

基于点云内骨架的分割算法

李仁忠*, 刘哲闻, 刘阳阳

西安工程大学电子信息学院, 陕西 西安 710048

摘要 提出一种有效的三维点云骨架分割的方法, 分割后的结果可用于三维点云物体识别和分类。利用稳健性较强的 L_1 -中心骨架算法对点云数据进行骨架提取, 可得到一系列骨架点; 利用基于八叉树的区域增长分割方法对已经得到的骨架点进行分割, 选取法向量和残值作为判定标准; 利用 OpenGL 库编程把分割出的各个部分进行骨架连线。对多种形状的点云数据(包括动物模型、植物模型、人体模型、字母模型)进行实验, 该方法均得到较好的结果。

关键词 成像系统; 三维点云; 骨架提取; 区域增长; 点云分割

中图分类号 TP391.41

文献标识码 A

doi: 10.3788/LOP56.221102

Segmentation Algorithm Based on Point Cloud Skeleton

Li Renzhong*, Liu Zhewen, Liu Yangyang

School of Electronics and Information, Xi'an Polytechnic University, Xi'an, Shaanxi, 710048, China

Abstract This study presents an effective method for segmenting the skeleton of a three-dimensional point cloud. Segmentation results can be used in point cloud object recognition and classification. First, an L_1 -central skeleton algorithm with strong robustness is used to extract the skeleton of point cloud data, and a series of skeleton points is obtained. Then, the skeleton points are segmented by using the octree based region growing segmentation method. The selected criteria are normal vectors and residuals. Finally, OpenGL is used to connect the segmented parts. Numerous experiments are conducted with point cloud data of various shapes, such as animal, plant, human, and alphabetic models, and good results are obtained.

Key words imaging system; three-dimensional point cloud; skeleton extraction; region growth; point cloud segmentation

OCIS codes 110.3010; 150.6910; 110.288; 100.6950

1 引言

三维点云分割一般根据三维点云信息的空间特征、几何特征或纹理特征等对其进行区分, 使得同一片区域内的点具有相同的属性, 分割后的结果可用于识别和分类等^[1]。受点云数据规模较大、采样密度不均匀以及点云数据几何结构不清晰等因素的影响, 点云分割具有一定的挑战性。比较有效的分割方法是联合点云内部骨架和外部特征点进行分割。

对于三维点云的骨架提取, 目前常用的方法有:

1) 对称轴分析法。该方法由 Blum^[2] 提出, 以视觉特

征为基础, 通过处理点云的中轴信息来提取骨架。该方法多用于处理多边形模型或参数曲线曲面模型, 使用此方法能够获取精确度较高的骨架。对于复杂的动植物模型, 该方法的计算比较复杂, 精度较低, 很难在实际应用中实现^[3]。2) 基于拉普拉斯(Laplace)收缩的三维点云骨架提取方法。该方法由 Cao 等^[4] 提出, 其先对点云进行 Laplace 算子收缩, 从外围向内部迭代收缩从而提取出骨架。这种方法容易在局部点云上出现过收缩。此外, 该方法的准确性很大程度上依赖于如何设定 Laplace 算子的参数, Laplace 算子参数的细小偏差都将导致点云

收稿日期: 2019-04-22; 修回日期: 2019-05-07; 录用日期: 2019-05-17

基金项目: 中国纺织工业联合会科技指导性项目(2017071)、陕西省高校科协青年人才托举计划(20180115)

* E-mail: lirenzhong@xpu.edu.cn

的分割结果达不到预期要求。3) 基于数学结构上的分析方法。这类方法中比较典型的有 Reeb 图法和 Voronoi 图法。Reeb 图法是由 Shinagawa 等^[5]提出的, 该方法利用输入数据点上的标量函数, 对数据边缘点与点之间的关系进行映射, 构造出一种数学结构关系图。Dey 等^[6]则是利用 Voronoi 图的特性对点云进行骨架提取, 但是在处理一些有噪声的点云数据时 Voronoi 图法的效果一般。4) 基于 L_1 -中心骨架提取方法。该方法由 Huang 等^[7]提出, 主要是利用 L_1 -中值定理进行骨架收缩, 可以很好地避免野点对骨架提取过程的干扰, 具有很好的稳健性。

一些研究团队对骨架分割进行了积极探索。Zhan 等^[8]提出一种基于颜色的点云分割方法。该方法是一种依据色度相似性和空间接近性的分割方法, 具有一定的局限性。Wang 等^[9]提出了一种基于聚类的分割方法。该方法首先将输入点云的正态线映射到高斯球上, 然后使用平移聚类算法将其划分为组, 最后基于欧氏聚类算法处理数据。该方法能够分割出建筑物、树木、路灯和其他物体, 其主要是用于处理大型城市建筑物数据。Rabbani 等^[10]提出了一种使用平滑约束度分割点云的方法。该方法使用局部表面法线和点连接, 使用 k -邻近或固定距离实现。不足之处是, 该方法需要手动调整参数。Wang 等^[11]提出了一种利用相似群建议网络(SGPN)的语义分割方法。该方法在各种三维场景下的实验效果都很好, 但是该方法还不适用于结构复杂的数据。在 2015 年的国际信息处理会议(ICIP)会议上, Vo 等^[12]介绍了一种在图像分割领域的区域增长算法。Vo 等使用基于八叉树(octree)的区域生长算法对点云数据进行处理, 首先把点云数据分割成子数据块, 然后对得到的不完整的数据进行细化处理, 最终得到了平滑的骨架。对于数据结构较为简单的骨架来说, 该方法操作简单, 效果也较好。本文将就如何进行点云的内部骨架分割展开研究。

2 算法流程图

本文算法的基本思路是: 首先利用 L_1 -中心骨架算法提取点云数据的骨架点, 然后利用基于 octree 的区域增长算法对其进行分割, 最后利用 OpenGL 编程进行骨架点连线。算法流程图如图 1 所示。

3 点云骨架提取

3.1 点云随机下采样

通过随机下采样把采样点收缩为骨架点, 从而

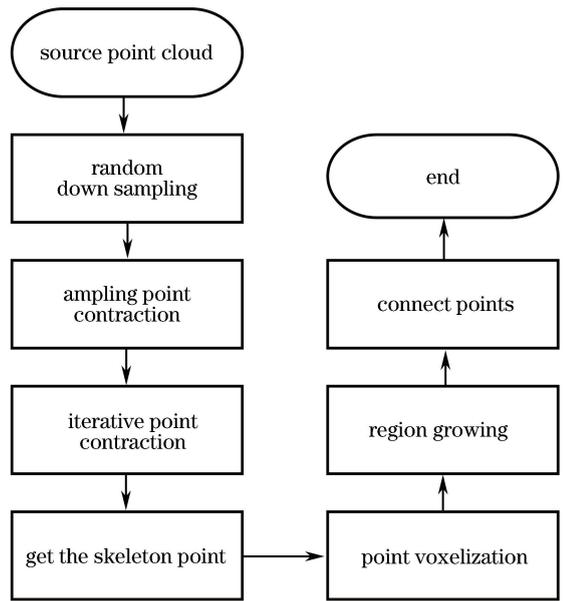


图 1 算法流程图

Fig. 1 Flow chart of algorithm

生成骨架点。一般来说, 采样点越多, 稳健性越好, 但是计算量也越大。实验发现采样点过多或过少对结果的影响均不大, 这里采样点的数量设置为总数的 10%。对于采样点的位置, 可以通过洗牌算法来选定采样点的位置。

为直观展示洗牌算法的步骤, 以表 1 为例加以说明。设采样点的数量 $n=8$, 即初始采样点的顺序为 1, 2, 3, 4, 5, 6, 7, 8。

表 1 洗牌算法

Table 1 Fisher-Yates algorithm

Range	Roll	Scratch	Result
[1,8]	6	1 2 3 4 5 8 7	6
[1,7]	2	1 7 3 4 5 8	2 6
[1,6]	6	1 7 3 4 5	8 2 6
[1,5]	1	5 7 3 4	1 8 2 6
[1,4]	3	5 7 4	3 1 8 2 6
[1,3]	3	5 7	4 3 1 8 2 6
[1,2]	1	7	5 4 3 1 8 2 6

表 1 中, 第一行表示, 从 1 到 8 中随机选择一个数, 得到 6, 则交换当前数组中第 8 和第 6 个数; 第二行表示, 从剩下的数中随机选择一个数, 得到 2, 则交换当前数组中第 7 和第 2 个数; 第三行表示, 剩下的数中随机选择的刚好是 6, 这意味着只需把当前线性表中的第 6 个数留在原位置, 接着进行下一步; 以此类推, 直到整个排列完成。得到的采样点的位置为 7, 5, 4, 3, 1, 8, 2, 6。

3.2 采样点收缩形成骨架

得到一系列采样点后利用 L_1 -中心骨架算法进

行骨架收缩提取。骨架收缩提取的基本思想是通过选取一系列点的中值而不是平均值进行收缩,产生新的骨架点,不停地迭代并重新将其分配到所在区域的输入点的中心。直接应用 L_1 -中值算法往往会产生稀疏分布^[7],造成结果中的一些中心点产生一团点簇。为了避免这种现象,首先需要进行条件正则化,然后进行骨架收缩,最后基于密度进行加权。对于有缺失的点云数据,还需增加一步——定义中心位置,才能完成整个过程。

1) 条件正则化

给出一系列乱序的点集 $\mathbf{Q} = \{\mathbf{q}_j\}_{j \in J} \subset \Omega^3$, 以及从 \mathbf{Q} 中随机下采样得到的点集 $\mathbf{X} = \{\mathbf{x}_i\}_{i \in I} \subset \Omega^3$, 利用最优公式可得出系列骨架点集, 其中最优化公式为

$$\operatorname{argmin}_{\mathbf{X}} \sum_{i \in I} \sum_{j \in J} \|\mathbf{x}_i - \mathbf{q}_j\| \theta(\|\mathbf{x}_i - \mathbf{q}_j\|) + R(\mathbf{X}), \quad (1)$$

式中: I 和 J 分别为点集 \mathbf{Q} 和 \mathbf{X} 中的点数; $R(\mathbf{X})$ 代表点 X 的邻域点分布度量; $\theta(r)$ 是高斯权重函数, $\theta(r) = \exp[-4(r/h)^2]$ 。

当(1)式中的能量梯度为 0 时, 代入 $\mu = \frac{\gamma_i \sum_{i' \in I(i)} \beta_{i'}}{\sigma_i \sum_{j \in J} \alpha_{ij}}$, 其中 $\alpha_{ij} = \frac{\theta(\|\mathbf{x}_i - \mathbf{q}_j\|)}{\|\mathbf{x}_i - \mathbf{q}_j\|}$, $\beta_{i'} = \frac{\theta(\|\mathbf{x}_i - \mathbf{x}_{i'}\|)}{\|\mathbf{x}_i - \mathbf{x}_{i'}\|^2}$, 即可得到

$$(1 - \mu\sigma_i)\mathbf{x}_i + \mu\sigma_i \frac{\sum_{i' \in I(i)} \mathbf{x}_{i'} \beta_{i'}}{\sum_{i' \in I(i)} \beta_{i'}} = \frac{\sum_{j \in J} \alpha_{ij} \mathbf{q}_j}{\sum_{j \in J} \alpha_{ij}}, \quad (2)$$

式中: γ_i 是控制输入点的引力和采样点的斥力之间的平衡参数; σ_i 为分布度量。

在(2)式中, 根据线性代数可得到关于 \mathbf{X} 的解集为 $\mathbf{X} = \mathbf{A}^{-1} \mathbf{BQ}$, 然后进行迭代, 其中 $\mathbf{X}^k = \{\mathbf{x}_i^k\}$, 且 $k=0, 1, \dots$, 可得到

$$\mathbf{x}_i^{k+1} = \frac{\sum_{j \in J} \alpha_{ij} \mathbf{q}_j}{\sum_{j \in J} \alpha_{ij}} + \mu\sigma_i^k \frac{\sum_{i' \in I(i)} (\mathbf{x}_i^k - \mathbf{x}_{i'}^k) \beta_{i'}^k}{\sum_{i' \in I(i)} \beta_{i'}^k}. \quad (3)$$

接下来, 通过设置参数的取值即可完成条件正则化。

2) 迭代收缩骨架

给定邻域大小 h , 即可生成一系列点 $\mathbf{X} = \{\mathbf{x}_i\}$, $i \in I$, 这些点代表了局部邻域的 L_1 中值点。在一些情况下, 这些点可以直接代表骨架点。在复杂数据的条件下, 这些点仅代表骨架的分支点, 还需要进行迭代收缩才能代表骨架点。为了确定标记的点是否为收缩后的点, 先计算所有的非分支点 \mathbf{x}_i , 然后

对这些点进行平滑处理并剔除其中的噪声点。

3) 基于密度的加权

对于一些点云密度不均匀的数据, 局部中心往往会偏向于密度较高的区域, 导致输出的点云骨架点的位置在切面方向上发生位移, 因此需要对密度进行加权。

将输入的点云集中的每个点 \mathbf{q}_j 的加权局部密度定义为

$$d_j = 1 + \sum_{j' \in J \setminus \{j\}} \theta(\|\mathbf{p}_j - \mathbf{p}_{j'}\|). \quad (4)$$

将其代入到(3)式中, 即可得到

$$\mathbf{x}_i^{k+1} = \frac{\sum_{j \in J} \alpha_{ij} \mathbf{q}_j / d_j}{\sum_{j \in J} \alpha_{ij} / d_j} + \mu\sigma_i^k \frac{\sum_{i' \in I(i)} (\mathbf{x}_i^k - \mathbf{x}_{i'}^k) \beta_{i'}^k}{\sum_{i' \in I(i)} \beta_{i'}^k}. \quad (5)$$

通过第一项中的加权局部密度可以有效地减小密度不均匀区域对收缩结果的影响。此外, 局部点的密度由函数 $\theta(\|\mathbf{p}_j - \mathbf{p}_{j'}\|)$ 决定。

4) 再次定义中心位置

在正常条件下, 生成的 L_1 中心是在输入点云的中心点附近, 但是如果输入点云有缺失, 可以通过重新定义中心位置来进行收缩骨架。为了增强系统的稳健性, 在进行下采样后分别在每个分支上对采样数据重新对齐, 然后对弯曲的分支作平滑处理, 最终重新定义中心位置。

4 对骨架点进行分割

4.1 点云体素化

将输入点云体素化, 利用八叉树算法覆盖输入点云^[13], 如图 2 所示。选择基于指针区域的八叉树进行实验, 这样做可达到如下 3 个目的: 1) 把点云数据组织好(索引数据); 2) 对点 P 进行光栅化处理(简化数据); 3) 为特征点估计定义邻域点。在这种八叉树结构中, 输入点云所占空间上的所有节点都被分解成 8 个一样的子节点, 并且在这个空间上的所有体素的几何信息是一致的。

体素化确定了在图 2(c) 中的包围 P 的最小立方体边界框, 称之为 0 级节点, 对应图 2(b) 中的 level 0。0 级节点再分裂成 8 个小体素, 重复上述步骤直至满足要求的判别条件。目前, 已有的判别条件有 3 种方案: Ayala 等^[14] 于 1985 年提出的最小体素值判定; Pulli 等^[15] 提出的预先确定的最大深度树方案以及 Wang 等^[16] 于 2005 年提出的体素最大采样数量方案。采用残差阈值 r_{th} 和最小体素值 d_{min}

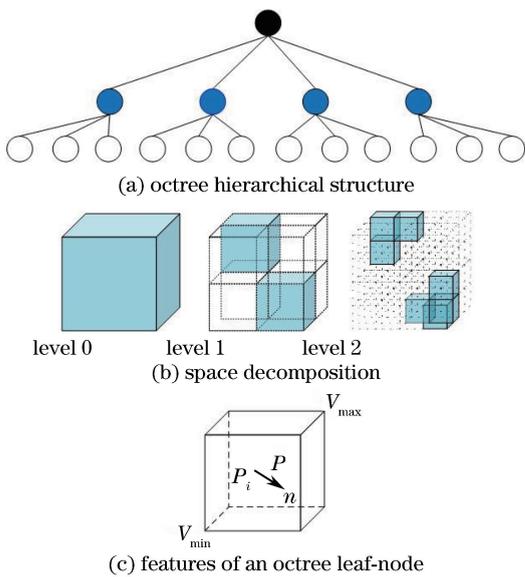


图2 八叉树实例

Fig. 2 Example of octree

判定的方法作为判别条件。先把0级节点分割成一个特定的体素大小,并将其定义为 d_U ,即可用体素网格代表输入点云的局部表面特征。初始分割后,非空的体素将被继续分割,直至满足判别要求。由于在使用此方法过程中,大体素会出现在平滑区域以内,小体素出现在边缘区域,故该方法可以显著地减少体素数量,减少迭代步骤,提高分割效率。

4.2 点云特征估计

研究人员除了可从三维激光扫描仪中获取一些诸如几何坐标^[17]、RGB颜色信息等,还可从点云数据的分布中获取一些有价值的信息,例如围绕中心的各个方向的数据以及每个主方向上数据点的方差。在三维点云中,通常把最小的方差对应的方向称为平面法向量的近似值。本研究所使用的区域增长算法主要是利用法向量的特征进行信息处理,这些信息都根据体素中的点进行计算。利用基于主成分分析(PCA)算法对以上法向量特征进行分析。

4.3 区域增长

利用基于八叉树的区域增长算法逐步将有相似特征的体素分组,在分组的过程中会产生一些体素簇,每一个体素簇都代表着平滑表面的一个部分。利用八叉树进行区域增长有以下3点好处:1)先对点云进行体素化,减少了后续进行区域增长的工作量;2)相邻体素点涉及到的邻域算法有助于减少对原始点云数据的 k -邻域搜索^[18];3)体素网格本身有着一定的空间结构,允许快速并相对容易地搜索相

邻体素。该算法的核心是把所有法向量相近的点全部提取出来,形成点簇,将每一个点簇都可看作是相似的一部分,即被分割出来。该算法的具体步骤如下:

1) 输入原始点云集,将整个点云按照每个点的曲率进行排序,搜索出曲率最小的点,并将其添加为种子点。

2) 搜寻种子点邻域的点,若邻域内某一点的法向量与种子点的法向量的差值小于一定的范围,则计算该点的曲率。若该点的曲率小于设定的阈值,则该点属于当前的部分。

3) 从原始点云中去除已经通过第二步测试后的点。

4) 根据点云中点的数量设置最小点云簇(分割出的最小数量)和最大点云簇(分割出的最大数量)。

5) 重复以上步骤,原始点云将会被分割成满足以上最小和最大点云簇的部分。对每个部分进行染色,使用不同颜色对不同部分加以区分。

6) 当剩余点的数量小于设置的最小点云簇时,停止分割。

5 实验结果及分析

实验所用计算机的CPU为Intel Core i5-7300HQ,CPU的主频为2.50 GHz,电脑内存为8 GB,操作系统为Windows10 64位。实验所用软件为Visual Studio 2013 Visual C++ 控制台应用程序,开源库为OpenGL,开源点云库为PCL1.8.0。首先利用 L_1 -中心骨架算法对输入的点云数据进行骨架点的提取,然后利用基于八叉树的区域增长算法对已经提取出的骨架点进行有效的分割,其中选取曲率阈值为0.7。因为之前的步骤都是对点进行处理,输出的分割结果也将都是点的集合,所以需要利用OpenGL库对这些点进行连线,以达到更好的显示效果。采用树枝(有叶)模型、树枝(无叶)模型、字母模型、人物模型以及动物模型对本文算法的稳健性进行验证。如图3~图8所示,其中树枝(有叶)模型含有40551个点,树枝(无叶)模型含有11004个点,字母模型含有7029个点,人物模型含有43545个点,珊瑚模型含有14207个点,动物模型含有25713个点。在骨架点提取部分,图3~图7的结果对原始三维点云的骨架点提取的效果最好。从视觉上看,图3~图7的结果更靠近点云骨架的位置,也很好契合了输入点云的拓扑结构。即使是对于有大量噪声点的字母模型(图5),本文

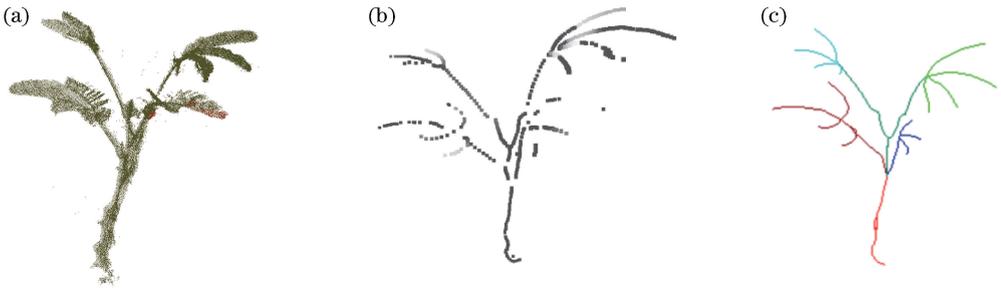


图3 树枝(有叶)模型。(a)原数据;(b)骨架点;(c)结果

Fig. 3 Tree (with leaf) model. (a) Raw data; (b) skeleton point; (c) result

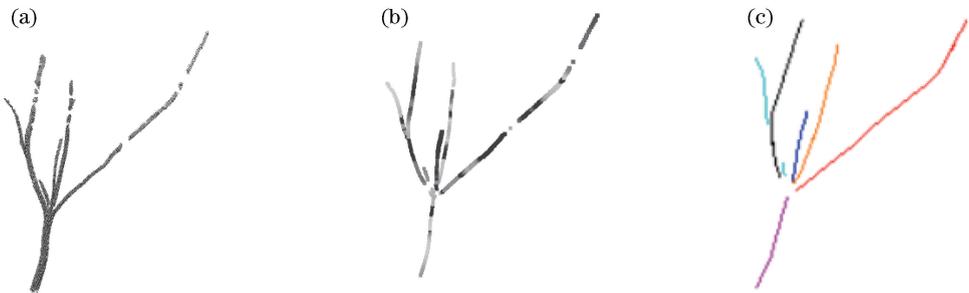


图4 树枝(无叶)模型。(a)原数据;(b)骨架点;(c)分割结果

Fig. 4 Tree (without leaf) model. (a) Raw data; (b) skeleton point; (c) result

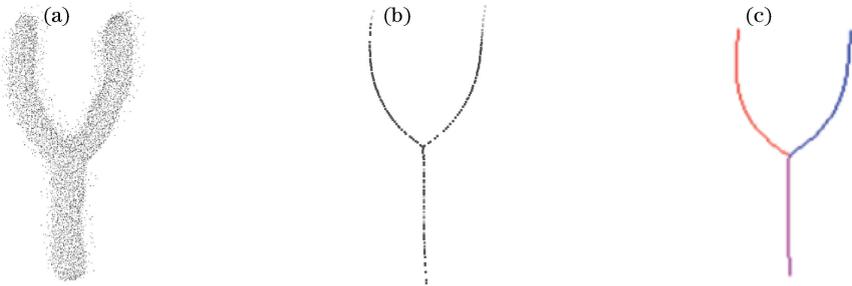


图5 字母模型。(a)原数据;(b)骨架点;(c)分割结果

Fig. 5 Alphabet model. (a) Raw data; (b) skeleton point; (c) result

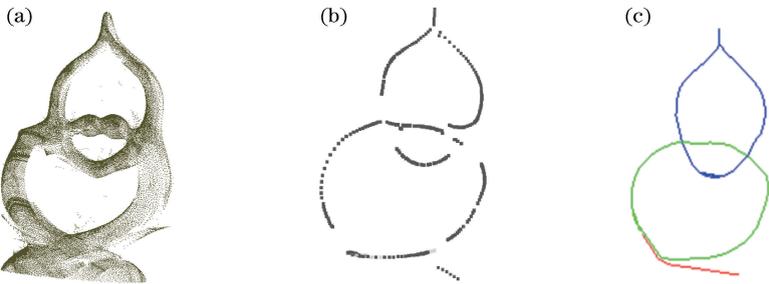


图6 人体模型。(a)原数据;(b)骨架点;(c)分割结果

Fig. 6 People model. (a) Raw data; (b) skeleton point; (c) result

算法也取得了很好的效果,能够准确地表达出字母“Y”的特征,这表明该算法具有较强的稳健性。但是对于图8的动物模型,本文算法的结果还有些许瑕疵。实验结果显示,图8中的恐龙的尾部和身体、四肢和身体相结合部分的骨架点过于稀

疏。这主要是由于该动物模型的身体部分的横截面较大,在进行骨架点提取时本文采用了一种收缩的方法,收缩后的骨架点会在整个身体的中心位置产生。同理,恐龙四肢的骨架点也是在其中中心位置。本来恐龙四肢是和身体相连接,但是本

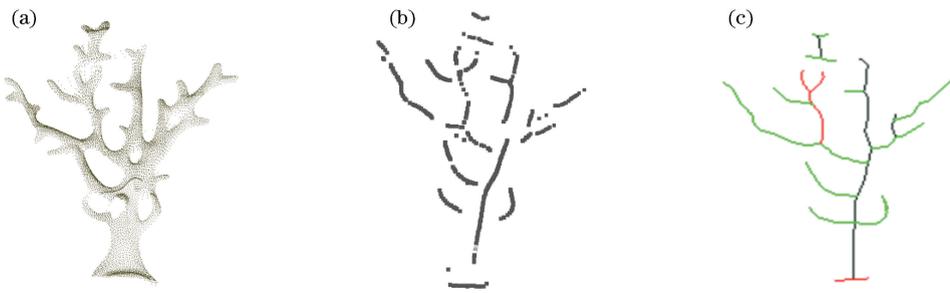


图7 珊瑚模型。(a)原数据;(b)骨架点;(c)分割结果

Fig. 7 Coral model. (a) Raw data; (b) skeleton point; (c) result



图8 动物模型。(a)原数据;(b)骨架点;(c)分割结果

Fig. 8 Animal model. (a) Raw data; (b) skeleton point; (c) result

研究采取的收缩方法增大了这两部分的距离,导致最终提取出的恐龙骨架的四肢和身体产生了较为明显的空隙。需在下一步的工作中继续改进算法,解决这类问题。

从分割层面来讲,图3和4的植物模型、图5的字母模型、图7的珊瑚模型以及图8的动物模型的分割结果都比较好,这主要是因为本研究所用到的分割方法是根据点云的曲率参数进行区域增长分割。若是前一个点的曲率和后一个点的曲率相近,则会两个点划分为同一区域。对于这些不同部分曲率差异较大的点云模型来讲,本文算法均能取得较好分割结果。然而,对于图6的双人人体模型,如果从人眼视觉层面来讲,分割的结果是有偏差的。图6的人体模型展现的是两个人头对头,手对手,分割的结果显然不符合常理,如图6(c)所示,这是由于算法是按照点的曲率进行分割的,此过程过于复杂。实际上,把图6的人体模型从中间直接进行切

割,效果应会更好。在实际应用中,要先根据模型的结构特点选择合适的方法。表2中显示了各算法的运行时间,从表中可知,对于图3的树枝模型(有叶)与图6的人体模型,算法的运行时间接近;对于图7的珊瑚模型和图8的动物模型,算法的运行时间接近;对于图4的树枝模型(无叶)与图5的字母模型,算法的运行时间相近。结合每个模型点的数量可知,在提取骨架点的过程中,影响算法运行时间的决定性因素是点云模型点的数量,与点云模型自身的空间结构关系不大。图3和图4两个模型虽然结构上相似,但是最终算法的运行时间相差很大,图7和图8两个模型尽管结构大不相同,但是因为两个模型点的数量相近,所以算法的运行时间也相近。从整体上看,上述模型处理的时间都有些缓慢,反观分割时间,几组数据的分割时间均是毫秒量级的,速度较快。如何提高骨架点的提取速度将是下一步要继续进行的工作。

表2 算法的运行时间

Table 2 Running time of algorithm

Model	Tree(with leaf)	Tree(without leaf)	Alphabet	People	Coral	Animal
Skeleton time /s	9.147	3.964	1.453	8.733	3.054	3.209
Segmentation time /ms	6.077	8.568	2.318	2.647	2.896	2.985

此外,将本文算法与传统的 Laplace 收缩方法进行对比,结果如图9所示。通过图9的结果可知,应用本文算法的图9(b)和(e)相比于应用 Laplace 收缩方法的图9(c)和(f)能更好地保持骨架点的完

整性。本文算法能够有效地保持原有数据的几何拓扑结构,并且提取出的骨架点更能贴合原始数据的自然骨架,所得结果符合人眼的视觉习惯,不会产生错位的问题,准确性更好。

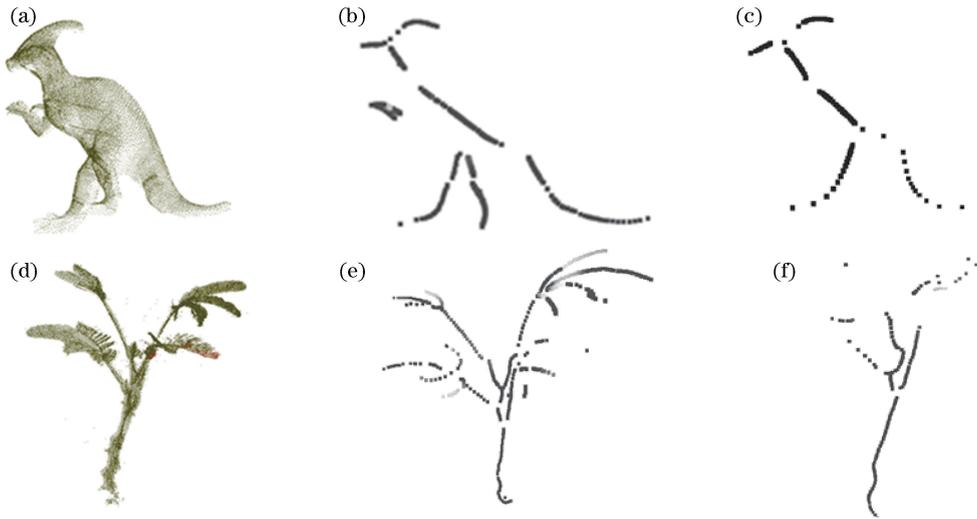


图9 骨架点提取对比实验。(a)(d)原始数据;(b)(e)本文算法;(c)(f)拉普拉斯算法

Fig. 9 Comparative experiment of skeleton point extraction. (a)(d) Raw data; (b)(e) our method; (c)(f) Laplacian method

6 结 论

针对三维点云内骨架分割计算量大,且已有的方法存在一定的局限性的问题,提出一种针对点云内骨架的分割方法。利用 L_1 -中心算法对输入点云进行点云数据的内骨架点提取,然后利用基于八叉树的区域增长算法对提取后的点云骨架点进行分割,最后利用 OpenGL 库完成对骨架点的连线。实验结果表明,对于环状点云模型或是比较细的点云模型(模型的切面半径较小),本文算法对点云骨架点的提取效果较好。对于不同位置切面半径变化较大的点云模型,本文算法还需继续改进。从分割的层面上讲,对于曲率变化较为明显的点云数据,应用本文的分割方法能够取得较好的分割效果。该方法可以快速准确地分割点云数据。本文所提出的方法利用了体素模型,可以简化数据、索引数据以及定义计算局部表面属性的邻域组,但是体素的大小仍需要手动决定。另外,对于各点曲率差值较小的模型,本文所提出的算法也不能取得最佳效果。在下一步的工作中,需对该算法进行进一步改进。

参 考 文 献

- [1] Reniers D, Telea A. Skeleton-based hierarchical shape segmentation [C] // IEEE International Conference on Shape Modeling and Applications 2007 (SMI'07), June 13-15, 2007, Lyon, France. New York: IEEE, 2007: 9855063.
- [2] Blum H. Biological shape and visual science (part I) [J]. Journal of Theoretical Biology, 1973, 38(2): 205-287.
- [3] Li T. Medial axis computation for algebraic curves and related issues [D]. Hangzhou: Zhejiang University of Technology, 2011: 4-6.
李涛. 代数边界曲线的中轴计算及其相关问题[D]. 杭州: 浙江工业大学, 2011: 4-6.
- [4] Cao J J, Tagliasacchi A, Olson M, *et al.* Point cloud skeletons via Laplacian based contraction [C] // 2010 Shape Modeling International Conference, June 21-23, 2010, Aix-en-Provence, France. New York: IEEE, 2010: 187-197.
- [5] Shinagawa Y, Kunii T L. Constructing a Reeb graph automatically from cross sections [J]. IEEE Computer Graphics and Applications, 1991, 11(6): 44-51.
- [6] Dey T K, Zhao W L. Approximating the medial axis from the Voronoi diagram with a convergence guarantee [J]. Algorithmica, 2004, 38(1): 179-200.
- [7] Huang H, Wu S H, Cohen-Or D, *et al.* L_1 -medial skeleton of point cloud [J]. ACM Transactions on Graphics, 2013, 32(4): 65.
- [8] Zhan Q M, Liang Y B, Xiao Y H. Color-based segmentation of point clouds [J]. Laser Scanning, 2009, XXXVIII: 248-252.
- [9] Wang Y H, Hao W, Ning X J, *et al.* Automatic segmentation of urban point clouds based on the Gaussian map [J]. The Photogrammetric Record, 2013, 28(144): 342-361.
- [10] Rabbani T, van den Heuvel F A, Vosselman G. Segmentation of point clouds using smoothness constraints [J]. ISPRS Commission V Symposium:

- Image Engineering and Vision Metrology, 2006, 35: 248-253.
- [11] Wang W Y, Yu R, Huang Q G, *et al.* SGPN: similarity group proposal network for 3D point cloud instance segmentation [C] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 2569-2578.
- [12] Vo A V, Truong-Hong L, Laefer D F, *et al.* Octree-based region growing for point cloud segmentation[J]. ISPRS Journal of Photogrammetry and Remote Sensing, 2015, 104: 88-100.
- [13] Zhang K, Qiao S Q, Zhou W Z. Point cloud segmentation based on three-dimensional shape matching [J]. Laser & Optoelectronics Progress, 2018, 55(12): 121011.
张坤, 乔世权, 周万珍. 基于三维形状匹配的点云分割[J]. 激光与光电子学进展, 2018, 55(12): 121011.
- [14] Ayala D, Brunet P, Juan R, *et al.* Object representation by means of nonminimal division quadrees and octrees [J]. ACM Transactions on Graphics, 1985, 4(1): 41-59.
- [15] Pulli K, Duchamp T, Hoppe H, *et al.* Robust meshes from multiple range maps[C] // Proceedings. International Conference on Recent Advances in 3-D Digital Imaging and Modeling (Cat. No. 97TB100134), May 12-15, 1997, Ottawa, Ont., Canada. New York: IEEE, 1997: 205-211.
- [16] Wang J N, Oliveira M M, Xie H, *et al.* Surface reconstruction using oriented charges [C] // International 2005 Computer Graphics, June 22-24, 2005, Stony Brook, NY, USA. New York: IEEE, 2005: 122-128.
- [17] Deng B W, Wang Z B, Jin Y, *et al.* Feature extraction method of laser scanning point cloud based on morphological gradient [J]. Laser & Optoelectronics Progress, 2018, 55(5): 051203.
邓博文, 王召巴, 金永, 等. 基于形态学梯度的激光扫描点云特征提取方法[J]. 激光与光电子学进展, 2018, 55(5): 051203.
- [18] Li R Z, Liu Y Y, Yang M, *et al.* Three-dimensional point cloud segmentation algorithm based on improved region growing [J]. Laser & Optoelectronics Progress, 2018, 55(5): 051502.
李仁忠, 刘阳阳, 杨曼, 等. 基于改进的区域生长三维点云分割[J]. 激光与光电子学进展, 2018, 55(5): 051502.