

自适应步长的 Alpha-shape 表面重建算法

李世林 李红军

(北京林业大学理学院, 北京, 100083)

摘要: 三维物体表面重建在现代临床医学、场景建模和林业测量等方面有着重要应用价值。为了更好地理解三维物体表面形状, 本文先介绍了三维空间离散点集的 Alpha 形状的相关概念。在分析表面重建的 Alpha-shape 算法的基础上, 本文提出一种自适应步长的 Alpha-shape 算法。通过 kd-tree 和 k 近邻平均距离来动态更新 α 值, 使得算法在处理点集密度较大的区域时也能以较少的遍历次数进行表面重建, 从而改善了重建效果并提高了算法运行效率。大量随机数据和现实三维采样数据的实验结果表明, 本文提出的改进算法与原始算法相比, 能大幅度地提高运行效率。

关键词: 表面重建; Alpha 形状; k 近邻平均距离; Alpha-shape 算法

中图分类号: TP391, TP311 **文献标志码:** A

Surface Reconstruction Algorithm Using Self-adaptive Step Alpha-shape

Li Shilin, Li Hongjun

(College of Science, Beijing Forestry University, Beijing, 100083, China)

Abstract: 3D object surface reconstruction has important applications in modern clinical medicine, scene modeling and forestry survey and so on. In order to better understand the reconstruction of 3D object surface, this paper first introduces the concept of the Alpha shape of the 3D discrete point set. Based on the analysis of surface reconstruction algorithm using self-adaptive step Alpha-shape is proposed. The value of Alpha is updated dynamically using the kd-tree structure and the average distance of k -nearest neighbors, so that the algorithm can reconstruct the surface with less number of times when the density of the point set is larger. Thus, the reconstruction effect is improved and the operation efficiency of the algorithm is improved. The experimental results with a large number of random data and realistic 3D scanning data show that the proposed algorithm can greatly improve the efficiency compared with the original algorithm.

Key words: surface reconstruction; Alpha shape; average distance of k -nearest neighbors; Alpha-shape algorithm

引 言

三维物体表面重建是获取三维空间信息建立该物体的三维模型, 在 3D 城市、3D 游戏等场景建模有着重要应用^[1]; 在林业信息化测量中, 利用重建的树冠形状, 准确求解树冠体积、表面积和占地面积等信

息量将会为绿地规划提供精确的数据支持^[2];在现代临床医学中,通过重建出病人的骨骼或者肿瘤块,将会为医学诊断提供辅助手段^[3];由于人类活动的日益影响,许多重要的古建筑古文物都遭受到了不同程度的破坏,对这些古建筑古文物进行数字化三维重建是文物保护的有效手段之一^[4]。三维激光扫描技术的快速发展,使得人们能够快速地获取高精度、高密度的物体表面的三维点集。所以如何准确、快速地进行三维点集的形状分析和表面重建成为当前急需解决的重要问题。

国内外的专家学者已经针对二维和三维的点集形状进行了大量的工作。1977年,文献[5]首次以二维平面点集凸包的推广为基础,来刻画点集的形状;文献[6]给出了点集形状严格的数学定义;文献[7]以Delaunary三角化为工具,采用搜索算法来“雕刻”三维点集的单连通形状,称为雕刻算法;文献[8]改进“雕刻”规则,提出了三维点集表面重建的Alpha-shape算法。这种方法的优势在于根据设定的参数的 α 值大小的不同,体现出点集所具有的不同空洞信息,但是处理密度不均匀的数据时,一般需要对其进行很多的人工干预。1998年文献[9-10]先后提出了一种利用点密度估计来确定 α 参数的方法,避免了繁琐的人工干预,并获得更准确的Alpha形状;文献[11]提出了采用双参数Alpha-shape算法,解决了单参数在表面重建过程中精度和完整性难以兼顾的问题。

本文在前人研究的基础上,考虑到原始Alpha-shape算法其参数值的选取需要过多的人工干预和在处理不均匀点云数据的重建效果不理想这两个问题,对参数值 α 的选取策略进行了改进,设计出了一种依赖于近邻点平均距离的自适应步长的Alpha-shape表面重建算法。

1 三维点集的Alpha形状

文献[6]中给出了二维平面上Alpha形状的严格的数学定义,本文将二维平面上的圆推广到三维空间里的球,从而得到三维空间的Alpha形状的定义。

定义1 对于任意充分小的正实数Alpha,所有包含点集S且半径为 $1/\text{Alpha}$ 的闭合球的交称为点集S的Alpha包,记为positive Alpha-hull,如图1(a)所示。

定义2 对于任意的负实数Alpha,所有包含点集S且半径为 $-1/\text{Alpha}$ 的闭合球的补集的交集称为点集S的Alpha包,记为negative Alpha-hull,如图1(b)所示。

定义3 对于点集S中的一点 p ,存在一个半径为 $1/\text{Alpha}$ 闭合球,该球包含点集S中所有的点,并且点 p 位于球边界上,那么点 p 称为点集S的Alpha-极端。如果3个点 p, q, r 位于同一个闭合球的边界上,那么这3个点称之为Alpha-近邻。

定义4 给定一个三维空间点集S和一个任意实数Alpha,那么会得到一个这样的空间多面体,该多面体的顶点是Alpha-极端,边界面是分别连接Alpha-近邻所得到三角形,这样的空间多面体称为三维空间点集S的Alpha形状。

为了便于理解后续的算法改进,本文提出了 k 近邻平均距离的概念。

定义5 对于三维空间点集S中的任意点 p ,点集S中距离点 p 最近的 k 个点的距离平均值称为点 p 的 k 近邻平均距离。

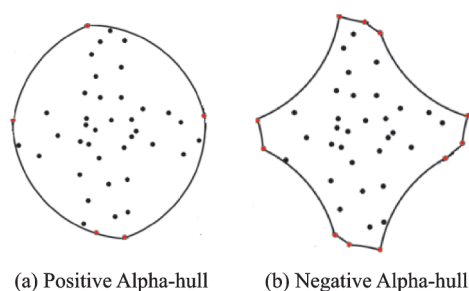


图1 Alpha包示意图

Fig.1 Alpha-hull sketch map

2 Alpha-shape 算法与分析

1994年, Edelsbrunner等提出了三维表面重建的 Alpha-shape 算法^[8]。算法思想是以一个半径固定的小球在点集上“滚动”, 半径由参数 α 确定, 若点集不能使小球通过, 那么不能使小球通过的3个点所构成的三角形就作为点集边界面, 当小球“滚动”结束后可确定整个点集的表面。

2.1 算法流程

Alpha-shape 算法的具体算法流程如下, 其输入为 $n \times 3$ 的三维空间点集 S 矩阵, 每一行的3维向量代表空间中的一个点。输出为 $m \times 3$ 的边界三角形序号集合矩阵 M , 其中 m 代表最终边界面所包含的边界三角形数量, 每一行的3维向量代表一个边界三角形。设三维空间点集 $S = \{p_1, p_2, \dots, p_n\}$ 。

第1步 从点集 S 中任意选择一点 p_1 , 将与之距离小于或等于 2α 的点组成新的点集 S_1 , 从 S_1 中任意选取一组点 $p_2 p_3$, 求出过点 $p_1 p_2$ 和 p_3 且半径为 α 的球的球心 o 和 o' 。

第2步 遍历点集 S_1 , 依次求出其他点到球心 o 和 o' 的距离集合 d 和 d' 。如果 d 和 d' 中有1个集合的距离均大于等于半径 α , 则可以判断出点 $p_1 p_2$ 和 p_3 是边缘轮廓点, 三角形 $p_1 p_2 p_3$ 是边界三角形; 如果 d 和 d' 中都有小于半径 α 的值时, 则可以判断不是边缘轮廓点, 停止遍历, 转到第3步。

第3步 选择 S_1 中的下一组点按步骤1, 2进行判断, 直到 S_1 中的全部点判断结束。

第4步 选择 S 中下一个点按步骤1~3进行判断, 直到 S 中的全部点判断结束, 输出边界三角形矩阵 M 。

2.2 算法效果分析

以一个密度不均匀的花篮模型为例对 Alpha-shape 算法进行效果分析, 其中花篮底部密度较大, 花篮顶部“把手”部分密度较小, 点集模型如图2(a)所示。当 $\alpha = 1$ 时, 实验效果图如图2(b)所示, 由于 α 值过小, 该算法只能计算出部分表面; 加大 α 值, 当 $\alpha = 2.5$ 时, 底部表面重建效果较好, 但是顶部“把手”部分尚未连通, 如图2(c)所示; 继续修改 α 值直到顶部“把手”部分连通, 此时 $\alpha = 3.1$, 最后的花篮模型表面重建效果图如图2(d)所示。

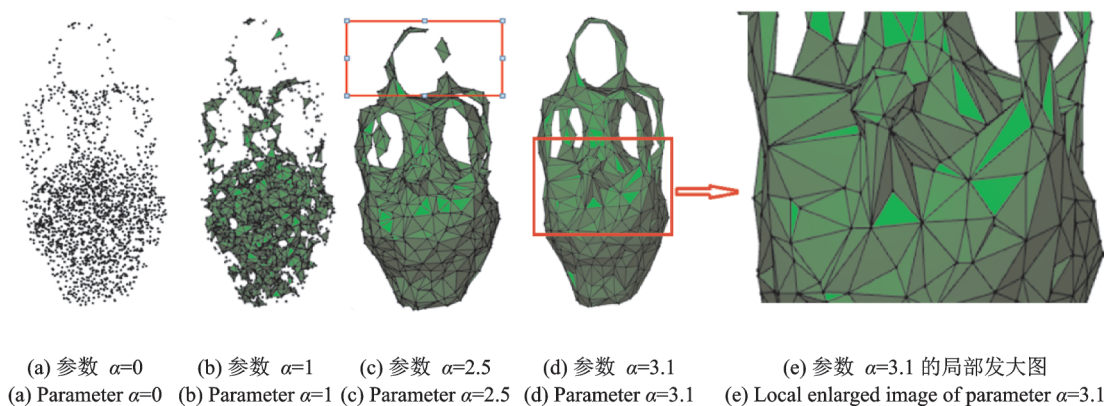


图2 Alpha-shape 算法效果图

Fig.2 Effect diagram of Alpha-shape algorithm

观察表面重建效果图后发现, 在中间“把手”和底部交接的部分, 由于过大的 α 值, 使得表面三角形偏大, 其重建效果过于粗糙, 不能很好地体现物体原有的表面凹凸信息, 如图2(e)所示; 同时, 过大的 α 值导致在处理密度较大的底部区域时需要遍历更多的点, 从而增加了大量的程序运行时间; 再者, 为了

得到完整的物体表面重建结果,该算法在实际重建过程中需要频繁的人工修改 α 参数值。综合以上几点,该算法在处理如林业测量等不均匀的数据时,有适应性不强、重建效果不理想等缺点。为了完善该算法,加大该算法的适应面,本文从 α 参数值的选取规则入手,提出了一种改进的Alpha-shape算法。

3 自适应步长的 Alpha-shape 算法

本文在文献[8]提出的 Alpha-shape 算法的基础上改进了 α 参数值的选取规则,核心思想是利用 kd-tree 计算每个点的 k 近邻平均距离,以该距离作为 α 参数值,采用动态的 α 值来进行后续的操作,称之为自适应步长的 Alpha-shape 算法(Variable Alpha-shape step algorithm, VAS)。

3.1 算法流程

第1步 对点集 S 建立 kd 树,从点集 S 中任意选择一点 p_1 ,计算点集 S 中距离点 p_1 最近的 k 个点的平均距离,记为 α ;将距离点 p_1 小于 2α 的点组成新的点集 S_1 ,从 S_1 中任意选取一组点 $p_2 p_3$,求出过点 $p_1 p_2$ 和 p_3 且半径为 α 的球的球心 o 和 o' 。

第2步 遍历点集 S_1 ,依次求出其他点到球心 o 和 o' 的距离集合 d 和 d' 。如果 d 和 d' 中有一个集合的距离均大于等于半径 α ,则可以判断出点 $p_1 p_2$ 和 p_3 是边缘轮廓点,三角形 $p_1 p_2 p_3$ 是边界三角形;如果 d 和 d' 中都有小于半径 α 的值时,则可以判断不是边缘轮廓点,停止遍历,转到第3步。

第3步 选择 S_1 中的下一组点按步骤1,2进行判断,直到 S_1 中的全部点判断结束。

第4步 取 S 中下一个点按步骤1~3进行判断,直到 S 中的全部点判断结束。

需要说明的是:在第1步中,过点 $p_1 p_2$ 和 p_3 且半径为 α 的球的球心有以下3种情况:(1)点 $p_1 p_2$ 和 p_3 所构成的外接圆半径大于 α ,此时不存在过该3点的球,但是该情况不在算法流程中出现,见3.2节;(2)点 $p_1 p_2$ 和 p_3 所构成的外接圆半径等于 α ,此时球心 o 和 o' 重合;(3)点 $p_1 p_2$ 和 p_3 所构成的外接圆半径小于 α ,此时球心 o 和 o' 关于面 $p_1 p_2 p_3$ 对称。

算法中涉及点集异常处理,主要包括如下3种情况:(1)对于3个及以上的点共线的情况,所形成的边只记距离最远的2个点所形成的边;(2)对于4个及以上的点共圆的情况,所形成的边界面以多边形三角剖分为基础,避免出现边界线交叉的情况;(3)对于4个及以上的点共球的情况,所形成的边界面以多面体三角剖分为基础,避免出现边界面交叉的情况。

3.2 两种算法的效果比较

VAS算法第1步中 kd-tree 的 k 近邻查询的平均时间复杂度为 $O(\log n)$,第2步求距离集合的时间复杂度是 $O(n)$,第3步遍历的时间复杂度是 $O(n^2)$,第4步遍历的时间复杂度是 $O(n^3)$,所以该算法总的时间复杂度为 $O(n^3)$ 。

对于点集密度不均匀的情况下,原始的 Alpha-shape 算法由于其固定的 α 值,在处理点集的稀疏部分时,过小的 α 值将不能反应点集的形状信息,过大的 α 值将导致算法第1步在处理点集密集区域时,所需要遍历的点数很多,直接加大了程序运行负担。而 VAS 算法,通过动态改变 α 值,在保证形状信息的前提下程序运行时间更短。同时,由于其改变 α 值,所以针对不均匀三维点集的表面重建效果,VAS 算法更能反应原始数据的表面信息。图3是原始算法和 VAS 算法针对密度不均匀的花瓶点集的表面重建效果图,原始算法的表面三角形偏大,其重建效果过于“粗糙”,如图3(a)所示。而 VAS 算法的表面三角形较小,其表面重建效果更“细腻”,能更准确地重建出点集表面的凹凸信息,如图3(b)所示。通过图4的对比,改进后的 VAS 算法的表面重建效果要优于原始的 Alpha-shape 算法。

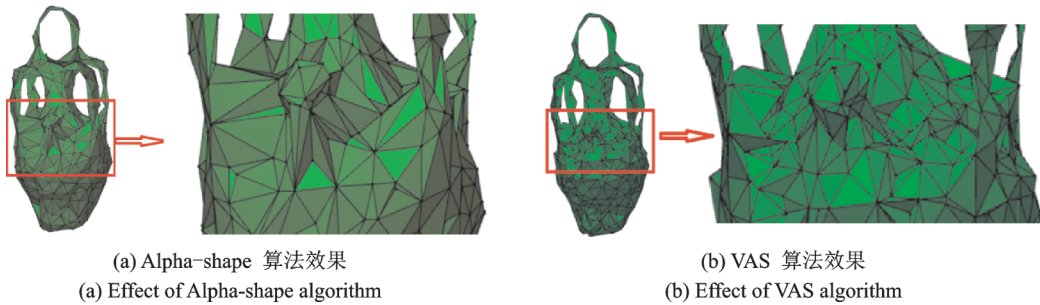


图3 两种算法效果对比

Fig.3 Comparison of two algorithms

4 实验验证

实验针对 Alpha-shape 算法和 VAS 算法的时间效率进行对比,实验数据分为两类:计算机随机生成点集(简称为随机点集)和现实对象采集的三维点集(简称为采样点集)。

4.1 随机点集的生成

实验所用的随机点集分为球形点集和矩形点集,每类又分为均匀轮廓点集、非均匀轮廓点集、均匀内部点集和非均匀内部点集,共 8 组数据,每组的点数为 1 000 个。

球形随机点集的解析式为

$$S = \{(x, y, z) | (x - 1)^2 + (y - 1)^2 + (z - 1)^2 \leq 1\} \tag{1}$$

轮廓点不均匀的点集分布情况为:上下距离轴线 $\pi/4$ 区域内的点数是中间 $\pi/2$ 区域内点数的 3 倍,如图 4(a)所示;包含内部点的不均匀点集分布情况为:下半球区域内的点数是上半球区域内的点数的 3 倍,如图 4(b)所示;而均匀点集是服从均匀分布的随机生成点。

矩形随机点集的解析式为

$$S = \{(x, y, z) | x \in [1, 4], y \in [1, 2], z \in [1, 3]\} \tag{2}$$

轮廓点不均匀的点集分布情况为:上、后、左三面的点数是下、前、右三面的点数的 3 倍,如图 5(a)所示;包含内部点的不均匀点集分布情况为:上半区域内的点数是下半区域内的点数的 3 倍,如图 5(b)所示;而均匀点集是服从均匀分布的随机生成点。



(a) 球形非均匀轮廓点集 (b) 球形非均匀内部点集
(a) Spherical non-uniform contour points set (b) Spherical non-uniform interior points set

图4 球形区域不均匀随机点集

Fig.4 Non-uniform random points in spherical region



(a) 矩形非均匀轮廓点集 (b) 矩形非均匀内部点集
(a) Rectangular non-uniform contour points set (b) Rectangular non-uniform interior points set

图5 矩形区域不均匀随机点集

Fig.5 Non-uniform random points in rectangle region

4.2 参数 k 的选取

关于 k -近邻平均距离中参数 k 的确定可以参照文献[12]的确定方法。而文献[13]认为近邻点 k 通常

在 9 ~ 25 之间选择一个数,即可满足实验基本要求。

由于 k 值的大小将直接影响局部参数 α 的大小,如果 k 值过小,相对应的局部 α 值将过小,此时重建并不完整,如图 2(c)所示;如果 k 值过大,相应的局部 α 值将过大,此时程序的运行时间代价将会加大,同时重建效果并不理想,如图 2(e)所示。本文为了确定合适的参数 k ,对文献[13]中提到的 k 值的选取范围一一进行了重建效果实验,实验数据是 4.1 节的球形非均匀轮廓点集、矩形非均匀轮廓点集和 2.2 节的花篮模型。具体程序运行时间以及重建效果对比如图 6 所示。从图 6 可以看出,直到当 $k \geq 20$ 时,该算法针对 3 种不同点集的重建效果才算完整,同时由于随着 k 的取值变大,其对应的程序运行时间将会增长,并且其增长速度在不断增大。综上所述,本文最终选取 $k = 20$,在实际操作过程中取得了比较好的实验效果。

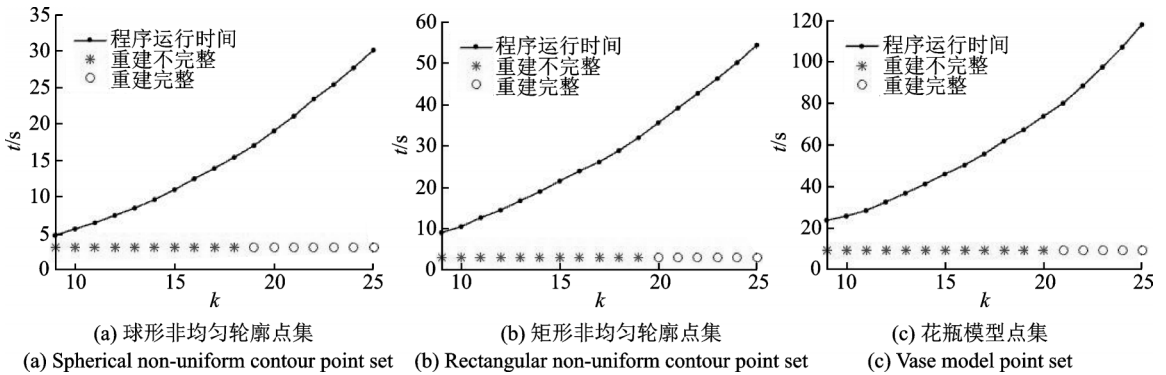


图 6 不同 k 值的程序运行时间及重建情况

Fig.6 Operation time and reconstruction of different k values

4.3 实验结果

上述算法的实验是在台式电脑上进行的,计算机配置是 Intel(R) Core(TM) i7-4790 CPU @ 3.60 GHz 处理器和 4 GB 内存,程序运行环境 MATLAB R2016b。表 1 是两种算法对于不同点集的平均运行时间以及相应的 α 参数值对比,运行时间是重复实验 20 次所求的平均时间。表 1 中,R/S 为矩形/球形;U/N 为均匀/不均匀;I/O 为内部点/轮廓点。例如:RU Ipts 代表矩形均匀包含内部的点集。Alpha-shape 算法的 α 参数值是满足表面重建效果的最小值,其中 α 参数值的改变步长为 0.05;而 VSA 算法的 α 参数值是 k 近邻平均距离求出来的最小 α 值和最大 α 值。

表 1 两种算法针对不同的随机点集的平均运行时间

Tab. 1 Average operation time of two algorithms for different random point sets

点集	Alpha-shape 算法		VAS 算法		时间压缩比/%
	α 值	运行时间/s	α 值(min/max)	运行时间/s	
RU Ipts	0.350	128.58	0.122/0.367	88.32	+31.31
RNIpts	0.400	286.74	0.105/0.435	92.06	+67.89
SUIpts	0.300	142.52	0.107/0.300	97.54	+31.56
SNIpts	0.400	312.52	0.087/0.412	97.09	+68.93
RU Opts	0.250	27.12	0.103/0.284	28.83	-6.31
RNOpts	0.350	72.35	0.054/0.380	57.87	+20.01
SU Opts	0.300	25.32	0.227/0.303	27.60	-9.00
SNOpts	0.350	49.76	0.068/0.374	32.19	+35.31

算法的实验运行效果图如图 7 所示。

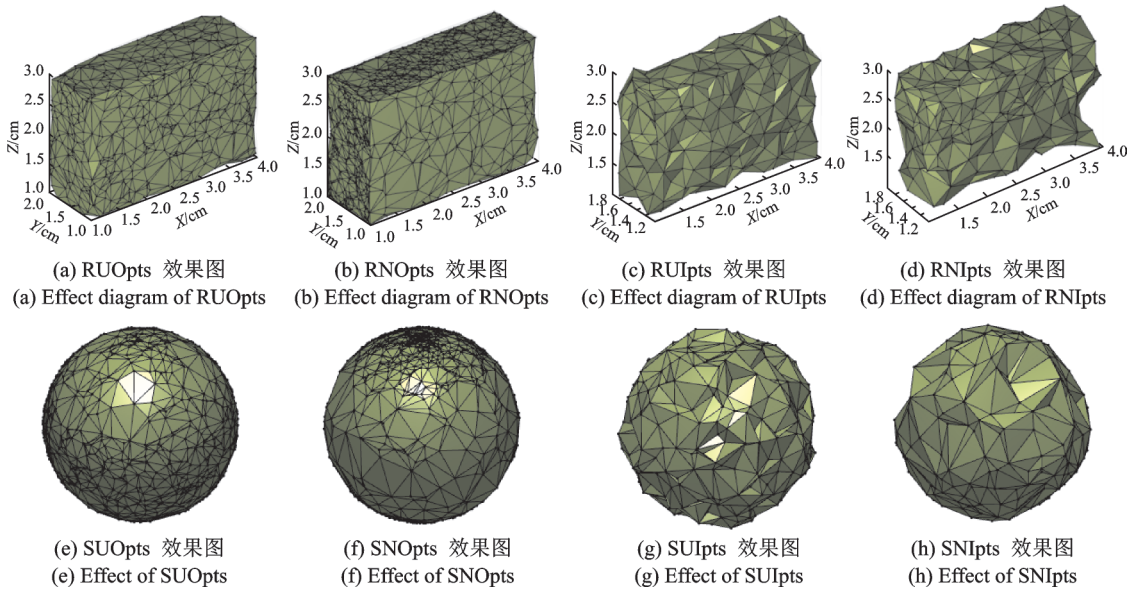


图 7 随机点集的实验效果图

Fig.7 Experimental effect diagram for random point sets

4.4 结果分析

从表 1 中的平均运行时间可以看出,本文提出的 VAS 算法在处理包含内部点的点集时平均运行时间能提高 30% 以上。特别地,在处理密度不均匀的点集时,平均运行时间能提高 60% 以上,这是因为在处理密度较大的区域时,采用较小的 α 值,减少了需要遍历点,从而提高了程序运行时间。但是在处理只有轮廓点的点集时,特别是均匀的点集,本文提出的算法在运行时间上多花费了近 10%,这是因为轮廓点全是边界点,减少遍历点并不能有效地减少程序运行时间,而本文的算法在处理 α 值时需要消耗一定的时间,所以整体上程序的运行时间要长一些。但是对于点集密度不均匀的情况,本文提出的算法在时间效率上能提高 20% 以上。

4.5 激光扫描数据

本文所用的采样点集是两组三维激光扫描的数据,分别是包含树枝的不均匀的树体数据和密度比较均匀的树冠数据,两种算法的平均运行时间对比如表 2 所示,实验运行效果图如图 8 所示。实验表明,本文提出的 VAS 算法在实际采样点集时,针对均匀和不均匀的情况下都能提高 50% 以上的时间运行效率。

表 2 两种算法针对采样点集的运行时间

Tab. 2 Operation time of two algorithms for sampling points

点集(n)	Alpha-shape 算法		VAS 算法		时间压缩比/%
	α 值	运行时间/s	α 值(min/max)	运行时间/s	
树体(8 586)	0.300	2 390.58	0.022/0.291	632.38	+73.55
树冠(3 275)	1.000	603.46	0.432/1.743	289.51	+52.02

5 结束语

对于三维空间离散点集的表面重建问题,本文先介绍了三维空间离散点集的 Alpha 形状的相关概念。在分析表面重建的 Alpha-shape 算法的基础上,本文提出一种改进算法——VAS 算法。该算法改进了原始算法针对点集密度不均匀的情况下需要过多的人工干预 α 参数值的缺点,采用 kd-tree 和 k 近邻平均距离来动态更新 α 值,这就使得该算法在处理点集密度较大的区域时也能以较少的遍历次数进行表面重建,从而为三维空间离散点集的表面重建节省了大量的时间。通过随机数据和现实三维模型采样数据等多组实验的反复验证,实验结果表明本文提出的改进算法与原始算法相比,除了轮廓点随机数据的运行时间比原始算法的时间长,其余数据的时间效率都有大幅度的提高。

未来的工作之一是利用本文提出的算法把 2015 年家里加州大学圣地亚哥分校(University of California)Ery Arias-Castro 教授等的周长计算工作^[14]推广到三维形状的表面积和体积计算;把新算法的体积计算结果与文献[15-16]的结果进行比较;另一项工作是进一步改进表面重建算法效率,采用文献[17]所提出的基于密度的快速聚类方法,将点云数据进行聚类,对每一类固定其参数 α 值,从而缩短确定局部参数 α 值的时间,这样在保证重建效果的基础上可以进一步提高算法的时间效率。

参考文献:

- [1] 刘进. 基于 MVS 点云的城市建筑三维重建应用研究[D]. 南宁:广西大学, 2015.
Liu Jin. Applied research on 3D reconstruction of urban building based on MVS point cloud[D]. Nanning: Guangxi University, 2015.
- [2] 喻焱慎, 云挺, 杨绪兵. 基于激光点云数据的植物器官多维特征分割方法[J]. 数据采集与处理, 2015, 30(5): 1054-1061.
Yu Yaoshen, Yun Ting, Yang Xubing. Multi-dimension feature segmentation method of foliage organs based on laser point cloud data[J]. Journal of Data Acquisition and Processing, 2015, 30(5): 1054-1061.
- [3] 邹豪, Heang K Ty, 刘积仁. 医学图像三维表面模型重建[J]. CT 理论与应用研究, 2000, 9(S1): 23-28.
Zou Hao, Heang K Ty, Liu Jiren. Surface shading of three-dimensional medical objects[J]. CT Theory and Applications, 2000, 9(S1): 23-28.
- [4] 孙家泽, 陈皓, 耿国华. 三维文物点云模型配准优化算法[J]. 计算机辅助设计与图形学学报, 2016, 28(7): 1068-1074.
Sun Jiaze, Chen Hao, Geng Guohua. Registration optimization algorithm for 3D cultural relics point clouds model[J]. Journal of Computer-Aided Design & Computer Graphics, 2016, 28(7): 1068-1074.
- [5] Jarvis R A. Computing the shape hull of points in the plane[C]//Proceedings of the IEEE Computing Society Conference on Pattern Recognition and Image Processing. New York: IEEE, 1977: 231-241.
- [6] Edelsbrunner H, Kirkpatrick D, Seidel R. On the shape of a set of points in the plane[J]. IEEE Transactions on Information Theory, 1983, 29(4): 551-559.
- [7] Boissonnat J D. Geometric structures for three-dimensional shape representation[J]. ACM Transactions on Graphics, 1984, 3(4): 266-286.
- [8] Edelsbrunner H, Mücke E P. Three-dimensional Alpha shapes[J]. ACM Transactions on Graphics, 1994, 13(1): 43-72.
- [9] Teichmann M, Capps M. Surface reconstruction with anisotropic density-scaled Alpha shapes[C]//Proceedings of Visualization. [S.l.]: IEEE Xplore, 1998: 67-72.
- [10] Xu X, Harada K. Automatic surface reconstruction with Alpha-shape method[J]. The Visual Computer, 2003, 19(7): 431-443.

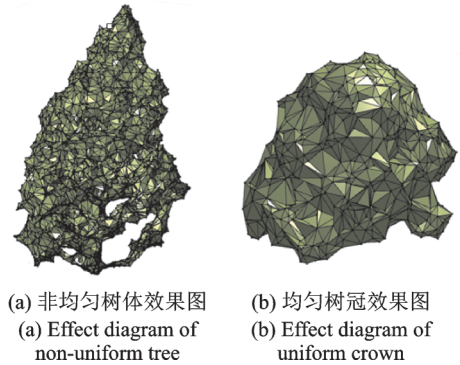


图8 采样点集的实验效果图

Fig. 8 Experimental effect diagram for sampling points

- [11] 李云帆, 谭德宝, 高广, 等. 双阈值 Alpha Shapes 算法提取点云建筑物轮廓研究[J]. 长江科学院院报, 2016, 33(11): 1-4.
Li Yunfan, Tan Debao, Gao Guang, et al. Extraction of building contour from point clouds using dual threshold Alpha shapes algorithm[J]. Journal of Yangtze River Scientific Research Institute, 2016, 33(11): 1-4.
- [12] Weinmann M, Jutzi B, Hinz S, et al. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers[J]. ISPRS Journal of Photogrammetry & Remote Sensing, 2015, 105: 286-304.
- [13] 李红军, 刘欣莹, 张晓鹏, 等. 局部形状特征概率混合的半自动三维点云分类[J]. 浙江大学学报:理学版, 2017, 44(1): 1-9.
Li Hongjun, Liu Xinying, Zhang Xiaopeng, et al. A semi-automatic 3D point cloud classification method based on the probability mixture of local shape features[J]. Journal of Zhejiang University: Science Edition, 2017, 44(1): 1-9.
- [14] Arias-Castro E, Casal A R. On estimating the perimeter using the alpha-shape[J]. Annales de l'Institut Henri Poincaré, Probabilités et Statistiques, 2017, 53(3): 1051-1068.
- [15] Zhu C, Zhang X, Hu B G, et al. Reconstruction of tree crown shape from scanned data[C]//International Conference on Technologies for E-learning & Digital Entertainment, Edutainment 2008, Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2008: 745-756.
- [16] Ariascastro E, Pateiro López B, Rodriguezcasal A. Minimax estimation of the volume of a set under the rolling ball condition[J]. Journal of the American Statistical Association, 2018: 1-37.
- [17] 张晓, 张媛媛, 高阳, 等. 一种基于密度的快速聚类方法[J]. 数据采集与处理, 2015, 30(4): 888-895.
Zhang Xiao, Zhang Yuanyuan, Gao Yang, et al. Fast density-based clustering approach[J]. Journal of Data Acquisition and Processing, 2015, 30(4): 888-895.

作者简介:



李世林(1992-),男,硕士研究生,研究方向:计算几何,
E-mail: lishilin_2015 - @bjfu.edu.cn。



李红军(1969-),通信作者,男,博士,副教授,研究方向:计算机图形学和计算几何等, E-mail: lihongjun - 69@bjfu.edu.cn。

(编辑:刘彦东)