m,总耗时超过90s。实例数据1、实例数据4属于高密度点云,激光脚点也分别有六百多万个和一千多万个点;实例数据2、实例数据5和实例数据6为稀疏点云,以上数据运行时间均在30s以内,实例数据5和实例数据6甚至在数秒内即可将滤波执行完成,滤波算法效率非常高效。

3 结论

布料滤波算法简单明了、参量较少、运算速度快、自适应性强、滤波性能基本上不受地形条件和地物数量的限制,针对相对平坦区域滤波效果非常好。但该算法对既包含陡峭区域又包含平坦区域的点云数据滤波效果并不理想,滤波后部分陡坡点云会有错分,这和算法本身有关系。因模拟的布料点在陡峭区域所受邻近点牵扯力作用回移的位移较多,使布料点远离地面,造成部分陡坡点被误分为非地面点,但算法经过边坡后处理部分弥补了这一缺点。另外,算法总体效率虽已很快,但若能够加入并行处理机制,则可以使算法效率得到更好的优化。

参考文献

- [1] TANG F F, RUAN Z M, LIU X. Research of filtering method for urban airborne LiDAR data[J]. Laser Technology, 2011, 35(4): 527-530(in Chinese).
- [2] SHAN J, APARAJITAN S. Urban DEM generation from raw lidar data: A labeling algorithm and its performance[J]. Photogrammetric Engineering & Remote Sensing, 2005, 71(2): 217-226.
- [3] LIU C, CHEN H Y, WU H S. Data processing and feature extraction of LiDAR remote sensing[M]. Beijing: Science Press, 2010: 189-195(in Chinese).
- [4] XUELIAN M, NATE C, KAIGUANG Z. Ground filtering algorithms for airborne lidar data a review of critical issues[J]. Remote Sensing, 2010, 2(3): 833-860.
- [5] ZHOU X M. Filtering algorithm of point cloud data for airborne LiDAR[D]. Zhengzhou: The PLA Information Engineering University, 2011: 1-4(in Chinese).
- [6] AXELSSON P. Processing of laser scanner data-algorithms and applications[J]. ISPRS Journal of Photogrammetry & Remote Sensing, 1999, 54(2/3): 138-147.
- [7] ZHANG X H. Theory and method of airborne lidar measurement technology[M]. Wuhan: Wuhan University Press, 2007: 96-104(in Chinese).
- [8] ZHANG X H. Airborne laser scanning altimetry data filtering and object extraction[D]. Wuhan: Wuhan University, 2002: 3-7(in Chinese).
- [9] GEORGE S, GEORGE V. Experimental comparison of filter algorithms for bare-earth extraction from airborne laser scanning point clouds[J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2004, 59(1/2): 85-101.
- [10] MENG X, WANG L, SILVAN-CARDENAS J L. A multi-directional ground filtering algorithm for airborne LiDAR[J]. ISPRS Journal of

- Photogrammetry & Remote Sensing, 2009, 64(1):117-124.
- [11] SUSAKI J. Adaptive slope filtering of airborne LiDAR data in urban areas for digital terrain model (DTM) generation[J]. Remote Sensing, 2012, 4(6):1804-1819.
- [12] ZHANG K, CHEN S C, WHITMAN D. A progressive morphological filter for removing non-ground measurements from airborne LiDAR data[J]. IEEE Transactions on Geoscience & Remote Sensing, 2003, 41(4):872-882.
- [13] MONGUS D, LUKAČ N, ŽALIK B. Ground and building extraction from LiDAR data based on differential morphological profiles and locally fitted surfaces[J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2014, 93(7):145-156.
- [14] MONGUS D, ŽALIK B. Parameter-free ground filtering of LiDAR data for automatic DTM generation[J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2012, 67(1):1-12.
- [15] LI Z L, ZHU Q. Digital elevation model[M]. Wuhan: Wuhan University Press, 2005;44-47 (in Chinese).
- [16] LAI X D. Fundamentals and applications of airborne LiDAR[M]. Beijing: Publishing House of Electronics Industry, 2010;150-201 (in Chinese).
- [17] ZHANG W M, QI J B, XIE D H. An easy-to-use airborne LiDAR data filtering method based on cloth simulation[J]. Remote Sensing, 2016, 8(6):501.
- [18] WEIL J. The synthesis of cloth objects[J]. ACM SIGGRAPH Computer Graphics, 1986, 20(4):49-54.
- [19] PROVOT X. Deformation Constraints in a mass-spring model to describe rigid cloth behaviour[EB/OL]. (2005-05-26) [2017-07-17]. <http://www.cs.cornell.edu/courses/cs667/2005sp/studentSlides/07budsberg.pdf>.
- [20] MOSEGAARD J. Mosegaards cloth simulation coding tutorial[EB/OL]. (2013-10-18) [2017-07-17]. <http://victorfrench.com/cgi-techniques-clothsim.php>.