

# 基于坡度阈值与卷积滤波处理的激光雷达地面分割算法

上官韬<sup>1</sup>, 谢荣<sup>1\*</sup>, 雷祖芳<sup>2</sup>, 刘峥<sup>1</sup>

<sup>1</sup>西安电子科技大学雷达信号处理全国重点实验室, 陕西 西安 710071;

<sup>2</sup>深圳市镭神智能系统有限公司, 广东 深圳 518100

**摘要** 针对传统激光雷达地面分割算法在处理复杂坡度路面时存在实时性差和阈值依赖等问题, 提出一种基于粗-细两阶段处理的地面分割算法。首先将点云划分成扇形区域, 自适应确定每个区域的局部坡度阈值, 以完成第一阶段粗分割; 然后将粗分割后的点云投影到 RGB 图像上, 使用图像膨胀算法获得“未知分类点”; 最后利用划分多区域确定距离阈值的思想, 筛选出有效点对“未知分类点”, 执行卷积滤波实现细分割。实测数据处理结果表明, 所提算法在平坦路面和复杂坡度路面场景下的地面点分割精确率均在 96% 以上, 召回率也维持较高水平, 均在 95% 以上, 且局部坡度阈值可以自适应调整, 具有较好的鲁棒性, 平均耗时为 16.57 ms, 满足无人车辆实时性的要求。

**关键词** 激光雷达; 地面分割; 坡度阈值; 多区域; 卷积

中图分类号 TP391.4

文献标志码 A

DOI: 10.3788/LOP230491

## LiDAR Ground-Segmentation Algorithm Based on Slope Threshold and Convolution Filtering Processing

Shangguan Tao<sup>1</sup>, Xie Rong<sup>1\*</sup>, Lei Zufang<sup>2</sup>, Liu Zheng<sup>1</sup>

<sup>1</sup>National Key Laboratory of Radar Signal Processing, Xidian University, Xi'an 710071, Shaanxi, China;

<sup>2</sup>Shenzhen Leishen Intelligent System Co., Ltd., Shenzhen 518100, Guangdong, China

**Abstract** In this study, we propose a ground-segmentation algorithm based on a two-stage coarse-fine processing approach to address the limitations of traditional LiDAR ground-segmentation algorithms, such as poor real-time performance and threshold dependence for complex slope roads. First, the point cloud was divided into fan-shaped regions, with the local slope threshold of each area being adaptively determined to complete the first stage of rough segmentation. Subsequently, the point cloud was projected onto the RGB image, and “unknown classification points” were obtained using an image expansion algorithm. Finally, effective points were screened out to perform convolution filtering on the “unknown classification points” to achieve fine segmentation and to determine the distance threshold by dividing multiple regions. The results demonstrate that the proposed algorithm achieves a ground-point segmentation accuracy exceeding 96% for both flat roads and complex slope roads. In addition, the recall rates consistently maintain a high level of over 95%. Moreover, the local slope threshold can be adjusted to achieve excellent robustness, and the average processing time is 16.57 ms, which satisfies the real-time requirements of unmanned vehicles.

**Key words** LiDAR; ground segmentation; slope threshold; multi-region; convolution

## 1 引言

激光雷达因数据量大、测量精度高、探测范围广、环境适应性强等特点已成为目前自动驾驶环境感知系统的主传感器。现有激光雷达按激光线数可分为单

线、4线、16线、32线、64线及128线雷达, 线数越高, 每帧数据量越大, 价格也越昂贵。本文采用的国产镭神智能 LS C32-151A 型 32 线激光雷达是实际应用中常用的一种激光雷达。激光雷达扫描得到的点云数据不仅包括障碍物点云, 还包括了大量的地面点云。而地

收稿日期: 2023-01-12; 修回日期: 2023-01-29; 录用日期: 2023-02-16; 网络首发日期: 2023-03-09

基金项目: 国家自然科学基金(62001346)、CASC 多传感器探测与识别技术研发中心种子基金(ZZJJ202102)

通信作者: \*rxie@mail.xidian.edu.cn

面点往往可视为后续目标检测、识别和跟踪等任务的冗余信息,所以首先需要对地面点进行分割<sup>[1]</sup>。此外,地面分割的准确性和实时性直接决定了无人车辆的安全行驶速度。低时延使系统可以接收更多的激光雷达扫描数据,高精度可以恢复更真实的周围环境。

现有的地面分割算法根据分割策略可分为 6 种:基于栅格地图的方法、基于拟合的方法、基于相邻扫描线特征的方法、基于深度图像的方法、基于马尔可夫随机场的方法、基于深度学习的方法。基于栅格地图的方法将三维点云投影到对应的二维栅格中,通过计算栅格内点的平均高度<sup>[2]</sup>、高度差<sup>[3]</sup>及最大相对高度<sup>[4]</sup>等特性表征栅格的属性,然后与预先设定的阈值进行对比,实现对栅格的标记。栅格地图法易实现、计算速度快,但当障碍物边界点云与栅格边缘重合时存在欠分割的风险,且易受到悬浮障碍物影响。基于拟合的方法包括 2 类:全局拟合和局部拟合。全局拟合方法假定地面是一个完整的平面,使用 RANSAC 算法计算平面模型参数,直接拟合整个场景<sup>[5]</sup>。Hermas 等<sup>[6]</sup>选取初始种子点集作为初始平面,利用设定的阈值不断迭代填充点集,以优化平面,最终完成地面点云的分割。局部拟合方法将整个场景分成几个部分,并在每个部分中拟合地面。如 Narksri 等<sup>[7]</sup>将点云沿着车辆行进方向划分为前后左右 4 个区域,梅圣明等<sup>[8]</sup>结合道路分布特征和激光雷达线束分布特征,将点云划分成多区域完成地面分割。这些方法能有效改善分割效果,但由于计算多个平面模型参数,实时性较差。基于相邻扫描线特征的方法根据激光雷达成像特性通过分析相邻点之间角度和高度差异<sup>[9]</sup>来分离出地面点。基于深度图像的方法将 3D 点云转换为 2.5D 深度图像,利用不同区域图像间属性的差异特征完成分割<sup>[10]</sup>。如 Bogoslavskyi 等<sup>[11]</sup>利用广度优先搜索标记出地面连通区域,Shen 等<sup>[12]</sup>将点云投影到 RGB 图像,然后使用卷积重新划分点云归属。基于马尔可夫随机场(MRF)的方法不仅考虑了点云中单个点属于某个分类的可能性,而且考虑了相邻点之间分类的连续性。然而,基于信念传播等的迭代算法求解 MRF 的计算成本较高,速度较慢,且难以保证最优结果<sup>[13]</sup>。基于深度学习的方法通过利用神经网络直接分割地面点云,如 PointNet<sup>[14]</sup>、PointPillars<sup>[15]</sup>等可以端到端处理点云数据的网络,这些方法在开源数据集上表现极好,但大量标注的点云数据和计算资源、较差的网络可解释性限制了深度学习方法在车上的应用。

上述地面分割算法存在场景单一特定、对参数设置依赖性强、计算复杂、实时性差等问题,因此本文提出了一种基于粗-细两阶段处理的地面分割算法。第一阶段,对点云按水平偏转角度进行扇形区域划分,自适应选出每个扇形区域的局部坡度阈值,最后依据局部坡度阈值和全局坡度阈值标记出大部分地面点和非地面点,完成粗分割。第二阶段,将粗分割后的点云投

影到 RGB 图像,使用图像膨胀算法生成“高置信度障碍物点”与“高置信度地面点”交界处的“未知分类点”,最后将点云划分成多区域,确定合适的距离阈值,选择有效点对“未知分类点”执行卷积滤波,以判别该部分点云的分类,并将处理后的 RGB 图像重新投影到点云中完成细分割。相比现有的地面分割算法,所提算法可有效处理有坡度的地面,提高地面点分割的准确性,且局部坡度阈值能自适应调整,鲁棒性强、实时性好,能够满足自动驾驶的需求。

## 2 基于粗-细两阶段处理的地面分割算法

所提地面分割算法分为两个阶段,如图 1 所示,第一阶段为基于坡度阈值的地面粗分割,包括划分扇形区域、自适应调整局部坡度阈值等处理;第二阶段为基于卷积滤波处理的地面细分割,包括将点云投影到 RGB 图像、图像膨胀获得“未知分类点”、划分多区域确定距离阈值、卷积滤波等处理。

### 2.1 基于坡度阈值的地面粗分割

基于坡度阈值的地面粗分割有如下几个步骤。

1) 划分扇形区域。将预处理后的点云数据划分为多个扇形区域,即将点云按照激光雷达水平角度分辨率  $\Delta\rho$  划分为  $360/\Delta\rho$  个扇形区域。通过计算每个点与 X 轴正方向的夹角,将这个夹角除以水平角度分辨率后向下取整,可确定每个点所属的扇形区域  $S_m$ ,表达式为

$$S_m = \left\lfloor \frac{\arctan(y_i/x_i) \times 180}{\Delta\rho} \right\rfloor, 0 \leq m \leq \frac{360}{\Delta\rho} - 1, (1)$$

式中: $x_i$ 为第  $i$  个点的横坐标, $y_i$ 为第  $i$  个点的纵坐标; $m$ 为该点所属扇形区域的索引; $\lfloor f(x) \rfloor$ 为向下取整函数。以使用的镭神 C32-151A 型激光雷达为例,水平角度分辨率为  $0.2^\circ$ ,可划分出 1800 个扇形区域。图 2 为点云扇形区域划分示意图。

每个扇形区域的点云组成一个集合,集合中的点云具有相同的水平偏转角度,对同一个扇形区域集合中的点到激光雷达原点的水平距离  $d_i$  由小到大进行排序。 $d_i$ 的表达式为

$$d_i = \sqrt{x_i^2 + y_i^2}. (2)$$

2) 局部坡度阈值自适应调整。文献[9]使用固定的局部坡度阈值、全局坡度阈值确定局部高度和全局高度,然后分别比较相邻点云高度差与局部高度、全局高度的大小关系,从而分离出地面点云。在实际环境中,路面整体趋于平缓,可以共用同一个全局坡度阈值,而局部路面往往存在不同起伏,因此需要自适应设置每个扇形区域的局部坡度阈值。

如图 3 所示,以第  $m$  个扇形区域中的点云为例,无人车辆行驶在有坡度路面上时,对激光线束上的点由内到外计算相邻点云角度  $\epsilon_{i+1}$  以及相对于激光雷达旋转中心原点的垂直角度  $\omega_i$ ,表达式为

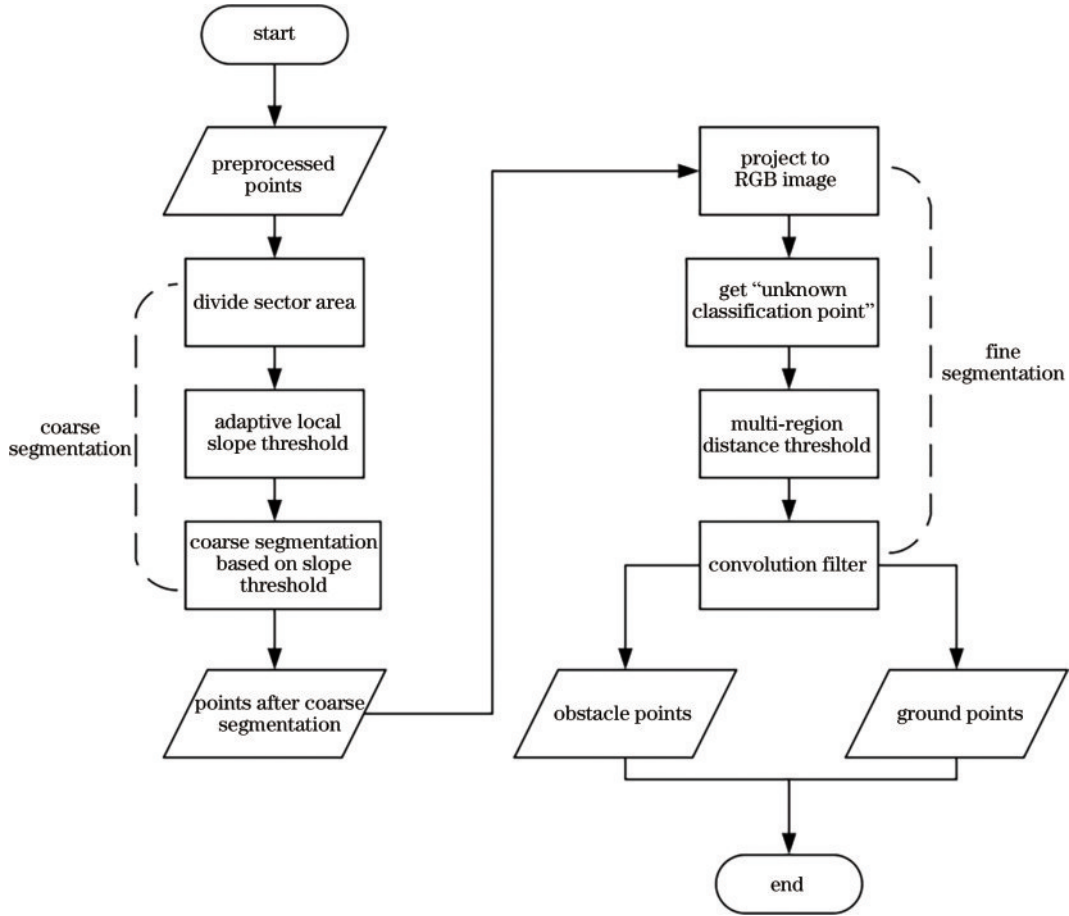


图 1 所提算法的整体流程

Fig. 1 Overall flow chart of the proposed algorithm

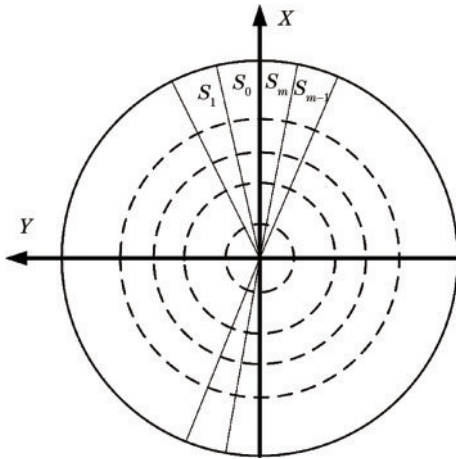


图 2 点云扇形区域划分示意图

Fig. 2 Schematic of point cloud fan-shaped area division

$$\epsilon_{i+1} = \left| \arctan \frac{z_{i+1} - z_i}{d_{i+1} - d_i} \right|, \quad (3)$$

$$\omega_i = \left| \arctan \frac{z_i + H}{\sqrt{x_i^2 + y_i^2}} \right|, \quad (4)$$

式中:  $(x_i, y_i, z_i)$  为第  $i$  个点坐标,  $(x_{i+1}, y_{i+1}, z_{i+1})$  为第  $i+1$  个点的坐标;  $H$  为激光雷达架设高度。在平坦路面或者有坡度路面时, 地面相邻点角度不会超过全局

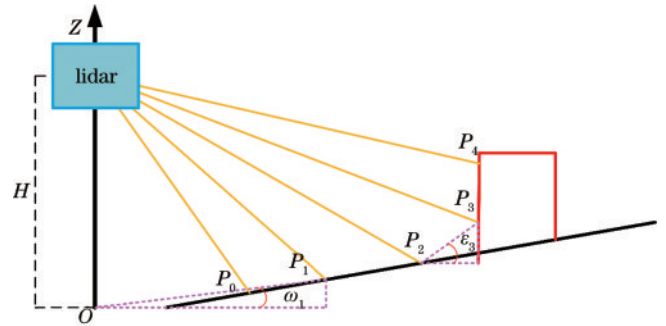


图 3 角度示意图

Fig. 3 Angle diagram

坡度阈值  $\theta_g$ 。如果当前点与后一个点的相邻角度大于  $\theta_g$  时, 可判断后一个点为障碍物点; 如果当前点垂直角度  $\omega_i \geq \theta_g$ , 也可判断当前点为障碍物点; 记录当前点序号为  $n$ , 结束后续点云角度计算。同时, 选择该区域内已计算的相邻点角度的最大值作为该扇形区域  $m$  内的局部角度阈值  $\theta_1^m$ , 表达式为

$$\begin{cases} n = i, & \epsilon_{i+1} > \theta_g \text{ or } \omega_i \geq \theta_g \\ \theta_1^m = \max \{ \epsilon_1, \epsilon_2, \dots, \epsilon_n \} \end{cases} \quad (5)$$

对扇形区域中最内侧激光束点可能存在的特殊情况进行分析, 若该点相对于激光雷达旋转中心原点的垂直角度  $\omega_0 \geq \theta_g$ , 则表示该激光点是障碍物点, 直接



结束后续点相邻角度的计算,并将该扇形区域的局部坡度阈值  $\theta_1^m$  设定为最小局部坡度值  $\theta_{\min}$ ,以防止对障碍物点云的误分割。

3) 地面点云与非地面点云判断。以第  $m$  个扇形区域为例,判断区域中两个相邻点角度正切值和局部坡度阈值是否满足

$$-\tan \theta_1^m \leq \frac{z_i - z_{i-1}}{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}} \leq \tan \theta_1^m, \quad (6)$$

式中:  $(x_{i-1}, y_{i-1}, z_{i-1})$  为第  $i-1$  个点的坐标;  $\theta_1^m$  为第  $m$  个扇形区域的局部坡度阈值。若前一个点即第  $i-1$  个点为地面点,且当前点即第  $i$  个点满足式(6),则当前点也为地面点;若前一个点为非地面点或者当前点不满足式(6),则继续按照式(7)比较当前点垂直角度正切值和全局坡度阈值的关系,

$$-\tan \theta_g \leq \frac{z_i + H}{\sqrt{x_i^2 + y_i^2}} \leq \tan \theta_g, \quad (7)$$

若当前点满足式(7),则判断当前点为地面点,否则为非地面点。依据式(6)和式(7)的判断条件,依次对每个扇形区域进行判断,即可实现对地面点的粗分割。第一阶段地面粗分割主要依赖全局坡度阈值  $\theta_g$  和局部坡度阈值  $\theta_1$ ,每个扇形区域的局部坡度阈值通过比较某区域内相邻点角度得到自适应确定。算法仅需设定激光雷达架设高度  $H$ 、全局坡度阈值  $\theta_g$  及最小局部坡度阈值  $\theta_{\min}$ 。所提算法的思想是:基于粗-细两阶段的分割,允许第一阶段粗分割的结果呈现过分割(over-segmentation),即结果包含部分非地面点,全局坡度阈值  $\theta_g$  可设置为覆盖大部分场景的最大阈值,最小局部坡度阈值  $\theta_{\min}$  针对特殊情况设置,所以不区分平坦路面和复杂坡度路面场景。据《公路工程技术标准(JTG-B01-2014)》的规定,城市公路最大坡度为  $5.14^\circ$ ,高速公路为  $1.72^\circ$ <sup>[8]</sup>,考虑实际路况中往往存在特殊地形,全局坡度阈值  $\theta_g$  设为  $10^\circ$ ,最小局部坡度阈值  $\theta_{\min}$  设为  $1.72^\circ$ 。第一阶段的粗分割算法对有坡度的地面分割较好,但仍包含了部分非地面点,粗分割的结果将作为第二阶段细分割的输入。

## 2.2 基于卷积滤波处理的地面细分割

将第一阶段粗分割后标记的点云投影到大小为  $R \times C$  的 RGB 图像中,为描述方便,将图像坐标系的原点从左上移到左下。其中  $R$  表示激光雷达线束数, $C$  表示激光雷达中一个发射器旋转  $360^\circ$  生成的激光点数。生成的 RGB 图像类似文献[11]所提出的深度图,不同之处在于深度图以每个点到激光雷达的距离作为像素值,而本文投影得到的 RGB 图像像素值根据粗分割后每个点的标记分类确定,即粗分割中标记的非地面点认为是“高置信度障碍物点”,在 RGB 图像中像素值为红色  $[255, 0, 0]$ ,而标记的地面点认为是“高置信度地面点”,在 RGB 图像中像素值为绿色  $[0, 255, 0]$ ,

空缺点则为黑色。

RGB 图像的行数为激光雷达线束数,即最下方表示垂直偏转角度最小的激光束,由下至上依次增大,每一行点云由相同的激光发射器生成,具有相同的垂直偏转角度。RGB 图像的列数即为一个发射器旋转  $360^\circ$  生成的激光点数,表示水平偏转角度依次增大,每一列点云具有相同的水平偏转角度。因而可根据每个点的水平偏转角度、垂直偏转角度确定第  $i$  个点在 RGB 图像中的位置  $(r, c)$  和像素值  $p_{(r,c)}$ ,表达式为

$$c = \left\lfloor \frac{\arctan(y_i/x_i)}{\Delta\rho} \times \frac{180}{\pi} \right\rfloor, \quad 0 \leq c \leq \frac{360}{\Delta\rho} - 1, \quad (8)$$

$$r = f \left( \arctan \frac{z_i}{\sqrt{x_i^2 + y_i^2}} \right), \quad 0 \leq r \leq R - 1, \quad (9)$$

$$p_{(r,c)} = \begin{cases} [255, 0, 0], & \text{if this point is marked as obstacle} \\ [0, 255, 0], & \text{if this point is marked as ground} \end{cases}, \quad (10)$$

式中:式(8)与式(1)一致;函数  $f(\cdot)$  为参照激光雷达垂直角度分布表,得出第  $i$  个点垂直偏转角度所对应激光束序列的映射函数。基于此方法,可以快速确定点云中某个点在 RGB 图像中的位置,也可从 RGB 图像中某个像素点得到对应的点云信息。点云投影后获得的 RGB 图像如图 4 所示。

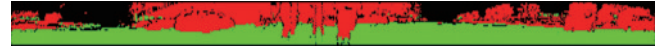


图 4 粗分割点云的 RGB 图像

Fig. 4 RGB image of the coarsely segmented point cloud

第一阶段粗分割获得的地面点云包含了部分非地面障碍物点,也就是产生了过分割(over-segmentation)。不准确的分割通常发生在地面点云与障碍物点云的交界处。所以,对代表“高置信度障碍物点云”的红色通道( $C_r$ )和代表“高置信度地面点云”的绿色通道( $C_g$ )附近点的分类需要进行再判别,将这一部分点云记作“未知分类点”。使用图像膨胀算法扩张 RGB 图像的红色通道,即扩大“高置信度障碍物点云”个数,表达式为

$$C_r(x, y) = \max_{x', y': \text{element}(x', y') \neq 0} C_r(x + x', y + y'), \quad (11)$$

扩张的红色通道与绿色通道产生的交叉点像素变为黄色  $[255, 255, 0]$ ,将这部分交叉点的像素值更改为蓝色  $[0, 0, 255]$  以表示“未知分类点”。原图像中为黑色点的像素值保持不变。如图 5 所示,蓝色通道( $C_b$ )存储“未知分类点”。

属于同一对象点之间的距离小于属于不同对象点

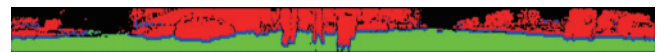


图 5 包含“未知分类点”的 RGB 图像

Fig. 5 RGB image containing “unknown classification points”

之间的距离,可以通过比较“未知分类点”与周围不同“高置信度点”的距离来确定它们的最终分类。因此,文献[12]使用  $W$  来表示像素点的权重,并根据“未知分类点”相邻像素的空间关系对 RGB 图像进行平滑处理。最后,整个图像的分割变成了一个卷积滤波过程,使卷积核在 RGB 图像上滑动,依次更新“未知分类点”

的归属。卷积核大小决定了卷积运算的候选点数,其大小取决于图像膨胀算法选择的内核大小,即通道扩张的大小。对于低线束激光雷达,内核选择不宜过大,否则会误判已正确分割的地面点。高线束激光雷达点云更加密集,可适当选择较大的内核。以大小为  $5 \times 5$  的卷积核为例,其作用在“未知分类点” $p_{(r,c)}$  上时,有

$$\text{Kernel}_{(r,c)} = \begin{bmatrix} W_{(r+2,c-2)} & W_{(r+2,c-1)} & W_{(r+2,c)} & W_{(r+2,c+1)} & W_{(r+2,c+2)} \\ W_{(r+1,c-2)} & W_{(r+1,c-1)} & W_{(r+1,c)} & W_{(r+1,c+1)} & W_{(r+1,c+2)} \\ W_{(r,c-2)} & W_{(r,c-1)} & 0 & W_{(r,c+1)} & W_{(r,c+2)} \\ W_{(r-1,c-2)} & W_{(r-1,c-1)} & W_{(r-1,c)} & W_{(r-1,c+1)} & W_{(r-1,c+2)} \\ W_{(r-2,c-2)} & W_{(r-2,c-1)} & W_{(r-2,c)} & W_{(r-2,c+1)} & W_{(r-2,c+2)} \end{bmatrix}, \quad (12)$$

式中:  $W(\cdot)$  表示像素点  $p(\cdot)$  的归一化权重。权重大小与“高置信度点”和“未知分类点”之间的欧氏距离  $\Delta d$  成负相关,因此可以选择高斯核函数计算像素点未归一化的权重  $w$ 。通过设置距离阈值  $d_{th}$ ,防止过远的点的干扰,且只有已标记为“高置信度点”的点云参与计算,之后计算归一化的权重,表达式为

$$w_{(r+i,c+j)} = \begin{cases} 0, & \Delta d > d_{th} \text{ or } (i=0, j=0) \text{ or } p_{(r+i,c+j)} = [0, 0, 255] \\ \exp\left(-\frac{|\Delta d|^2}{2\sigma^2}\right), & \text{otherwise} \end{cases}, \quad (13)$$

$$W_{(r+i,c+j)} = \frac{w_{(r+i,c+j)}}{\sum_{u=-2, v=-2}^{u=2, v=2} w_{(r+u,c+v)}}, \quad (14)$$

式中:  $\sigma$  控制高斯核函数的局部作用范围。文献[12]使用固定的距离阈值  $d_{th}$  防止过远点的干扰,但考虑到激光雷达点云近密远疏的特性,RGB 图像底部点云的密度要远大于顶部,固定的距离阈值过大会造成远处无效点被纳入卷积滤波的情况,而过小又会造成近处有效点未被选择的情况。当车辆靠近护栏时,水平偏转角为  $\delta_i$  的激光线束打在栏杆上,如图 6 短实线激光线束所示,水平偏角度与  $\delta_i$  相邻的激光线束穿过护栏间的空隙打在地面上,如图 6 长实线激光线束所示。在 RGB 图像上,这些点处于是否参与卷积滤波的候选邻域,使用过大的距离阈值,则 RGB 图像中靠近护栏上部的区域会选择相对较远的障碍物点参与卷积计算,

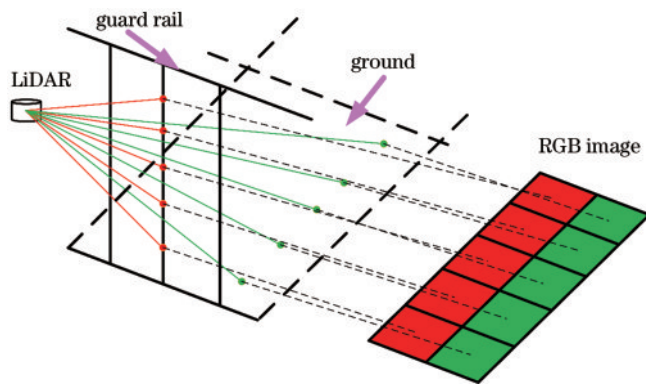


图 6 护栏障碍物点云与 RGB 图像示意图

Fig. 6 Diagram of guardrail obstacle point cloud and RGB image

而使用过小的距离阈值,RGB 图像中靠近护栏底部的区域则会抛弃有效的“高置信度点”,进而造成误检。

因而本文按水平距离  $d$  设定区域阈值  $r_1, r_2, \dots, r_n$  后将点云划分成不同的同心圆区域  $A_1, A_2, \dots, A_n$ , 然后在不同区域设置不同距离阈值  $d_{th1}, d_{th2}, \dots, d_{thn}$ , 如图 7 所示。

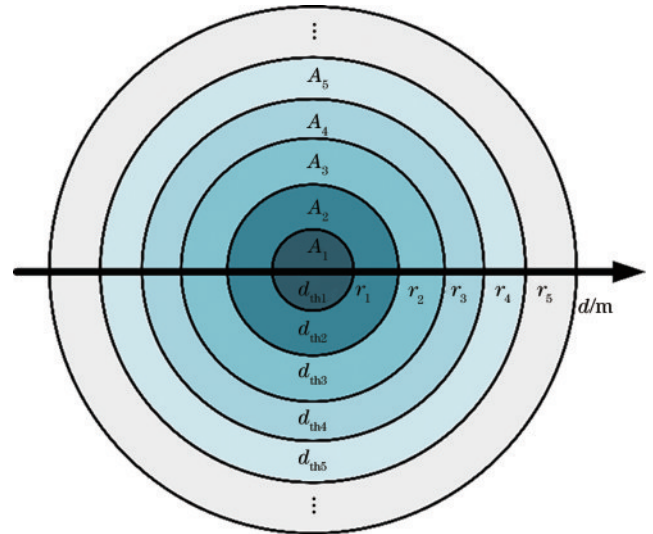


图 7 多区域距离阈值示意图

Fig. 7 Diagram of multi-region distance threshold

卷积操作从 RGB 图像底部到上部依次执行。因为只需要关心“未知分类点”的归属,所以当卷积核遇到“高置信度点”时,跳过计算。当卷积核作用在“未知分类点”上时,首先按式(2)计算水平距离,确定该点所划分的区域,进而确定距离阈值  $d_{th}$ 。然后按式(15)分

别计算红色通道“高置信度障碍物点”的卷积结果  $S_r(r, c)$  和绿色通道“高置信度地面点”的卷积结果

$S_g(r, c)$ , 比较两者卷积结果, 以更新“未知分类点”的像素值。

$$\begin{cases} S_r(r, c) = C_r(r, c) * \text{Kernel}_{(r,c)} = \sum_{i=-2, j=-2}^{i=2, j=2} C_r(r+i, c+j) \cdot W_{(r+i, c+j)} \\ S_g(r, c) = C_g(r, c) * \text{Kernel}_{(r,c)} = \sum_{i=-2, j=-2}^{i=2, j=2} C_g(r+i, c+j) \cdot W_{(r+i, c+j)} \end{cases}, \quad (15)$$

如果  $S_r(r, c) > S_g(r, c)$ , 则认为  $p_{(r,c)}$  属于障碍物点, 图像坐标  $(r, c)$  的像素值更新为  $[255, 0, 0]$ ; 否则, 像素值变为  $[0, 255, 0]$ , 认为  $p_{(r,c)}$  属于地面点。卷积滤波过程如图 8 所示。完成整幅 RGB 图像的卷积滤波后, 原先

的“未知分类点”也重新确定类别归属, 图像最终的平滑结果如图 9 所示。将 RGB 图像重新投影到点云中即可获得精确的地面分割结果。

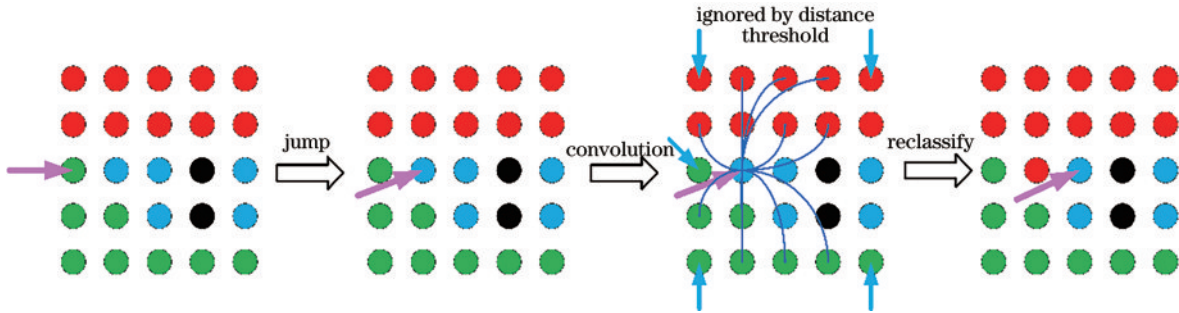


图 8 卷积滤波过程示意图

Fig. 8 Diagram of the convolution filtering process

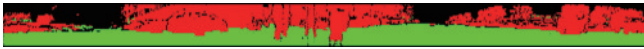


图 9 细分割点云的 RGB 图像

Fig. 9 RGB image of finely segmented point clouds

综合两个阶段的算法流程, 粗分割阶段输入预处理后的点云, 输出的地面点云产生了过分割, 即包含部分非地面点, 粗分割的结果又作为细分割阶段的输入点云。影响细分割结果准确性的重要参数是卷积核的大小与多区域距离阈值  $d_{th_i}$ 。卷积核大小限制参与卷积运算的候选邻域, 而多区域距离阈值  $d_{th_i}$  从候选邻域中选择有效的点参与卷积计算。细分割能对粗分割结果中误分割的点进行修正, 但也要求粗分割结果保证一定的准确性, 粗分割的准确性依赖全局坡度阈值  $\theta_g$  与局部坡度阈值  $\theta_l$ 。经过两个阶段处理后, 获得最终的分割结果。

### 3 实验结果及分析

#### 3.1 数据集与评价指标

分别采用仿真实验数据和实测数据, 验证所提算法的有效性和可靠性。仿真实验环境在机器人开发平台的 GAZEBO 中搭建, 仿真车辆底盘为四轮差速运动底盘, 所搭载的仿真激光雷达为 32 线 Velodyne HDL 32E 雷达, 如图 10 所示。

实测数据为项目组无人驾驶车辆搭载的镭神智能 LS C32-151A 型的 32 线激光雷达采集的点云数据。

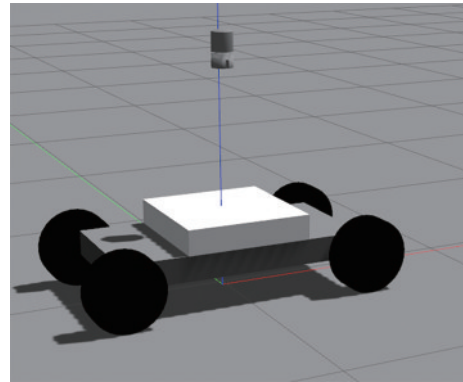


图 10 仿真四轮车平台

Fig. 10 Simulated four-wheel vehicle platform

该激光雷达的采集频率为 10 Hz, 垂直角度分辨率为  $1^\circ$ , 垂直视场角范围为  $-16^\circ \sim 15^\circ$ , 水平角度分辨率为  $0.2^\circ$ 。激光雷达扫描一周产生约有 5.76 万个激光点。实验场景中, 由于激光雷达安装在车辆顶部, 后续检测任务不需要使用过高区域的点云, 同时为了避免自车反射点干扰, 因此对原始数据进行裁剪。原始点云数据中激光雷达设备本身生成的测量噪声属于无用数据, 其结构特征是空间密度小且分布范围广<sup>[16]</sup>, 使用统计滤波算法去除噪声, 以保留更有效的点云数据。

将数据人为标记为非地面点与地面点, 参考混淆矩阵, 采用精确率  $P_{precision}$  和召回率  $P_{recall}$  作为评价指标, 计算公式分别为



$$P_{\text{precision}} = \frac{N_{\text{TP}}}{N_{\text{TP}} + N_{\text{FP}}}, \quad (16)$$

$$P_{\text{recall}} = \frac{N_{\text{TP}}}{N_{\text{TP}} + N_{\text{FN}}}, \quad (17)$$

式中:  $N_{\text{TP}}$  表示正确分割为地面点的数量;  $N_{\text{FP}}$  表示障碍物点误分割为地面点的数量;  $N_{\text{FN}}$  表示地面点误分割为障碍物点的数量。  $P_{\text{precision}}$  表示被分割为地面点的点云中真实地面点的比例,  $P_{\text{recall}}$  表示标定数据中真实地面点被提取为地面点的比例。如果某种算法的地面分割效果很好, 对应的精确率  $P_{\text{precision}}$  和召回率  $P_{\text{recall}}$  表现较大。

所提算法参数设置: 第一阶段粗分割的全局坡度阈值  $\theta_g$  为  $10^\circ$ , 最小局部坡度阈值  $\theta_{\text{min}}$  为  $1.72^\circ$ ; 第二阶段将空间点云按水平距离  $d_1 = 10 \text{ m}$ 、 $d_2 = 20 \text{ m}$ 、 $d_3 = 30 \text{ m}$ 、 $d_4 = 40 \text{ m}$  划分 5 个区域, 每个区间内对应的距离阈值  $d_{\text{th}}$  为  $0.2 \text{ m}$ 、 $0.4 \text{ m}$ 、 $0.6 \text{ m}$ 、 $0.8 \text{ m}$ 、 $1 \text{ m}$ ; 选择一个  $5 \times 5$  的内核进行图像膨胀, 并选择 24 像素的邻域进行卷积处理。用 C++ 编程语言编写软件, 操作系统为 Ubuntu 18.04, 开发环境依赖于 ROS Melodic 和 PCL 1.8, 车载工控机配置为 Intel i7-10750H 2.6 GHz, 16 GB 内存, 显卡为 RTX 2070。

### 3.2 仿真实验结果

仿真实验分别模拟了不同的平坦路面场景和斜坡路面场景, 部分代表场景如图 11(a) 和图 11(b) 所示。分割结果如图 11(c) 和图 11(d) 所示, 绿色为地面点, 红色为障碍物点。如图 11(c) 所示, 所提算法对平坦路面场景的分割结果较好, 不仅分割出了当前车辆行驶的地面, 也能分割出道路两旁可通行的平面区域和设置的障碍物。如图 11(d) 中方框所示, 对于斜坡路面场景, 所提算法较完整地分离出了车辆正前方的斜坡, 同时也能分离出设置在斜坡上的障碍物, 而对车辆右侧坡度较大而无法通行的斜坡, 则识别为障碍物点。

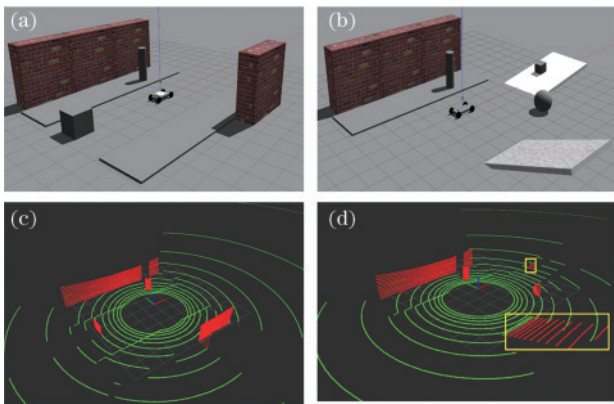


图 11 模拟的不同路面场景与分割结果。(a) 平坦路面场景; (b) 斜坡路面场景; (c) 平坦路面分割结果; (d) 斜坡路面分割结果

Fig. 11 Simulated road scenes and segmentation effect. (a) Flat road scene; (b) slope road scene; (c) flat road segmentation result; (d) slope road segmentation result

从仿真数据中选取平坦路面和斜坡路面场景各 10 帧数据, 评估所提算法的分割效果与平均耗时, 结果如表 1 所示。由于仿真数据比实测数据更理想, 点云成像质量更好, 所提算法在两种仿真场景下的精确率和召回率均维持较高数值, 且处理每帧数据平均耗时均不超过 40 ms, 满足自动驾驶的实时性需求。

表 1 所提算法在不同仿真场景下的分割结果

Table 1 Segmentation results of the proposed algorithm in different simulation scenes

Scene	$P_{\text{precision}} / \%$	$P_{\text{recall}} / \%$	Average consuming time /ms
Flat road	99.1	98.2	10.34
Slope road	98.4	97.8	11.28

### 3.3 实测实验结果

实测数据采集于深圳宝安区某会展中心附近, 分别从中选择平坦路面场景和有坡度复杂路面场景进行实验。

#### 3.3.1 两阶段结果对比

首先验证所提算法在各阶段的有效性, 在实测点云数据集中选出平坦路面、复杂坡度路面进行实验, 其中有代表性的两个场景的处理结果如图 12 所示, 绿色为地面点, 红色为障碍物点。图 12(a) 和图 12(b) 分别是平坦路面一阶段粗分割和二阶段细分分割结果。由图 12(a) 可以看出, 在平坦路面粗分割阶段, 所提算法剔除了大部分地面点, 但同时使得部分障碍物点被标记为地面点, 如图 12(a) 中方框所示。经过第二阶段细分分割后, 图 12(b) 方框中的误分割点云被重新标记为障碍物点, 修正了粗分割的结果。图 12(c) 和图 12(d) 分别是复杂坡度路面一阶段粗分割和二阶段细分分割结果。图 12(c) 中车辆前方的长缓坡与左侧的陡坡均能得到完整分割, 但与图 12(a) 类似, 图 12(c) 方框中同样有部分误分割, 这部分点云在图 12(d) 中已被正确分割。图 12(c) 中间小方框中部分绿色点云为较远处地面点, 由于二阶段算法中设置的分区域距离阈值限制, 这部分点云并不会受 RGB 图像中超过距离阈值的相邻树木障碍物点的影响, 所以在图 12(d) 中间小方框中仍为地面点。

从采集数据中选取平坦路面和复杂坡度路面各 10 帧数据, 来评估所提算法各阶段的分割效果, 结果如表 2 所示。

表 2 结果表明: 在平坦路面与复杂坡度路面场景中, 第一阶段自适应坡度阈值地面分割算法已经取得了较高的精确率和召回率, 但复杂坡度路面场景下局部坡度阈值自适应调整后, 仍存在部分误分割, 表现不如平坦路面场景; 经过第二阶段基于卷积滤波的细分分割后, 部分误分割点云被重新标记, 精确率和召回率均获得了提升, 尤其是所提算法对复杂坡度路面场景的精确率提升了 3.1%。虽然实测点云数据中存在部分

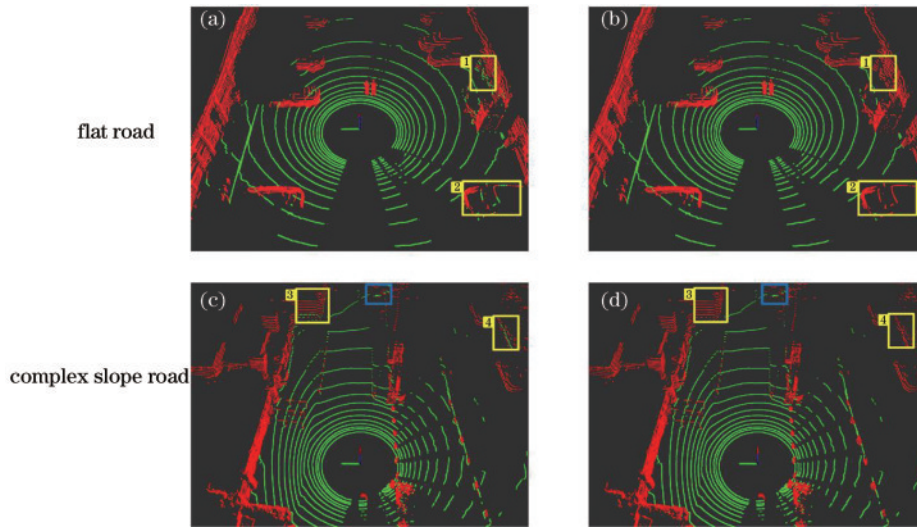


图 12 所提算法各阶段在不同场景下的分割效果。(a) (c)第一阶段结果;(b) (d)第二阶段结果  
 Fig. 12 Segmentation effect of each stage of the proposed algorithm in different scenes. (a) (c) Results of the first stage;  
 (b) (d) results of the second stage

表 2 所提算法各阶段在不同场景下的分割结果

Table 2 Segmentation results for each stage of the proposed algorithm in different scenes

Scene	Stage	$P_{precision} / \%$	$P_{recall} / \%$
Flat road	The first stage	95.6	94.5
	Second stage	97.5	96.8
Complex slope road	The first stage	93.2	92.3
	Second stage	96.1	95.8

能表明所提算法在各个阶段均可获得较好的地面分割效果。

### 3.3.2 与其他方法对比

对所提算法与文献[6]基于平面拟合的方法和文献[12]中的方法进行地面点云分割效果对比,部分结果如图 13 所示,绿色为地面点,红色为障碍物点,方框标记了对比区域。图 13(a)~(c)为平坦路面场景,图 13(d)~(f)为缓坡路面场景,图 13(g)~(i)为陡坡路面场景。

噪声,导致精确率和召回率表现略低于仿真数据,但仍

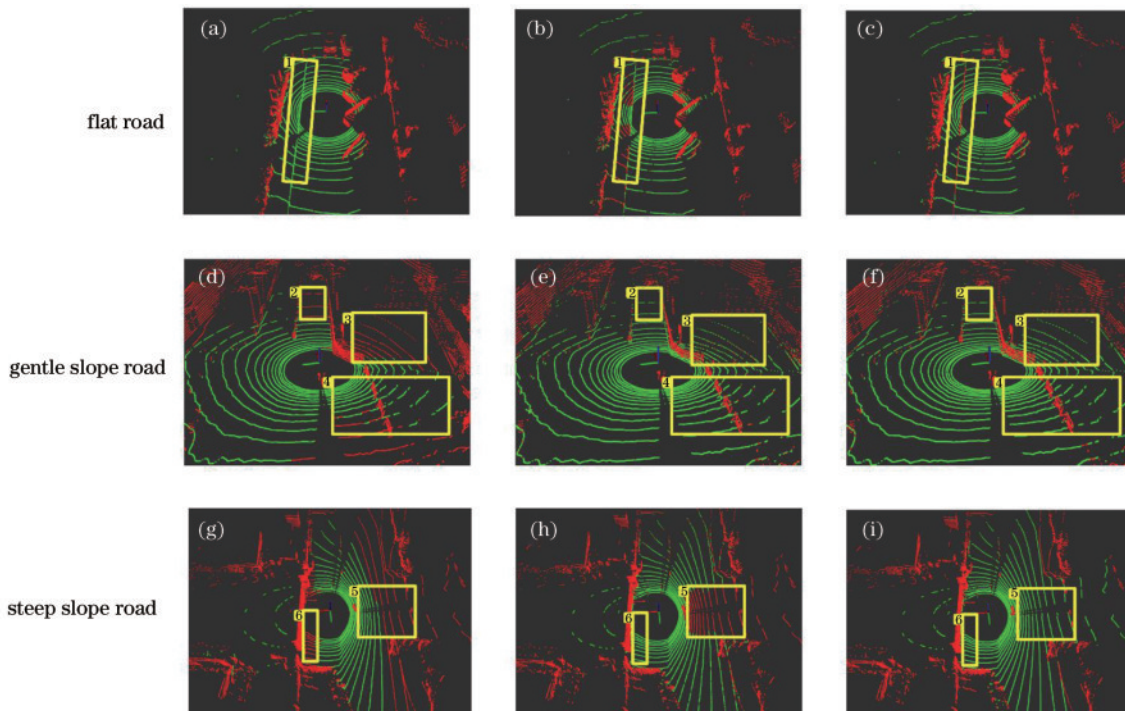


图 13 3 种算法在不同场景下的分割效果。(a) (d) (g) 文献[6]中的方法;(b) (e) (h) 文献[12]中的方法;(c) (f) (i)所提方法  
 Fig. 13 Segmentation effects of three algorithms in different scenes. (a) (d) (g) Method in Ref. [6]; (b) (e) (h) method in Ref. [12];  
 (c) (f) (i) proposed method



文献[6]中的方法在完整的平坦路面上分割效果较好,但未能分割出人行道的台阶,如图 13(a)方框 1 所示;文献[12]中的方法使用栅格高度差完成地面点粗分割,而此方法会将与地面平行的高平台型障碍物误判为地面,如图 13(b)方框 1 所示;所提方法能较好地完成平坦路面分割,还能识别出有一定高度的台阶。如图 13(e)和图 13(f)所示,在缓坡路面中,文献[12]中的方法和所提方法分割效果较好,能完整分离出斜坡路面,但如图 13(e)方框 3 和 4 显示,在激光雷达靠近护栏障碍物时,障碍物后的地面点云中产生了部分离散稀疏的非地面点,作为对比,所提方法将点云划分为多区域后按区域确定距离阈值,筛选参与卷积计算的“高置信度点”,因而图 13(f)方框 3 和 4 中并没有出现误分割点云。图 13(e)和图 13(f)中护栏实物如图 14 所示,方框 3 和 4 点云局部细节如图 15 所示。文献[6]中的方法拟合整个路面方程,造成大量的误分割,如图 13(d)方框 2、3、4 所示。在陡坡路面场景下,文献[6]中的方法的分割效果依然不理想,如图 13(g)方框 5、6 所示;文献[12]中的方法虽增加了栅格间的坡度约束,但并未考虑全局坡度情况,因而当车辆行驶在陡

坡上,后方地面点云产生大量误检,如图 13(h)方框 5 所示;所提算法在陡坡路面场景下保持了较好的分割效果。

为进一步验证所提算法的效果,从采集数据中选取平坦路面和复杂坡度路面各 10 帧数据,与其他算法的分割结果进行定量分析,结果如表 3 所示。从表 3 可以看出:所提算法保持了较高的精确率,比其他两种算法均有所提高,且在两种场景下均维持在 96% 以上;在平坦路面场景下,所提算法召回率维持在最高;在复杂坡度路面下,所提算法的召回率较文献[12]中的方法提升了 4.1 个百分点,达到了 95.8%。结果表明,所提算法在两种场景下均保持了较高的精确率和召回率,鲁棒性较好。

表 3 不同场景下 3 种算法的分割结果

Table 3 Segmentation results of the three algorithms in different scenes

Scene	Method	$P_{precision} / \%$	$P_{recall} / \%$
Flat road	Method in Ref. [6]	95.8	94.7
	Method in Ref. [12]	96.8	95.9
	Proposed method	97.5	96.8
Complex slope road	Method in Ref. [6]	85.7	82.3
	Method in Ref. [12]	92.4	91.7
	Proposed method	96.1	95.8



图 14 护栏实物图

Fig. 14 Guardrail physical map

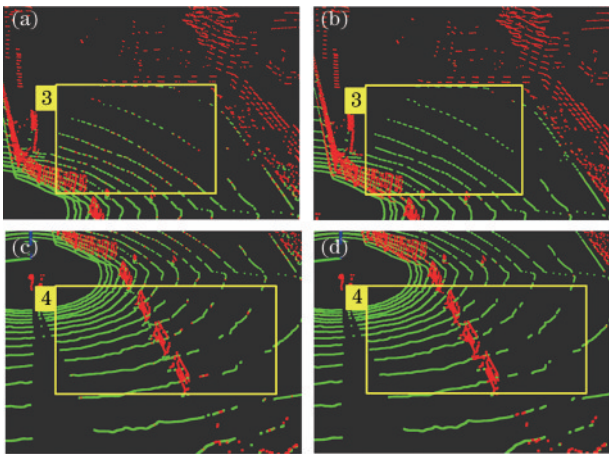


图 15 存在护栏障碍物时文献[12]中的方法和所提方法的分割效果。(a) (c) 文献[12]中的方法;(b) (d) 所提方法

Fig. 15 Segmentation effects of the method in Ref. [12] and the proposed method when guardrail obstacles exist.

(a) (c) Method in Ref. [12]; (b) (d) proposed method

地面分割算法不仅需要准确分割出地面,更需要满足车辆行驶实时性的要求以保证行驶安全。因而对上述几种算法的实时性进行对比,一帧点云数据处理时间用于评估算法的处理速度。本文采用的激光雷达采样频率为 10 Hz,采集一帧时间为 100 ms。一般认为低频激光雷达(采样频率 10 Hz)的地面分割处理时间不应超过 40 ms,高频激光雷达(采样频率  $\geq 20$  Hz)不应超过 20 ms。在点云数据中,筛选了不同路况的 1000 帧激光雷达数据进行对比,各算法的实际检测时间如图 16 所示,算法平均耗时如表 4 所示。由表 4 和图 16 可以看出,所提算法处理不同路况一帧数据的时间主要为 15~18 ms,不超过 40 ms。所提算法处理每

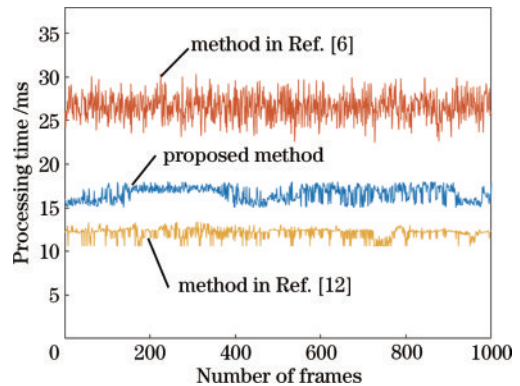


图 16 三种地面分割算法的耗时

Fig. 16 Consuming time of three ground segmentation algorithms

表 4 3 种地面分割算法的平均耗时

Table 4 Average consuming time of three ground segmentation algorithms

Parameter	Method in Ref. [6]	Method in Ref. [12]	Proposed method
Average consuming time/ms	26.63	12.27	16.57

帧的时间比文献[12]中的方法多 4.3 ms,但仍低于文献[6]中的方法,满足无人驾驶的实时性要求。综上所述,所提算法在地面分割任务中鲁棒性与实时性较好,具有高精确率和高召回率。

## 4 结 论

提出了一种基于粗-细两阶段处理的点云地面分割算法。该方法利用自适应坡度阈值对地面点云进行第一阶段粗分割,然后将点云投影到 RGB 图像,使用图像膨胀算法来获得地面点云与非地面点云交界处的“未知分类点”,将点云划分多区域后按区域确定距离阈值,筛选出参与卷积操作的“高置信度点”;然后对“未知分类点”执行卷积滤波,重新标记分类,进而完成第二阶段细分割。在平坦路面、复杂坡度路面场景下,所提算法的分割效果较好,有效分割出了坡度路面,且局部坡度阈值可以自适应调整,具有较好的鲁棒性,同时也保持了较好的实时性,满足实际工程应用需求。所提算法适用于激光雷达噪声较少、成像质量较好的场景,后续将研究在恶劣天气等激光雷达低信噪比场景下鲁棒性更好的地面分割算法。

## 参 考 文 献

- [1] 黄思源, 刘利民, 董健, 等. 车载激光雷达点云数据地面滤波算法综述[J]. 光电工程, 2020, 47(12): 3-14. Huang S Y, Liu L M, Dong J, et al. A review of ground filtering algorithms for vehicle LiDAR scan point cloud data[J]. Opto-Electronic Engineering, 2020, 47(12): 3-14.
- [2] Bekey G A. Springer handbook of robotics (B. siciliano and O. khatib; 2008)[book review][J]. IEEE Robotics & Automation Magazine, 2008, 15(3): 110.
- [3] Asvadi A, Peixoto P, Nunes U. Detection and tracking of moving objects using 2.5D motion grids[C]//2015 IEEE 18th International Conference on Intelligent Transportation Systems, September 15-18, 2015, Gran Canaria, Spain. New York: IEEE Press, 2015: 788-793.
- [4] Li Q Q, Zhang L, Mao Q Z, et al. Motion field estimation for a dynamic scene using a 3D LiDAR[J]. Sensors, 2014, 14(9): 16672-16691.
- [5] Li L, Yang F, Zhu 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.
- [6] Hermas D, Izzet I, Papanikolopoulos N. Fast segmentation of 3D point clouds: a paradigm on LiDAR

data for autonomous vehicle applications[C]//2017 IEEE International Conference on Robotics and Automation (ICRA), May 29-June 3, 2017, Singapore. New York: IEEE Press, 2017: 5067-5073.

- [7] Narksri P, Takeuchi E, Ninomiya Y, et al. A slope-robust cascaded ground segmentation in 3D point cloud for autonomous vehicles[C]//2018 21st International Conference on Intelligent Transportation Systems (ITSC), November 4-7, 2018, Maui, HI, USA. New York: IEEE Press, 2018: 497-504.
- [8] 梅圣明, 黄妙华, 柳子晗, 等. 基于三维激光雷达的复杂场景中地面分割方法[J]. 激光与光电子学进展, 2022, 59(10): 1028003. Mei S M, Huang M H, Liu Z H, et al. The ground segmentation method in complex scenes based on three-dimensional lidar[J]. Laser & Optoelectronics Progress, 2022, 59(10): 1028003.
- [9] 冯绍权, 花向红, 段成文, 等. 一种自适应的坡度阈值地面点云分割方法[J]. 测绘科学, 2021, 46(1): 156-161. Feng S Q, Hua X H, Duan C W, et al. An adaptive slope threshold method for ground -point cloud segmentation[J]. Science of Surveying and Mapping, 2021, 46(1): 156-161.
- [10] 张凯, 于春磊, 赵亚丽, 等. 基于自适应阈值的三维激光点云地面分割算法研究[J]. 汽车工程, 2021, 43(7): 1005-1012. Zhang K, Yu C L, Zhao Y L, et al. Research on a ground segmentation algorithm based on adaptive thresholds for 3D laser point clouds[J]. Automotive Engineering, 2021, 43(7): 1005-1012.
- [11] Bogoslavskiy I, Stachniss C. Efficient online segmentation for sparse 3D laser scans[J]. PFG-Journal of Photogrammetry, Remote Sensing and Geoinformation Science, 2017, 85(1): 41-52.
- [12] Shen Z H, Liang H W, Lin L L, et al. Fast ground segmentation for 3D LiDAR point cloud based on jump-convolution-process[J]. Remote Sensing, 2021, 13(16): 3239.
- [13] Guo C Z, Sato W, Han L, et al. Graph-based 2D road representation of 3D point clouds for intelligent vehicles [C]//2011 IEEE Intelligent Vehicles Symposium (IV), June 5-9, 2011, Baden-Baden, Germany. New York: IEEE Press, 2011: 715-721.
- [14] Charles R Q, Hao S, Mo K C, et al. PointNet: deep learning on point sets for 3D classification and segmentation[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 77-85.
- [15] Lang A H, Vora S, Caesar H, et al. PointPillars: fast encoders for object detection from point clouds[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE Press, 2020: 12689-12697.
- [16] 史永喆. 基于三维激光雷达的障碍物目标检测与跟踪[D]. 合肥: 合肥工业大学, 2020. Shi Y Z. Obstacle target detection and tracking based on 3D LiDAR[D]. Hefei: Hefei University of Technology, 2020.