

# 基于三维形状匹配的点云分割

张坤\*, 乔世权, 周万珍

河北科技大学信息科学与工程学院, 河北 石家庄 050018

**摘要** 随着三维扫描技术的迅猛发展,点云数据的数据量变得异常庞大,这对点云计算的性能提出了更高的要求。因此,如何有效提高算法的执行效率一直是该领域的研究热点和难点。日益增大的数据量隐藏了丰富的三维(3D)形状模型,将形状模型参与到点云计算过程中,为提高点云计算的执行效率提供了一种新的方法和思路。利用3D几何特征分析技术,获取与形状相关的特征参数,并使其参与到点云分割过程中,提出了形状分割方法。利用八叉树算法组织点云数据,发现数据之间的相邻关系,依靠点云数据的密度自适应地双向线性调整八叉树并建立数据索引。使用规则图形建立3D形状模型库,实现模型与分割区域的匹配,进而提取分割区域的形状参数,为提高点云数据计算的精度和速度奠定基础。在分割效果和分割时间上,对比了不同算法,验证了基于形状的点云分割算法的可行性以及稳健性。

**关键词** 图像处理; 点云数据; 区域分割; 主成分分析法; 随机抽样一致算法; 三维形状匹配

中图分类号 TP242

文献标识码 A

doi: 10.3788/LOP55.121011

## Point Cloud Segmentation Based on Three-Dimensional Shape Matching

Zhang Kun\*, Qiao Shiquan, Zhou Wanzhen

School of Information Science and Engineering, Hebei University of Science and Technology,  
Shijiazhuang, Hebei 050018, China

**Abstract** With the rapid development of three-dimensional (3D) scanning technique, the huge volume of point cloud has been produced, which puts forward higher requirements for the performance of point cloud computing. Therefore, how to improve the efficiency of the algorithm has become a hot topic in this field. There are rich 3D shape models hidden in the ever-increasing amount of point cloud data. Inspired by the relationship between 3D shape models and the point cloud, we provide a new method to improve the execution efficiency of algorithms about point cloud computing. The 3D geometric feature analysis technology is used to obtain shape-related feature parameters, and the point cloud segmentation algorithm is proposed based on it. We use octree algorithm to organize point cloud and obtain the neighbor relationship. A self adaptive and dual linear octree algorithm is designed based on the density of point clouds to establish the data index. We build a 3D shape library by using regular shape models, and realize the algorithm for matching models with data segmentation regions. Further, we extract the shape parameters of the segmented region, which are the foundation for improving the accuracy and speed of point cloud data processing. Moreover, the segmentation effectiveness and time performance of different algorithms are compared, and the experimental results indicate that the proposed algorithm is feasible and robust.

**Key words** image processing; point clouds data; region segmentation; principal component analysis; random sample consensus algorithm; three-dimensional shape matching

**OCIS codes** 100.2960; 100.6890; 110.6880; 100.5010

## 1 引言

三维激光扫描技术的发展拓展了三维图形的应

用领域,使得虚拟现实(VR)、增强现实(AR)技术得到了快速的发展。尤其是便携式三维数据采集设备的出现,例如 Microsoft 公司生产的 Kinect 和

收稿日期: 2018-05-16; 修回日期: 2018-06-16; 录用日期: 2018-07-12

基金项目: 国家自然科学基金(51271033)、河北省自然科学基金(2018208116)、河北省高等学校科学技术研究项目(Z2017008)

\* E-mail: euphkun@163.com

Hololens, 促使了大量、多样的三维(3D)模型的涌现。在这种情况下,3D 计算受到了人们的关注。

点云数据是三维扫描设备提供的最基础的数据形式,具有散乱、无序以及数据量大等特点。以 Kinect V2 为例,每秒可以采集到 12MB 的点云数据。面对海量的、散乱的点云数据,数据分析和数据处理都会消耗大量的系统资源和时间。因此,构建点云的拓扑结构并实现数据分割(将数据划分为独立的区域),是提高 3D 数据重建速度和质量的基础与保证。

点云数据分割算法是点云计算的基础。点云分割算法的实现效果和运行效率,在 3D 对象获取、对象识别以及目标跟踪等应用中起到至关重要的作用。通常,点云分割算法是利用点云特征提取技术获取数据的法向量<sup>[1-2]</sup>、颜色<sup>[3-4]</sup>以及反射光强弱度<sup>[5]</sup>等特征参数,依据特征参数实现分割。然而,随着 3D 数据量的增大,扩充了现有的 3D 数据模型,例如,Trimble 3D Warehouse 就包含了超过 220 万的 3D 模型数据(<https://3dwarehouse.sketchup.com/>)。大量的 3D 数据模型为点云数据分割提供了新的方法和途径。与传统的方法不同,本文从数据模型的角度出发,通过计算点云数据和数据模型的形状特征,实现模型与数据的匹配,从而建立点云分割区域。基于模型的分割可以发现分割区域的几何规则,为数据精简提供帮助,进而加速数据重建和对象识别的速度。主要包含以下几个方面:

1) 创建了基于形状的点云数据分割(SS)方法。该方法以形状相关的特征为驱动,获取数据的结构信息,将点云数据分割成不同的部件,分割后的部件可以使用规则形状进行数学描述。

2) 点云数据的数据量大,为了提高算法的执行效率,改进了八叉树算法,并在此基础上构建了数据索引。首先,利用数据密度值的变化率,沿着两个线性方向向量重组数据。其次,利用重组后的数据参与点云分割,提高了算法的执行效率。此外,重组算法对多目标的场景中不同对象的边界数据敏感,可以有效提取某一对象的边界。

3) 在八叉树数据索引之上,建立了基于法向量参数的一致性分割算法。使用该算法可以提高分割区域内点的相似度,减少形状分割的迭代次数,提高形状分割的效率。此外,该算法可以提高分割区域形状参数估算的准确度。

4) 基于随机抽样一致性(RANSAC)<sup>[6]</sup>算法,定义了形状匹配算法的目标函数,并利用八叉树索引,

迭代式地实现 3D 形状分割,进而提取区域形状参数。

## 2 相关工作

数据区域分割问题属于点云应用系统中的基础研究部分;分割算法的精度和速度起着关键的作用。目前,有关点云区域分割算法有很多,例如 Edge-based 算法、区域增长法、聚类分析、模型匹配方法等。Edge-based 算法是在 1987 年由 Fan 等<sup>[7]</sup>提出,该算法复杂度大,而且边识别的准确度对分割效果影响很大,近几年与之相关的研究较少。

区域增长法,首先选取“种子”数据,通过曲面的平滑度,迭代式地调整区域大小,从而实现点云分割。胡怀宇等<sup>[8]</sup>利用曲率信息分割成七种不同的表面类型,通过区域增长的方法发现每一个种子区域中曲率近似平面的区域。庞世燕等<sup>[9]</sup>通过主成分分析(PCA)提出平面特征,并利用区域增长的方法提取凸壳,结合凸壳算法提取三角面片的轮廓数据。Dimitrov 等<sup>[10]</sup>考虑了数据的多特征实现的分割。Nurunnabi 等<sup>[11]</sup>利用最小协方差行列式(MCD)结合区域增长法实现分割;2015 年提出利用最大一致性与最小距离(MCMD)测量区域数据一致性并实现分割<sup>[12]</sup>。Vo 等<sup>[13]</sup>利用八叉树重采样技术结合区域增长法实现分割。区域增长法的优点是稳定性好,缺点是参数调整不够灵活,分割可控性差。

聚类分析法是多元统计学方法的一种,以“物以类聚”的原则进行划分。点云的聚类分割算法是将点云数据看作具有一定关联属性的多维度数据集,按照多维度属性聚类划分,实现点云数据分割。Velmurugan 等<sup>[14]</sup>利用 K-means 方法完成点云聚类分割。王帅等<sup>[15]</sup>通过谱聚类实现分割。熊风光等<sup>[16]</sup>通过 K-means 算法实现关键点剔除及匹配等。Yan 等<sup>[17]</sup>利用具有噪声的基于密度的聚类方法(DBSCAN)实现点云数据分割。为了优化点云分割算法,相关学者使用多个步骤混合分割数据,比如基于边的分割算法与 RG 方法相结合,通过领域关系以及几何特征建立分割区域等<sup>[18-20]</sup>。基于聚类的点云数据分割算法具有参数设置灵活,算法稳健性好等特点,但收敛过程慢,点云分割算法的时间复杂度大。

模型匹配方法也被称为体分割方法,该方式常用于 man-made<sup>[21]</sup>的数据集。该类算法是利用数学表达式的方式描述分割区域集,例如圆柱、平面等。其中,以 RANSAC 和霍夫变换(HT)算法应用最为广泛。RANSAC 算法最早由 Bolles 等<sup>[6]</sup>于 1981

年提出,目前多应用于3D平面的检测和识别,识别效果较好,但效率不高。例如,文献[22-25]利用RANSAC方法分析点云的平面特征,实现平面的提取和分割。为了获得精准的分割效果,2010年Biegelbauer等<sup>[26]</sup>利用分层的RANSAC算法完成建筑物立体面的提取工作。该研究按照低分辨率到高分辨率的方式,分层完成分割,并通过投票和排名等方法给出最终分割结果。颜色属性作为分割的依据,提高了算法的精准度。例如,Zhang等<sup>[27]</sup>通过将Meanshift算法扩展到3D颜色树上,以聚类算法改进RANSAC迭代执行,使其找到更准确的分割区域。为了降低噪声对分割效果的影响,Kang等<sup>[28]</sup>提出利用Bayesian采样一致性(BaySAC)算法改进RANSAC,文献通过统计候选数据集的直方图表示分割原语,通过分层、并行化的方式完成数据分割。Li等<sup>[29]</sup>通过正态分布变换(NDT)提高了RANSAC算法对平面识别的正确性。

HT方法经常应用于图像中几何形状的识别。2004年,Overby等<sup>[30]</sup>通过3DHough变换实现了地面建筑物的分割工作。2005年,Rabbani等<sup>[31]</sup>又通过3DHough变换识别了众多物体中的圆柱。Tarsha-Kurdi等<sup>[32]</sup>比较了RANSAC和Hough在平面分割方面的性能,指出RANSAC时间消耗相对较低,此外,还通过实验证明了Hough算法对参数的依赖性较大。Liu等<sup>[33]</sup>利用Hough变换分析点云数据中直线的提取技术。Hulik等<sup>[34]</sup>提出利用Hough检测点云的连续平面,并将其和PCL的RANSAC算法对比,结果表明该算法可以得到更小的角度和距离误差。

与3DHough点云分割算法比较,RANSAC算法具有较高的稳健性;然而面对散乱的、不规则的数据时,RANSAC算法时间消耗较大。不幸的是,通过3D扫描仪采集获取的点云数据往往是不完整的、散乱的、并且无任何结构信息。另外,被测对象,尤其是自然对象,其形状是无规则的,很难通过数学的方式去描述与表达。因此,利用RANSAC算法完成形状提取和识别,不但成功率低,而且时间消耗大。为了将RANSAC算法应用到自然对象的点云数据集中,从而使得被测对象中满足或者近似满足规则形状的分割区域被获取,并参与点云的精简和重建过程。本文尝试建立形状模型库,提取点云数据与形状相关的几何特征,重新组织点云数据,并完成模型与数据集的匹配操作。在此基础之上,改进RANSAC算法,提高算法的执行效率以及提高形状

识别的成功率。利用改进的RANSAC算法实现基于形状的数据分割,使得分割区域的形状特征可以用数据的方法描述,从而为点云精简和重建提供理论支撑。

### 3 问题描述和方法概览

#### 3.1 问题描述

点云分割作为点云数据的一个核心计算,其应用范围相当广泛。例如提取被测对象,即在扫描场景将某一个或多个目标对象从背景中提取出来。此外,针对某单一对象实现点云分割也是很有意义的。例如,针对斯坦福大学著名的扫描数据集bunny、dragon等。将其分割成有意义的部件,从而为数据精简、数据重建服务。本文分割算法的目的就是将数据划分为有意义的部件,其部件的特征可以使用数学方式进行描述。

1996年,Hoover等<sup>[35]</sup>提出三维点云分割的形式化描述。点云数据分割的形式化定义中,假设点云数据集用 $R$ 表示,点云分割将会获得 $n$ 个 $R$ 的子集 $R_i$ ,使得 $R_i$ 内的数据集具有相同的或者相似的特征。面对不同的点云分割应用需求,数据特征的描述也不尽相同。

设计基于形状的点云分割,需提取数据集中与形状相关的点云属性,计算与形状相关的数据特征。因此,在文献[35]的基础上,按照形状分割的要求,对基于形状的点云分割进行了形式化描述。假设点云数据为 $R$ ,将数据集进行SS分割成子集 $R_i (i=1,2,\dots,n)$ ,假设 $f(x)$ 为 $R_i$ 数据集的形状特征函数, $g(x)$ 为规则模型的形状特征函数,则 $R_i$ 需满足:

- 1)  $R_i = \{x | x \in R, |f(x) - g(x)| < \sigma\}$ 。
- 2)  $\bigcup_{i=1}^n R_i = R$ ; 数据分割应该考虑的对象是点云数据的全集,而非局部数据。
- 3)  $R_i \cap R_j = \Phi (i \neq j, \wedge j = 1, 2, \dots, n)$ 。
- 4)  $R_i$  内的数据集在空间上是联通的。
- 5)  $R_i$  内的数据集具有相同形状特征。
- 6) 不同的两个子集满足特征属性不相同。

#### 3.2 方法概览

SS根据形状信息,完成散乱点云数据的部件分割。首先通过点云数据的几何特征,提取并计算点云数据的形状特征。其次建立形状模型,并提出了形状模型与点云数据之间的模型匹配算法。最后,在此基础上,实现基于形状的点云数据分割,从而为精简和重建提供规则。如图1所示,SS的具体实施

步骤如下。

1) 数据预处理。首先对数据归一化处理,通过损失函数,自动过滤离群点,从而避免噪声对分割效果的干扰。其次,实现数据配准并建立规范化的、统一的坐标系;提取点云数据的主成分方向,根据主成分方向建立坐标系,并将点云数据投影到此坐标系中,从而实现数据方向的调整。

2) 初始化分割。沿着主成分方向,依据数据之间的近邻关系,建立初始化分割盒子。如图 1 所示,产生若干盒子  $B(p_1, p_2)$ , 盒子以对角线距离最远的点对  $(p_1, p_2)$  为边界。根据盒子的体积和盒子内

点的数量估算点云的密度。

3) 依据  $B(p_1, p_2)$  的密度,按照双方向,线性地调整分割盒  $B(p_1, p_2)$ , 并按照  $B(p_1, p_2)$  的中心,重新组织点云数据,建立有序的数据集。

4) 建立规则图形的 3D 形状库。按照 3D 形状库中的模型,建立与尺度无关的几何特征,包括曲率方差、法向的方向向量的余弦距离等,实现基于几何特征的一致性分割。

5) 建立快速 RANSAC 算法,从而实现基于 3D 形状的分割,并提取匹配参数用于指导点云精简和重建等应用。

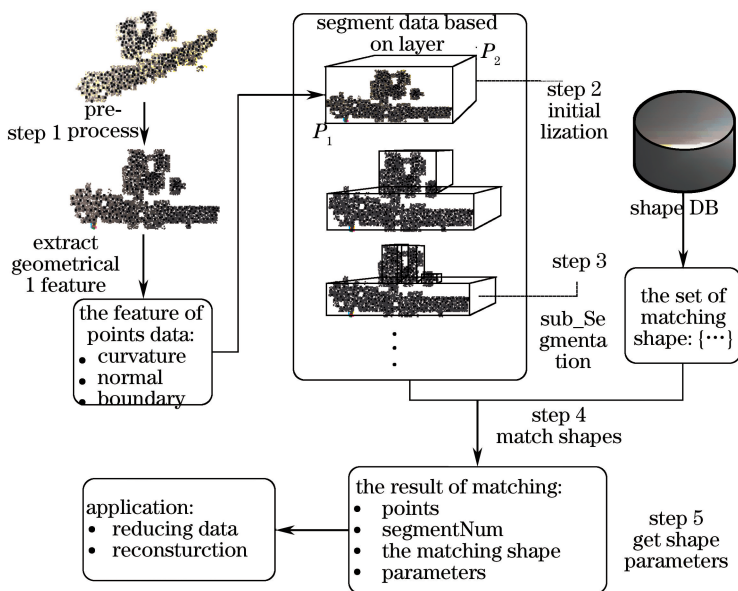


图 1 基于 3D 形状匹配的区域分割方法概览

Fig. 1 Schematic of region segmentation based on 3D shape matching

综上所述,SS 首先根据规则形状,建立了形状的模式库作为分割的依据。其次,利用形状库的模式特征  $g(x)$ , 计算  $g(x)$  与数据集  $R_i$  的几何特征  $f(x)$  之间的距离,并实现与尺度无关的一致性分割,从而满足“ $R_i$  内的数据集具有相同的属性”的要求。

## 4 SS 分割方法

SS 目的是将点云数据按照数据的形状特征相似性进行分割,分割成不同的区域,分割区域的特征可以用规则形状的几何描述来近似表示。本节详细介绍了 SS 关键步骤,包括数据预处理、基于密度的双向线性八叉树数据组织算法以及参数一致性分割和基于形状约束的分割算法。

### 4.1 数据预处理

三维激光扫描设备采集获取的数据是一系列距离值的集合,为了实现采集对象的逆向重建,不得不

将其转换成点云的数据形式,并变换到统一的坐标系中,使其参与点云分割、点云重建等过程。统一坐标系后的数据,由于数据尺度的不同,给数据处理,尤其是参数调整,制造了很大的困难。因此,首先对数据进行归一化处理,并通过数据过滤技术去除噪声的干扰。此外,通常三维激光扫描设备所建立的坐标系是以设置中心为坐标原点,而数据分割算法是在数据对象自身基础上展开的。为了实现高精度的计算,本文通过 PCA 提取数据对象的主成分方向,并利用数据投影技术调整数据方向。

### 4.2 基于密度的双向线性八叉树数据组织方法

点云数据的数据量大,在实现形状特征提取和数据区域分割时,算法执行耗时很大。因此,为了提高算法的执行效率,在已有的八叉树算法基础上,提出了基于密度的双向线性八叉树数据组织方法,并建立数据索引。该方法的目的是发现数

数据集之间的近邻关系,并利用近邻关系建立数据索引,减少SS执行时间。基于密度的双向线性八叉树数据组织方法按照包围盒组织数据集,设定坐标系的原点为整个数据的中心,按照坐标系三个正交方向以及数据中心建立包围盒。图2为基

于密度的双向线性八叉树数据组织方法的示意图,其中种子节点是包含数据中心的节点,在图2中用绿色标出。双向线性八叉树的建立是从种子节点出发,沿着两个线性方向,依据包围盒的密度变化率,进行数据重组。

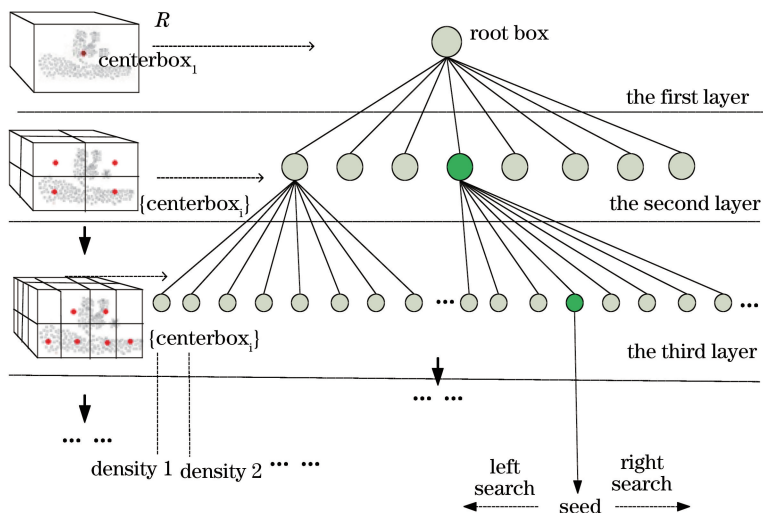


图2 双向线性八叉树数据组织示意图

Fig. 2 Schematic of bidirectional linear octree data organization

假设点云数据集为  $R, n = |R|_1$ 。经过预处理后  $R$  满足:

$$R = \{(x_i, y_i, z_i) \mid |x| < 1, |y| < 1, |z| < 1, i < n\}. \quad (1)$$

针对  $R$  建立根节点的包围盒  $B(l, w, h)$ , 包围盒的中心为坐标原点,  $l, w, h$  分别为包围盒的长、宽、高, 则其应满足:

$$\begin{cases} l = \max_{i=0, \dots, n-1} |x_i| \\ w = \max_{i=0, \dots, n-1} |y_i| \\ h = \max_{i=0, \dots, n-1} |z_i| \end{cases} \quad (2)$$

通过包围盒  $B(l, w, h)$ , 建立八叉树的根节点为  $N_{\text{Rootbox}}$ , 即:

$$N_{\text{Rootbox}} = \{(x, y, z) \mid (x, y, z) \in B(l, w, h) \cap (x, y, z) \in R\}. \quad (3)$$

按照八叉树的规则构建下一层子结点。假设子节点的中心坐标表示为  $N_{\text{Centerbox}_i}$ , 子节点相对于父节点的偏移距离设为  $d$ , 那么  $d = (d_x, d_y, d_z)$ , 并且:

$$(d_x d_y d_z)_B = [(i \bmod 8) - 1]_B, \quad (4)$$

式中  $B$  表示二进制数。

中心坐标  $N_{\text{Centerbox}_i}$  为

$$N_{\text{Centerbox}_i} = N_{\text{Centerbox}_{\lfloor (i-1)/8 \rfloor}} - \vec{d} \times \left( \frac{l}{2^H}, \frac{w}{2^H}, \frac{h}{2^H} \right). \quad (5)$$

子节点  $N_{\text{Nextbox}_i}$  即为  $R$  的一个分割子集  $R_i$ , 即:

$$N_{\text{Nextbox}_i} = \left\{ (x, y, z) \mid (x, y, z) \in B_i \left( \frac{l}{2^H}, \frac{w}{2^H}, \frac{h}{2^H} \right) \right\}, \quad i = 0, 1, \dots, 7, \quad (6)$$

式中  $H$  为构建八叉树的深度。

利用八叉树中包围盒的相邻关系, 通过双向线性方式获取初始化分割数据集。分为以下几个步骤来完成。

1) 按照(7)式估算每一个包围盒的数据密度:

$$P(R_i) = \frac{\|R_i\|_1}{f(H)}, \quad (7)$$

式中  $\|R_i\|_1$  为集合元素的个数。

2) 根据包围盒的数据密度  $P(R_i)$ , 删除冗余包围盒。其中冗余包围盒即为满足  $P(R_i) = 0$  的包围盒。

3) 通过双向线性方式, 合并相邻且密度相近的包围盒。

4) 找到密度值小或区域面积变化率大的区域作为初始化分割的边界盒子。

5) 生成若干初始化分割盒子, 将盒子的中心点作为数据的索引。

图3是进行基于密度的双向线性八叉树组织后, 边界数据的效果图。其中边界数据在图中用黄色标记标出。



图3 边界盒与原图对应位置的对比

Fig. 3 Comparison of the position of the boundary box and the original image

### 4.3 SS 点云分割算法

通过上述步骤,点云数据可以实现噪声过滤、建立投影以及数据索引等预处理。预处理后的点云数据减少了噪声的干扰,并具有相对一致的方向向量。除此之外,利用 4.2 节建立的数据索引,能够获取数据之间空间距离的相近度,为 SS 的实施奠定了基础。SS 的目的是通过规则形状模型去近似地描述分割区域。因此,SS 要求能够用数学方法来表达规则模型,并且这些规则模型能够描述足够全面的分割区域。然而,自然对象是多样的,满足上述要求的规则模型很难找到。

Várady 等<sup>[36]</sup>根据多数计算机辅助设计/计算机辅助工程(CAD/CAM)系统中所提供的曲面设计手段,将曲面分为主要曲面和次要曲面两类。主要曲面决定了对象的主体部分以及基本形状特征,主要曲面包括简单曲面和自由曲面两部分。最常出现的简单曲面为二次曲面;其中,球面、圆柱面和圆锥面被称为自然二次曲面。在实际工程应用中,产品的表面通常是由平面和自然二次曲面构成的,一般二次曲面很少出现。

为了实现满足上述要求的规则模型,主要以自然二次曲面作为点云分割的形状库。SS 与其他分割算法最大的不同在于,SS 目的是建立分割区域与规则模型之间的对应关系,从而利用规模模型指导点云精简和重建等。因此,SS 依赖于形状特征的检测和匹配。现实中,对象表面形状变化多样,极少有规则形状的对象模型。在进行规则形状检测和匹配之前,SS 先提取形状特征的关键点,并对对象模型的连续性和参数一致性进行检查和一致性分割,之后利用 RANSAC 算法确定分割区域,并提取分割区域的形状参数。

#### 4.3.1 模型的一致性分割

与曲面不同,点云数据是散乱的、离散的。点云数据不适合用曲面、曲线的光滑度参数( $G^0 \sim G^4$ )来表示。由 4.2 节得到基于密度的八叉树方法重组后的点云数据,数据集按照空间相邻关系,以  $B(l, w, h)$  为单位进行重新组织。然而,将  $B(l, w, h)$  内部的数据子集实施重建,重建后的曲面不一定是连续

和光滑的。这给该区域的形状特征提取和分析造成了一定的困难。例如,图 4(a)中  $\{points1\}$  和  $\{points2\}$ ,按照基于密度的八叉树方法进行重组。重组后的按彩图展示如图 4(b)所示。图 4(b)黑色圈内是包含  $\{points1, points2\}$  的盒子(盒子中的数据用绿色标识)。如图 4(b)所示,  $\{points1\}$  和  $\{points2\}$  被包含在相同的盒子中。图 4(a)的数据被重建之后结果如图 4(c)所示。图 4(c)中  $\{points1\}$  和  $\{points2\}$  对应于重建区域  $area1$  和  $area2$ ,这两个区域是连续的,两区域的切平面并不连续。因此,其形成曲面后,曲面参数并不一致,而这个不一致性利用 4.2 节的方法也无法辨识出来。

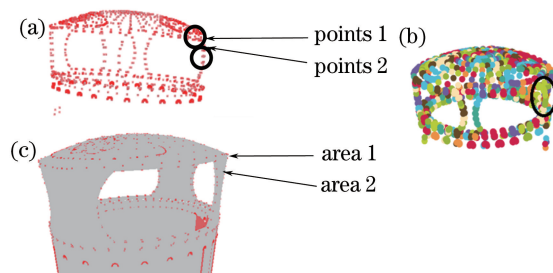


图4 点云参数不一致示例图。(a)点云数据;(b)重组后的数据;(c)曲面

Fig. 4 Diagram of inconsistent point cloud parameters.

(a) Point clouds data; (b)reorganized data; (c) curved surface

为了满足准确的基于形状的区域分割,本节计算点云数据特征点,按照特征点和主成分方向建立一致性分割线,并对数据集进行一致性分割。在上述基于密度约束的八叉树数据组织基础之上,依据相邻数据集的法向量夹角,检测分割后模型的连续性和一致性,如图 5 所示。通过一致性检测后,找到区域分割的关键点,最终实现一致性分割。

1) 一致性分割算法随机选取  $n$  个不同的  $\{B_{Box}\}$  作为  $B_{Boxbase}$ ,并通过  $B_{Boxi}$  的  $N_{centerboxi}(x, y, z)$  计算距离值:

$$D_{distance} = \| B_{Boxi} - B_{Boxbase} \|_2 = \| N_{centerboxi} - N_{centerboxbase} \|_2. \quad (8)$$

2) 设定扇形的角度  $\alpha, \alpha \in (0, 2\pi]$ 。按照主方向,利用分割关键点,建立以  $B_{Boxbase}$  为中心的扇形区域,并对扇形区域内的数据投票计数。

3) 算法通过  $n$  次投票的统计结果来确定区域的一致性分割,分割结果存储为  $R = \{R_i\}$

$$R_i = \{p_i(x, y, z) \mid \text{roll}(p_i) = \text{roll}(p_j)\}, \quad (p_j \in R_i, i \neq j), \quad (9)$$

式中  $\text{roll}(p_i)$  表示  $p_i$  的投票数。

图 4 中点云数据按照一致性分割后效果如图 6

所示。图 6 中  $R_i$  数据集使用 9 种不同的颜色循环展示效果图。如图 6 黑色椭圆标识出的区域,一致性分割后,可以识别出位置相邻而曲面参数不连续的数据集。

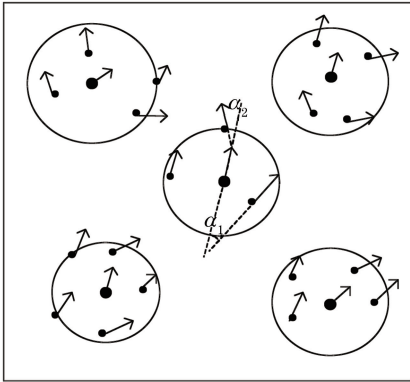


图 5 法式夹角判断连续性

Fig. 5 Inconsistent estimated by angle of normal

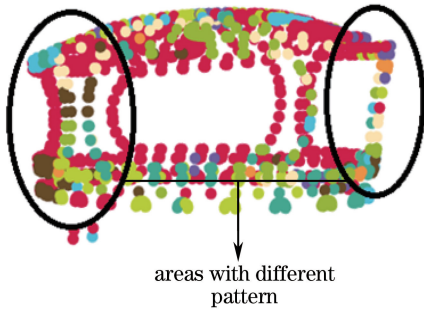


图 6 一致性分割后效果图

Fig. 6 Effect of consistence segmentation

#### 4.3.2 基于 RANSAC 的形状分割算法

SS 算法在一致性分割后,可以获取的分割模型用  $R_i$  表示。受 RANSAC 算法的启发,建立规则形状的数据库,数据库中的形状  $M$  可以利用  $k$  个变换参数  $\theta_k$  来表示,  $M(\theta_k)$ 。SS 算法为了找到  $R_i$  与  $M(\theta_k)$  的相似性,提取了形状特征,并根据形状特征建立了目标函数  $C$ 。

1)  $M_{\text{plane}}$  的目标函数  $C$ 。建立模型  $M$  的曲率特征。图 7 为标准二次曲面的曲率特征直方图。 $M_{\text{plane}}$  代表标准的平面模型,即:

$$M_{\text{plane}}(\theta_k) = \theta_0 x + \theta_2 y + \theta_3 z + \theta_4. \quad (10)$$

$M_{\text{plane}}$  与图 7 中的其他规则模型的曲率特征存在显著的差异,  $M_{\text{plane}}$  的曲率绝对值接近于 0。因此,通过曲率特征可以辨识出。SS 算法通过 PCA 获得三个主成分方向及其对应的特征值  $\lambda_0, \lambda_1, \lambda_2$ , 其中最小值为  $\lambda_0$ 。分割子集  $R_i$  的曲率值  $c(R_i)$  可以表示为

$$c(R_i) = E \left( \left\| \frac{p^{\lambda_0}}{p^{\lambda_0} + p^{\lambda_1} + p^{\lambda_2}} \right\| \right), \quad (11)$$

式中  $p$  为  $\{p_i(x, y, z) \mid p_i(x, y, z) \in R_i\}$ ,  $E$  为  $R_i$  中各点曲率的均值函数。

$R_{\text{cur}}$  为进行曲率值过滤后的数据集,即:

$$R_{\text{cur}} = \{R_i \mid |c(R_i)| < \epsilon\}. \quad (12)$$

目标函数  $C$  为  $|R_{\text{cur}}|_1$ ,  $\epsilon$  为曲率阈值。

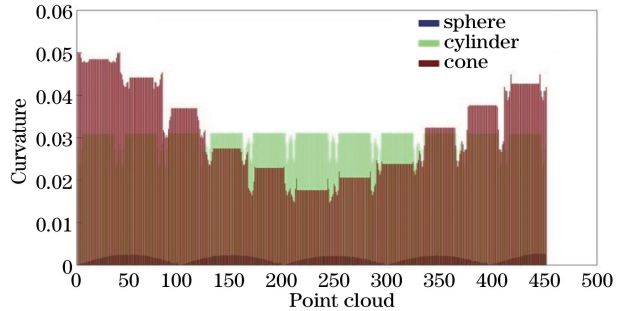


图 7 规则形状的曲率特征直方图

Fig. 7 Histogram of curvature of regular shape

2) 除  $M_{\text{plane}}$  外,其他  $M$  模型的目标函数为  $C$ 。其他  $M$  模型的生成和匹配是通过改进的 RANSAC 算法实现的。RANSAC 算法的正确执行在一定程度上依赖于采样点的质量。因此为了提高采样点的质量,避免采样点集中在某一个分割区域,从而造成分割结果陷入局部最优。本文选择  $\{N_{\text{centerbox}}\}$  中的  $m$  个最小采样点,  $m$  的大小刚好满足计算  $\theta$  的个数即可。通过采样点估算模型获取  $M(\theta_k)$ , 并建立基于形状分割子集  $R_{\text{conicoid}}$ :

$$R_{\text{conicoid}} = \{R_i \mid \|p - M(\theta_k)\| < \zeta\}, \quad (13)$$

式中  $p$  表示为  $\{p_i(x, y, z) \mid p_i(x, y, z) \in R_i\}$ ,  $\zeta$  为用户定义的阈值。

目标函数  $C$  为  $|R_{\text{conicoid}}|_1$ 。

## 5 SS 算法实验及实验分析

### 5.1 实验环境及数据集介绍

实验环境: CPU, Intel (R) Core (TM) i5-5200U, 频率为 2.3 GHZ, 系统内存 8.0 G。软件平台采用 QT5.6.2 + PCL1.8 + VS2015。数据集使用 MySQL 管理。实验选择三种数据集,分别是单对象、多对象以及模型数据,具体数据信息如表 1 所示。

### 5.2 基于形状分割实验的执行

#### 5.2.1 基于密度的双向线性八叉树数据组织算法的实验

本实验验证了基于密度的双向线性八叉树数据组织算法的可行性。本实验利用数据集 Buuny 和 Dragon 建立数据组织  $\{B_{\text{Box}i}\}$  以及  $\{N_{\text{centerbox}i}\}$ , 并按

表 1 实验数据集介绍  
Table 1 Experiment data set

Data type	Data set	Size	Producter	Device	Download	
Natural data set	Single_ Object	Bunny	35947	Stanford University	Cyberware 3030 MS scanner	<a href="http://graphics.stanford.edu/data/3Dscanrep/">http://graphics.stanford.edu/data/3Dscanrep/</a>
		Coal	8568	Hebei university of science and technology	SICK LMS 100	
	Multi_ Object	Kitchen	350706	Washington University	Kinect	<a href="http://rgbd-dataset.cs.washington.edu/dataset.html">http://rgbd-dataset.cs.washington.edu/dataset.html</a>
Man-made data set	Model	Model_normalized_9	3844	Stanford University	Sketch	<a href="https://www.shapenet.org/download">https://www.shapenet.org/download</a>

照  $\{N_{centerboxi}\}$  建立数据索引。其中图 8(a) 是 Bunny 数据集的重组结果；图中按照不同的盒子重建组织数据集的结果展示。图 8(b) 是 Bunny 数据的  $\{N_{centerboxi}\}$  展示。图 8(c)、(d) 是 Dragon 数据的盒子以及  $\{N_{centerboxi}\}$  结果展示。

本实验验证了基于密度的双向线性八叉树点云数

据组织算法的执行效果。本实验利用八叉树算法建立数据索引如图 9(a) 所示，八叉树算法产生 505 个体素 (voxel)。计算每个 voxel 的数据密度，并建立密度直方图如图 9(a) 所示。利用基于密度的双向线性八叉树点云数据组织算法，算法执行后生成  $\{B_{Boxi}\}$  集合，集合大小为 77，并建立密度直方图如图 9(b) 所示。

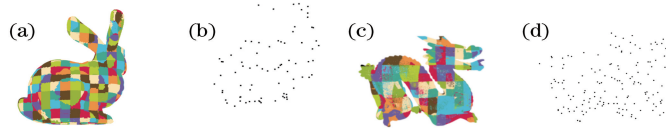


图 8 基于密度的双向线性八叉树数据组织实验效果展示。(a) Bunny =  $\{B_{Boxi}\}$ ; (b) Bunny:  $\{N_{centerboxi}\}$ ; (c) Dragon =  $\{B_{Boxi}\}$ ; (d) Dragon:  $\{N_{centerboxi}\}$

Fig. 8 Effect of bidirectional linear octree data organization. (a) Bunny =  $\{B_{Boxi}\}$ ; (b) Bunny:  $\{N_{centerboxi}\}$ ; (c) Dragon =  $\{B_{Boxi}\}$ ; (d) Dragon:  $\{N_{centerboxi}\}$

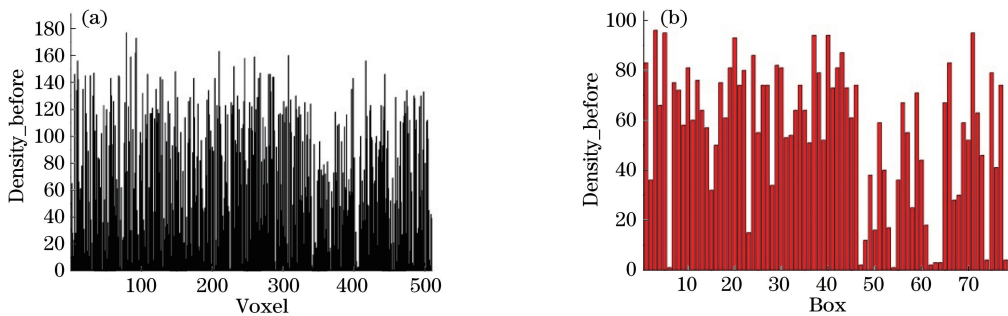


图 9 数据预处理前、后密度对比图。(a) Voxel; (b) box

Fig. 9 Comparison of density before and after data preprocessing. (a) Voxel; (b) box

图 9 是数据预处理前、后的密度直方图。图 9(b) 是对数据进行基于密度的双向八叉树算法后的数据密度直方图。由图 9 可以看出，图 9(b) 将密度相近的邻域数据进行重新组织。重组后的数据并没有改变数据的密度分布变化，却减少了索引数，从 510 个 voxel 减少到 77 个盒子。

针对多场景数据，本实验使用基于密度的双

向线性八叉树点云数据组织算法，发现满足边界条件的盒子集合采用红色区域标识。效果如图 10 所示。

采用基于密度的双向八叉树算法应用于多对象的点云数据应用场景，数据密度参数对不相同的数据对象较为敏感。图 10 所示为应用基于密度的双向八叉树算法，并按盒子的区域面积以及密度值从



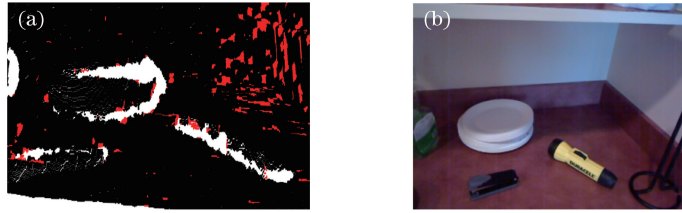


图 10 多目标场景中不同对象的分界盒子。(a)不同对象的分界盒子;(b)原始图片

Fig. 10 Boundary box for different objects in a multi-object scenario. (a) Boundary box for different objects; (b) original image

小到大进行排序,取前 500 的盒子,如图 10(a)所示。从图 10 可以看出,这些盒子包含了托盘、订书器和手电筒的边界。此外,在图 10(a)右上方,对应于图 10(b)的墙拐角和墙面的踢脚线处,也存在对数据密度敏感的数据集。

## 5.2.2 SS 与其他分割算法对比实验

### 1) 分割效果对比实验

本实验使用区域增长法、K-means 算法、RANSAC 算法和 SS 算法在 Bunny 和 Coal 数据集上进行分割,分割效果如图 11 和 12 所示。

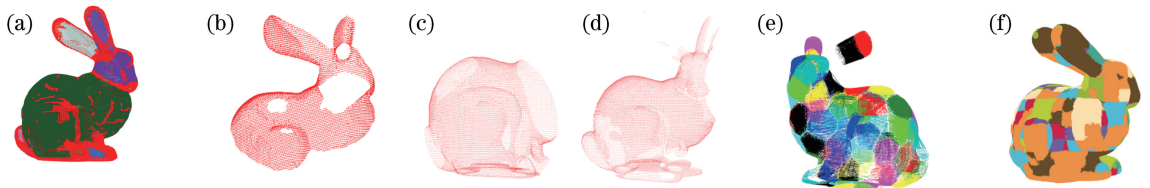


图 11 Bunny 数据集分割效果图。(a)区域增长法;(b) RANSAC(平面);(c) RANSAC(球体);(d) RANSAC(圆柱体);(e) K-means;(f) SS

Fig. 11 Results of segmentation algorithms using Bunny data set. (a) Region growing; (b) RANSAC (plane); (c) RANSAC (sphere); (d) RANSAC (cylinder); (e) K-means; (f) SS

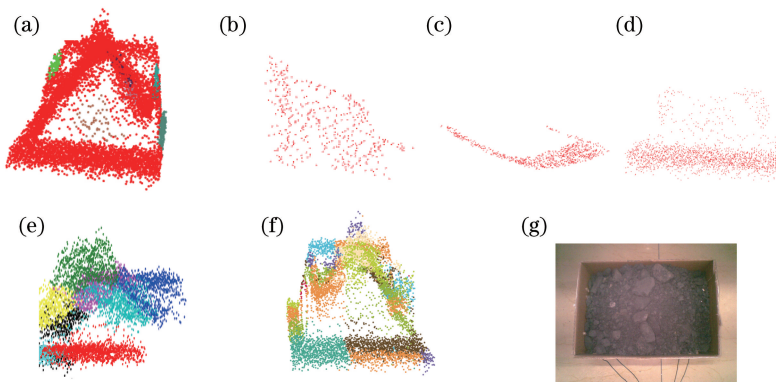


图 12 Coal 数据集分割效果图。(a)区域增长法;(b) RANSAC(平面);(c) RANSAC(球体);(d) RANSAC(圆柱体);(e) K-means;(f) SS;(g)被测对象

Fig. 12 Results of segmentation algorithms using Coal data set. (a) Region growing; (b) RANSAC (plane); (c) RANSAC (sphere); (d) RANSAC (cylinder); (e) K-means; (f) SS; (g) measured object

图 11、12 的(a)是基于区域增长法的分割效果图。可以看出,算法执行后,曲面的边界点和特征点被划分为一个个单独的分割区域。例如,图 11(a)中兔子耳朵及其边缘,就被分割为两个不同的区域。因此区域增长法算法不能保证分割区域在形状上是完整的。

图 11、12 的(b)~(d)是应用 RANSAC 分割算

法执行后的效果图。同时,本实验还对 RANSAC (圆锥)进行了测试,RANSAC (圆锥)算法执行后没有生成满足要求的分割区域,因此算法执行失败。失败的原因与实验数据集自身形状特征有关,由于数据集自身不存在满足或者近似满足“圆锥”形状的数据区域。通过实验表明,使用 RANSAC 只能针对某一个 3D 形状进行分析和计算,不能同时获取

符合多个 3D 形状的分割区域。因此,同时处理多规则形状的计算时,该算法不适用。图 11(e)是  $K$ -means 算法的分割效果图,该算法不能实现基于形状特征的分割。图 11(f)是 SS 分割算法实施后的效果图。可以看出,SS 算法分割后的区域包含了形状特征点,因此保障了分割区域在 3D 形状上的完整性。和 RANSAC 算法相比,SS 算法可以对同一对象,同时展开多个形状区域的分割。

## 2) 分割区域曲面曲率方差对比实验

为了量化基于形状的区域分割算法的执行效果,本实验分别利用  $K$ -means 算法、区域增长法、八叉树法和 SS 算法进行分割,并统计了分割区域内曲面的曲率方差。如图 13 所示。其中区域增长法受算法的参数精度限制,分割区域数量范围为 10~50,分割数据集的曲率方差范围为(0.003, 0.07)。 $K$ -means 分割后区域设定为 70,曲率方差范围为(0.003, 0.07),而八叉树算法分割后数据的曲率方差的变化平稳,曲率方差范围为(0.005, 0.07)。SS 算法曲面的曲率方差范围为(0.0002, 0.06)。通过对比,SS 算法分割后的数据集曲面变化平稳,曲面参数一致性优于其他算法。

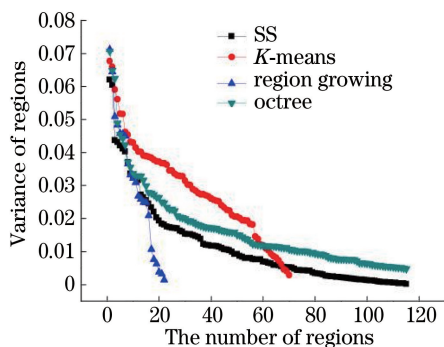


图 13 分割曲面一致性的对比图

Fig. 13 Contrast of consistency among the regions

## 3) 分割区域形状相似度对比实验

实验验证了 SS 算法分割区域内数据集的形状相似性。实验采用 SS 算法,  $K$ -means 算法以及区域增长算法对 Bunny 数据集进行数据分割。实验设定分割区域中满足目标形状的数据集的大小与分割数据集的大小比值作为形状的不确定值。形状不确定值范围为[0, 1],形状不确定值表明了分割区域与目标形状的相似程度。不确定值越大,说明目标形状与分割区域的形状越接近。从图 14 可以看出,SS 分割算法,形状相似度为 1 的分割区域在数量上多于其他算法,分割效果优于其他分割算法。

## 4) 算法运行时间对比实验

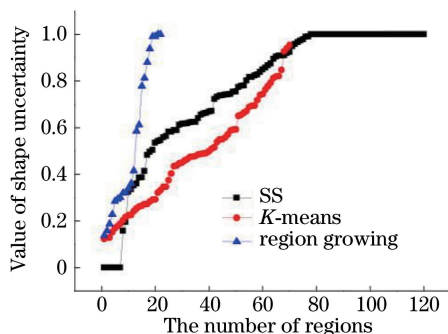


图 14 分割曲面形状不确定性对比

Fig. 14 Contrast of shape uncertainty among the regions

对不同算法的运行时间进行了对比,如图 15 所示。可以看出,区域增长法的运行时间最短,该分割算法的灵活性差,分割区域数量范围为 10~50。 $K$ -means 算法可以分割成不同大小的数据集,算法灵活性好,但运行时间长。与  $K$ -means 算法相比,本文算法提高了分割算法灵活性的同时,减少了算法的执行时间。

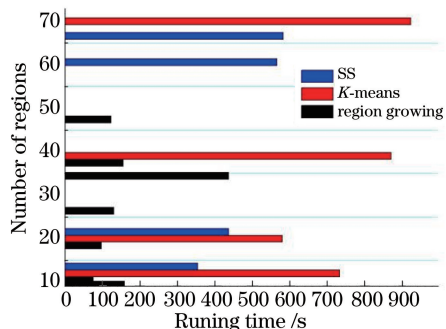


图 15 分割算法执行时间对比

Fig. 15 Contrast of algorithm running time

## 6 结 论

从三维形状分析的角度出发,对点云对象实施分割,使得分割后的数据包含 3D 形状信息,而该形状信息可用于指导点云数据的精简和重建,从而提高点云数据的计算速度和准确度。为了达到点云分割的要求,需定义 3D 形状特征,并在此形状特征基础上实现点云数据分割。由于点云数据的散乱性,通过数据之间的近邻关系和数据密度,提出了基于密度的双向线性八叉树的点云组织算法,利用线性八叉树的结构重新组织点云数据;重组后的数据集使用  $\{B_{Boxi}\}$ ,  $\{N_{centerboxi}\}$  建立数据索引。实验证明该算法有效降低了数据的散乱性,使得  $\{B_{Boxi}\}$  内部数据分布均匀、一致;尤其是当算法在应用于多对象的场景中时,算法对不同对象之间的分割区域较为敏感,对不同的对象有较好的区分度。其次,为了

使分割后的数据在形状上相似,对重组后的点云数据展开一致性分割,使得分割区域内数据的曲面特征相似或相近,能够识别出曲面变化率较大的交界区域。最后,建立规则点云库,在一致性分割的基础上根据点云库中的3D形状,进行基于形状的数据分割,并提取出与规则库中的形状相似的分割区域及其规则形状参数。

提出的基于形状的区域分割算法可以有效提取出被测对象的形状特征。分割区域内的数据具有一定的形状相似度。因此,在对点云精简和压缩过程中,可以通过形状模型的特征,有针对性地识别出精简部分。同时,形状参数可以作为先验知识,提高曲面重建的效率。

本文算法存在一定的局限性。建立的规则形状库有限,分割区域与形状的映射关系少。此外算法的分割是以点云的位置关系作为依据而实施的,并未涉及到颜色等其他信息。今后的工作:1)扩展3D规则形状模型,使得形状更为丰富,与真实的被测对象更接近,提高分割的精准度;2)为了减少算法的执行时间,将在分割算法的并行化方面进行优化;3)基于形状的3D点云分割算法在点云精简和重建等应用上进行更多的探索和尝试。

### 参 考 文 献

- [1] Che E, Olsen M J. Multi-scan segmentation of terrestrial laser scanning data based on normal variation analysis [J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2018, 143: 233-248.
- [2] Zhao Z Z, Zhang Y J. Building extraction from airborne laser point cloud using NDVI constrained watershed algorithm[J]. *Acta Optica Sinica*, 2016, 36(10): 1028002.  
赵宗泽,张永军. 基于植被指数限制分水岭算法的机载激光点云建筑物提取[J]. *光学学报*, 2016, 36(10): 1028002.
- [3] Hołowko E, Wojsz J, Sitnik R, *et al.* Color-based algorithm for automatic merging of multiview 3D point clouds[J]. *ACM Journal on Computing and Cultural Heritage*, 2014, 7(3): 16.
- [4] Li C L, Zhong F, Zhang Q, *et al.* Accurate and fast 3D head pose estimation with noisy RGBD images [J]. *Multimedia Tools and Applications*, 2018, 77(12): 14605-14624.
- [5] Miyake S, Toda Y, Kubota N, *et al.* Intensity histogram based segmentation of 3D point cloud using

growing neural gas[C]// *International Conference on Intelligent Robotics and Applications*, Springer International Publishing, 2016: 335-345.

- [6] Bolles R C, Fischler M A. ARANSAC-based approach to model fitting and its application to finding cylinders in range data [C]// *International Joint Conference on Artificial Intelligence*, 1981: 637-643.
- [7] Fan T J, Medioni G, Nevatia R. Segmented descriptions of 3-D surfaces [J]. *IEEE Journal on Robotics and Automation*, 1987, 3(6): 527-538.
- [8] Hu H Y, Cui H G, Dai X. Segmentation of scattered point data based on region growing method [J]. *Journal of Computer Applications*, 2009, 29(10): 2716-2718.  
胡怀宇,崔汉国,代星. 基于区域生长法的散乱点云分区方法[J]. *计算机应用*, 2009, 29(10): 2716-2718.
- [9] Pang S Y, Liu Y W, Zuo Z Q, *et al.* Combination of region growing and TIN edge segmentation for extraction of geometric features on building facades [J]. *Geomatics and Information Science of Wuhan University*, 2015, 40(1): 102-106.  
庞世燕,刘亚文,左志奇,等. 结合区域增长法和TIN边缘分割的建筑物立面几何特征提取[J]. *武汉大学学报(信息科学版)*, 2015, 40(1): 102-106.
- [10] Dimitrov A, Golparvar-Fard M. Segmentation of building point cloud models including detailed architectural/structural features and MEP systems [J]. *Automation in Construction*, 2015, 51: 32-45.
- [11] Nurunnabi A, Belton D, West G. Robust segmentation in laser scanning 3D point cloud data [C]// *International Conference on Digital Image Computing Techniques and Applications (DICTA)*, 2012: 1-8.
- [12] Nurunnabi A, West G, Belton D. Outlier detection and robust normal-curvature estimation in mobile laser scanning 3D point cloud data [J]. *Pattern Recognition*, 2015, 48(4): 1404-1419.
- [13] 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.
- [14] Velmurugan T, Santhanam T. Computational complexity between  $K$ -means and  $K$ -medoids clustering algorithms for normal and uniform distributions of data points[J]. *Journal of Computer Science*, 2010, 6(3): 363-368.

- [15] Wang S, Sun H Y, Guo H C, *et al.* Mixed manifold spectral clustering adaptive segmentation method for laser point cloud [J]. *Acta Optica Sinica*, 2017, 37(10): 1011001.  
王帅, 孙华燕, 郭惠超, 等. 激光点云的混合流形谱聚类自适应分割方法 [J]. *光学学报*, 2017, 37(10): 1011001.
- [16] Xiong F G, Huo W, Han X, *et al.* Removal method of mismatching keypoints in 3D point cloud [J]. *Acta Optica Sinica*, 2018, 38(2): 0210003.  
熊风光, 霍旺, 韩燮, 等. 三维点云中关键点误匹配剔除方法 [J]. *光学学报*, 2018, 38(2): 0210003.
- [17] Yan J Z, Qi M Y, Fang L Y, *et al.* Forecast the distribution of urban water point by using improved DBSCAN algorithm [C] // *International Conference on Intelligent System Design and Engineering Applications*, 2013: 784-786.
- [18] Zhang X, Zang A, Agam G, *et al.* Learning from synthetic models for roof style classification in point clouds [C] // *ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2014: 263-270.
- [19] Mahmoudabadi H, Olsen M J, Todorovic S. Efficient terrestrial laser scan segmentation exploiting data structure [J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2016, 119: 135-150.
- [20] Holz D, Behnke S. Approximate triangulation and region growing for efficient segmentation and smoothing of range images [J]. *Robotics and Autonomous Systems*, 2014, 62(9): 1282-1293.
- [21] Boulch A, Guerry J, le Saux B, *et al.* SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks [J]. *Computers & Graphics*, 2018, 71: 189-198.
- [22] Schnabel R, Wahl R, Klein R. Efficient RANSAC for point-cloud shape detection [J]. *Computer Graphics Forum*, 2007, 26(2): 214-226.
- [23] Papazov C, Burschka D. An efficient RANSAC for 3D object recognition in noisy and occluded scenes [C] // *Asian Conference on Computer Vision*, 2010, 6492(1): 135-148.
- [24] Zuliani M, Kenney C S, Manjunath B S. The multi RANSAC algorithm and its application to detect planar homographies [C] // *IEEE International Conference on Image Processing*, 2005, 3: III-153.
- [25] Tran T T, Cao V T, Laurendeau D. Extraction of cylinders and estimation of their parameters from point clouds [J]. *Computers & Graphics*, 2015, 46: 345-357.
- [26] Biegelbauer G, Vincze M, Wohlkinger W. Model-based 3D object detection [J]. *Machine Vision and Applications*, 2010, 21(4): 497-516.
- [27] Zhang X M, Wan W G, Xiao L, *et al.* Mean shift clustering segmentation and RANSAC simplification of color point cloud [C] // *International Conference on Audio, Language and Image Processing*, 2014: 837-841.
- [28] Kang Z Z, Li Z. Primitive fitting based on the efficient multiBaySAC algorithm [J]. *PLoS ONE*, 2015, 10(3): e0117341.
- [29] Li L, Yang F, Zhu H H, *et al.* An improved RANSAC for 3D point cloud plane segmentation based on normal distribution transformation cells [J]. *Remote Sensing*, 2017, 9(5): 433.
- [30] Overby J, Bodum L, Kjems E, *et al.* Automatic 3D building reconstruction from airborne laser scanning and cadastral data using Hough transform [J]. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2004, 34 (Part B3): 296-301.
- [31] Rabbani T, von den Heuvel F. Efficient Hough transform for automatic detection of cylinders in point clouds [J]. *ISPRS Workshop on Laser Scanning*, 2005, 3: 60-65.
- [32] Tarsha-Kurdi F, Landes F, Grussenmeyer P. Extended RANSAC algorithm for automatic detection of building roof planes from LiDAR data [J]. *The Photogrammetric Journal of Finland*, 2008, 21(1): 97-109.
- [33] Liu Y E, Li Z R, Hayward R, *et al.* Classification of airborne LIDAR intensity data using statistical analysis and Hough transform with application to power line corridors [C] // *Digital Image Computing: Techniques and Applications*, 2009: 462-467.
- [34] Hulik R, Spanel M, Smrz P, *et al.* Continuous plane detection in point-cloud data based on 3D Hough transform [J]. *Journal of Visual Communication and Image Representation*, 2014, 25(1): 86-97.
- [35] Hoover A, Jean-Baptiste G, Jiang X, *et al.* An experimental comparison of range image segmentation algorithms [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1996, 18(7): 673-689.
- [36] Várady T, Martin R R, Cox J. Reverse engineering of geometric models: an introduction [J]. *Computer-Aided Design*, 1997, 29(4): 255-268.