doi:10.3772/j.issn.1002-0470.2023.10.008

基于虚幻引擎与三维点云的列车障碍物检测仿真系统设计①

禹鑫燚② 何伟琪 崔朱帆 欧林林③

(浙江工业大学信息工程学院 杭州 310023)

摘 要随着轨道技术的高速发展,列车安全的主动防护技术在列车行驶的过程中极为 重要,例如非接触式障碍物检测系统。然而在真实列车运行的铁路环境中难以采集到带 有障碍物的场景,并存在采集实验数据效率低的问题,如图像和点云数据。本文设计了一 种具有虚拟激光雷达的轨道列车障碍物检测仿真系统,为轨道列车非接触式障碍物检测 的研究提供可靠的实验环境和点云采集传感器。为了精确识别列车前方轨道位置,提出 一种基于体素分割和稀疏卷积算子构建点云语义分割网络识别轨道点云的方法。基于最 小二乘法拟合离散的轨道点云获得连续的安全行车区域,并且基于平面分割和点云聚类 确定了障碍物的三维位置。通过采集仿真环境中3种轨道场景的点云,制作数据集用于 训练网络模型,并通过采集带有障碍物场景的点云测试所提出的障碍物检测方法。实验 结果验证了仿真系统可行性以及障碍物检测算法的有效性。

关键词 虚幻引擎;列车仿真环境;三维点云;障碍物检测;神经网络

0 引言

近年来,我国轨道交通的运输量不断增加,列车 运行速度也越来越快,据国家铁路局公布的数据显 示,铁路客货运量持续增长^[1]。铁道交通运输的发 展给人们的生活带来便利的同时,对列车运行的安 全性提出了更高的要求。轨道列车非接触式障碍物 检测,在列车安全行驶的过程中显得极为重要^[2]。 列车运行时依靠驾驶员的人为判断前方是否存在障 碍物,但无法避免人为因素导致潜在的危险。因此, 鉴于行车安全需要智能化保障的背景下,列车前方 障碍物检测是必要的。但是铁路交通运行环境封 闭,使用传感器采集真实轨道运行时的数据十分困 难。

基于仿真平台进行算法仿真测试,不仅可以提 高研究测试效率,还可以将列车与自然环境的交互 考虑在算法内。轨道列车仿真系统对于轨道列车的 障碍物检测研究具有重要的意义。Koh^[3]基于 Sim-Mobility 模拟平台设计了一个铁路仿真系统,可以根 据配置文件灵活地对多种场景和策略进行建模。郭 彦宏^[4]基于微软的列车模拟器(microsoft train simulate, MSTS)搭建了仿真平台并研究了列车控制算 法和列车自动驾驶仿真实验等,但缺少对轨道障碍 物检测的仿真研究。Tang^[5]基于 LabVIEW 软件创 建了一个火车检测和物体检测的仿真环境,并在铁 路站台的 2D 图像中检测障碍物实现列车进站安全 预警系统,对轨道交通安全起到有效的作用。目前 主流的轨道仿真系统过于理想化,未充分考虑真实 场景中的问题,尤其是在轨道障碍物实时检测方面。

根据上述分析,为满足轨道列车障碍物检测仿 真系统的模拟真实列车场景的需求,本文设计了轨 道列车障碍物检测仿真系统。首先,基于虚幻引擎 构建了仿真系统的虚拟仿真环境。通过 3D 建模技

① 国家重点研发计划(2018 YFB1308000)和浙江省自然科学基金(LY21F030018,LQ22F030021)资助项目。

② 男,1979年生,副教授;研究方向:智能学习与机器人系统;E-mail: yuxinyinet@163.com。
 ③ 通信作者,E-mail: linlinou@zjut.edu.cn。 (收稿日期:2022-01-20)

术,使用建模软件在虚幻引擎构建的场景中创建了 地形地貌并覆盖植被,在场景中搭建轨道和列车。 其次,根据真实传感器原理设计虚拟激光雷达传感 器并采集环境的实验数据。

针对列车行驶安全系统中的轨道障碍物检测问 题,现有的研究^[6-10]主要通过安装在列车头部的传 感器(如相机和激光雷达)获取环境数据进行障碍 物检测。基于图像的方法因数据获取便利和物体辨 识度高等优点被用于障碍物检测^[6-8].但是容易受 到极端天气与光照变化的影响,在雨、雪、光强瞬变 (进出隧道时)等情况下存在鲁棒性不足的问题。 激光雷达主动发射射线获取环境点云,不受天气与 光照影响,同时包含精确的目标物体的三维空间信 息,因此不断地被应用于环境感知领域^[9-11]。郭双 全^[9]提出采用雷达传感器实现铁路线路异物侵限 检测,通过构建轨道地图和设计检测区域模型来检 测障碍物,但不能满足实时障碍物检测的需求。蔚 增岩[10]提出一种适用于铁路环境的地面滤除方法 用于分离目标范围内的地面与障碍物,通过建立物 体模型库匹配障碍物类别,但只能用于近距离障碍 物检测。

传统的点云处理方法主要是通过人工设计的特 征来制定各个类别点云的统计规则和几何模型约 束,通常只适用于三维物体模型。深度学习技术的 发展^[12]为点云语义分割领域提供了新的技术路线. 可以分为基于体素和基于点的方法。PointNet^[13]是 一个开创性的网络框架,它直接对每个点执行全连 接操作,利用全连接网络和最大池化层来近似函数 的置换不变特性,解决了点云数据的排列顺序问题, 但忽略了点云局部特征的提取而导致复杂场景下精 度不高。Hu 等人^[14]提出了 RandLA-Net,其中随机 降采样有效地提高模型计算速度,但也容易受到噪 声点云影响。基于体素的方法将体素与 3D 卷积网 络相结合, VoxNet^[15]首先被用于点云的目标检测任 务中,它使用固定体积的立方体分割点云,但此方法 较难适应室外场景点云的稀疏性和密度易变等特 点。因此,Zhou 等人^[16]提出了圆柱形体素分割,将 公路场景的雷达点云根据距离动态划分到扇形体素 中,从而产生更均衡的体素与点云分布的关系,在语 义分割任务中获得了更高的精度。

针对图像传感器对光照条件敏感与图像数据在 空间信息检测上的缺陷,本文研究了基于三维点云 的轨道检测方法。首先,根据铁路场景雷达扫描视 角与点云数据稀疏性和密度易变的特点,分别设计 了锥形体素分割和点云局部特征聚合方法来提取铁 轨的高阶几何特征,提出了基于稀疏 3D 卷积的点 云语义分割网络模型对每个体素分类,根据体素与 点的索引关系将体素类别映射到对应的点云,实现 轨道点云分割。最后,通过拟合离散的轨道点云获 得连续的安全行车 3D 区域,基于随机抽样一致性 算法^[17]分割轨道平面,对区域内剩余点云聚类得到 障碍物的三维位置。

1 铁路仿真环境搭建

本节基于虚幻引擎(unreal engine, UE)开发 系统仿真环境,如图1所示,包括地形地貌、轨道列 车和障碍物,通过虚拟激光雷达传感器获取铁路环 境数据。地行地貌中有地形、植被等自然环境的元 素,还有轨道和运行的列车。构建仿真系统的目的 是验证轨道列车障碍物检测方法,需要获得仿真环 境的三维点云数据,因此设计了虚拟激光雷达传感 器。传感器被安装在列车运行的头部,获取列车运 行时环境中的三维点云,用来检测列车运行前方的 环境中是否有障碍物。



图1 沃西历兵师党及二维总公须取恒米

1.1 铁路环境构建

环境地形地貌承载整个虚拟仿真场景,是环境的基础。其包括山坡、草坪、岩壁、树林等自然环境

元素。地形地貌构建如图 2 所示,将建模软件制作 的高度图、实物模型、材质资源导入到环境中,使用 World Machine 制作基础地形,基于 UE 地形编辑系 统 Land Scape 进一步优化地形细节,并利用 Speed Tree 工具编辑树木模型并批量地添加到虚拟仿真环 境中,完成地形地貌的构建。基于 UE 提供的材质 库为地表设计添加草坪的材质纹理,通过 Land Scape 的地形修改组件去优化地形,雕刻和绘制山 坡和沟堑,对地形高度图生成的地形进行平滑处理。



在 Speed Tree 软件中为仿真环境设计制作绿 植。Speed Tree 使用节点编程模式,依次添加树干、 树枝、树叶并调整大小、节点数量、位姿等参数。给 树添加材质贴图,树干需要颜色贴图、法线贴图,树 叶另外需要透明贴图。将模型导出并添加到 UE 的 Land Scape 地形编辑器中,批量的设置添加树木到 仿真环境中,效果如图 3 所示,环境表现接近自然环 境。



图 3 地形地貌

除了自然环境,仿真环境中还包含轨道、电杆、 站台等静物以及在轨道上运行的列车。使用 3D Max 三维建模软件搭建轨道、列车、电线杆、站台等 模型,将模型按照要求转换为静态网格体导入到 UE 中,并布置在场景中。仿真环境中的障碍物模型有 人体、石头、正方体和圆柱体等。图4给出了导入完 成后的列车、枕木、铁轨、电杆和障碍物等模型。



图 4 轨道上的元素与障碍物模型

以单轨道的实际铁路场景为参考,仿真环境构 建结果如图5所示,左转轨道包含人、圆球和石块等 障碍物。



图 5 铁路仿真环境

1.2 虚拟激光雷达传感器获取点云数据

根据激光雷达扫描单元的结构不同,激光雷达 分为机械式、混合式和纯固态激光雷达。机械式激 光雷达因原理简单、360°扫描的特点,被广泛应用 于智能驾驶实验测试中。

混合式激光雷达把机械结构集成到体积较小的 硅基芯片上,激光发射元件在竖直方向上排布成激 光光源线阵,在竖直面内产生不同指向的激光光束, — 1079 — 通过微振镜改变单个发射器的发射角度,竖直面内 的激光不断旋转由"线"扫描形成"面",实现对被测 对象、环境的三维扫描。以 innovusion Jaguar Gen-1 激光雷达的参数为依据,其参数如表 1,基于混合式 激光雷达的原理,使用 UE 的光线追踪性能设计仿 真环境中获取数据的虚拟激光雷达传感器。

参数	值
扫描频率	6~10 Hz,可调
垂直视场角	40 °
水平视场角	65 °
测量精度	<3 cm
测量距离	200 m

虚拟激光雷达感知基于虚幻引擎的光线渲染系统,用于射击游戏中获取被射击目标。其原理如 图6所示,射线发出后会去在指定范围内对碰撞体 (三角形)求交,依据着色表对交点判断,如果交点 有效则触发碰撞(圆点)返回最近交点的距离。



以雷达的位置作为射线的起点,若触发碰撞则 得到原点到物体交点的距离,如图 7 所示,根据已知 的射线垂直角度 α 和水平角度 β,可以得到交点的 x、y 和 z 坐标值如式(1)所示。



$$x = r \cdot \cos\alpha \cdot \cos\beta$$

$$y = r \cdot \cos\alpha \cdot \sin\beta$$

$$z = r \cdot \sin\alpha$$

(1)

2 基于三维点云的障碍物检测方法

针对铁路场景列车运行的安全性问题,本节提出的基于三维点云障碍物检测方法分为2个部分: 轨道点云分割和列车安全行驶区域的障碍物判定。

本节提出一种基于深度学习的三维点云语义分 割网络与障碍物判定的方法,可以精确地检测出轨 道障碍物。方法组成如图 8 所示,包括轨道点云分 割和障碍物判定。首先对不同的铁路场景(例如直 轨道、弯轨道)分割出轨道点云;然后基于离散的轨 道点云和列车尺寸信息拟合出列车前方的安全行驶 3D 区域,对区域内的轨面点云进行地面分割;最后 对剩余的点云聚类并根据每个类别中点的数量判断 是否存在障碍物,若存在,则计算障碍物的 3D 位 置,完成障碍物检测。



2.1 轨道点云分割

点云语义分割网络分为基于体素的三维卷积方 法和基于投影的二维语义分割方法。投影法通常是 将点云投影到特定的平面或曲面上,再基于图像中 的语义分割网络得到点云分割结果。这类方法的最 终性能通常要低于基于体素的方法,因为经过投影 之后的点云会丢失三维空间的几何结构信息。传统 的三维卷积属于密集计算,复杂度达到 O(n³),而 点云在三维空间中的稀疏性导致三维卷积计算量 大,包含大量无效计算。基于稀疏 3D 卷积的方法 只对非空体素执行卷积,避免了对无效体素的计算, 从而加速整体卷积运算速度。

本文创建的仿真激光雷达安装在列车头部,有 一个固定的探测范围。如图9所示,激光雷达的扫 描区域为四棱锥形,水平和俯仰角度由仿真雷达的 参数决定。在点云采集过程中,由于距离、遮挡和相 对位置等因素,近距离的点云密度要比远区的大得 多。在相关的工作中,点云被划分为立方体体 素^[15],这将导致体素的点密度的非均衡性。基于扇 形的体素划分思想^[16],将点云在极坐标系下表示为

$$\alpha = \arctan(\frac{z}{\sqrt{x^2 + y^2}})$$

$$\beta = \arctan(\frac{y}{x})$$

$$r = \arctan(\sqrt{x^2 + y^2 + z^2})$$
(2)

式中α、β、r分别为点在极坐标系下的垂直角、水平 角和距离。然后进行体素划分,使得每个区域中都 包含相同数量的雷达射线。如图 10 所示,经过分割 后,在笛卡尔坐标系中显示为锥形的体素,随着扫描 范围内物体距离变远,每个体素覆盖的体积更广。



图 9 雷达探测范围示意图

最后将每个点划分到对应的体素中,即给每个 点配对体素编号,具体步骤如下2步所述。

(1)给定一帧点云 $P = \{p_n \mid n = 1, \dots, N\}$, 设 定体素分割的网格大小为 $\Delta \alpha_{\chi} \Delta \beta \ \pi \Delta r$, 它们分别 表示垂直、水平和距离维度。假设雷达三维扫描的 范围是 α_{\min} 、 α_{\max} 、 β_{\min} 、 β_{\max} 、 r_{\min} 和 r_{\max} 、点云被分割 到体素为

$$H = \frac{\alpha_{\max} - \alpha_{\min}}{\Delta \alpha}, \ W = \frac{\beta_{\max} - \beta_{\min}}{\Delta \beta}, \ L = \frac{r_{\max} - r_{\min}}{\Delta r}$$
(3)

其中, H、W和L是体素总的个数。



图 10 体素分割示意图

(2)对于点云中的每个点,体素编号为

$$h = \left[\frac{\alpha_n - \alpha_{\min}}{\Delta \alpha}\right], \ w = \left[\frac{\beta_n - \beta_{\min}}{\Delta \beta}\right],$$
$$l = \left[\frac{r_n - r_{\min}}{\Delta r}\right]$$
(4)

其中, α_n 、 β_n 、 r_n 为点 p_n 在极坐标系下的坐标值,h、w、l为点 p_n 划分到体素的位置坐标。

在 3D 卷积中,每个体素包含的信息用一个向 量表示,此时需要将各个体素中包含的点云特征聚 合到这个向量。仿真环境采集的一帧点云的点个数 高达 10 万个,过多的点云数量聚集到体素中会带来 很多的计算量,因此在体素分割的基础上进行降采 样,被去除的点不参与后续的卷积网络计算,减少了 计算量。同时,为了减少降采样带来的物体几何信 息丢失,将点云的局部特征进行编码,聚合到体素的 向量中。点云的局部特征聚合的步骤如下 4 步所 示。

(1)一帧点云经过扇形体素分割并降采样后, 留下一个点作为该体素的代表。基于最近邻算法 (k-nearest neighbor, KNN),对每个非空体素中的点 p_n (查询点)查找它附近的 K 个最近邻点集 { p_n^1 , p_n^2 , …, p_n^k , …, p_n^K },其中 p_n^k 是距离 p_n 的第 k 个最 近点。然后为 p_n 最近的 K 个点进行相对位置编码 操作,其中第 k 个最近点特征信息表示为

个点的相对空间位置。

(2)经过上述步骤后,点的几何信息和相邻点 间的关系被编码到对应的查询点,特征 *F*_n 可以表示 为

$$\boldsymbol{F}_{n} = \{\boldsymbol{f}_{n}^{1}, \boldsymbol{f}_{n}^{2}, \cdots, \boldsymbol{f}_{n}^{k}, \cdots, \boldsymbol{f}_{n}^{K}\}$$
(6)

(3)采用多层感知机(multilayer perceptron, MLP) 来提取查询点几何的高维特征,对于K个最相邻点 的C个特征维度,基于最大池化层(max)保留特征 各个维度中最大特征值:

 $\hat{\boldsymbol{F}}_{n} = \max(MLP(\boldsymbol{F}_{n}))$ (7)

其中, \hat{F}_n 表示查询点的最显著特征。

(4)每个体素经过特征提取步骤得到高阶的、 带有最显著的局部几何特征。点云通过锥形体素集 合表示为 $\mathbb{R} \in C \times H \times W \times L$,其中C表示特征维 度,H、W和L分别是垂直、水平和距离维度。

考虑铁路场景是由轨道床、供电杆和线缆等组 成的固定结构。在列车转弯或变轨时,轨道在激光 雷达视角中不断变化,需要结合多层语义信息提取 特征。本文提出了基于稀疏 3D 卷积算子^[18]的多尺 度点云语义分割网络,网络的核心是将高维轨道环 境信息作为有效的先验条件,基于 mobilenet^[19]的可 分离卷积结构,构建轻量化的三维卷积升维下采样 模块和降采样模块,此外利用上采样和下采样模块 构建基于 U-Net^[20]语义分割的网络结构,最后通过 逻辑回归分类器给每个体素预测类别。



点云语义分割网络模型结构如图11所示,网络

采用编码和解码结构,由4个D模块组成,共经过3次下采样;解码部分由4个U模块组成,共经过3次上采样。编码部分属于收缩路径,用于获取上下文信息即局部特征,每一次下采样,卷积的感受域会扩大一倍;解码部分属于扩张路径,用于精确定位语义分割的位置,同时收缩路径和扩张路径是相互对称的。

网络模型中的 D 和 U 模块是可分离结构的三 维稀疏卷积模块。如图 12 所示,方块表示输入与输 出的体素特征向量,spconv 是稀疏 3D 卷积算子,deconv 是反卷积算子。D 模块的输入是低阶的体素特 征,经过一个[1×1×1, stride =1]的稀疏卷积核对 每个特征通道数进行卷积操作,再经过[3×3×3, stride =1]的深度稀疏卷积和实现特征融合,并将此 时的特征图传给上采样模块,最后通过[1×1×1, stride =2]对体素进行逐点卷积与下采样,得到高阶 小尺寸的输出特征图。U 模块的输入是低阶的体素 特征,经过一个[1×1×1, stride =2]的反卷积核对 特征图进行反卷积操作并恢复其尺寸,然后聚合与 U 模块对称结构的 D 模块的特征图,再进行深度可 分离稀疏卷积操作得到低阶的特征图。



图 12 下采样编码与上采样解码模块

在轨道点云分割的任务中,D和U模块基于深 度可分离结构,通过减少[3×3×3]卷积核中的通 道数核增加[1×1×1]卷积核的通道数,目的是在 保持模型性能(点云分割精度)的前提下降低模型 大小和计算量。以输入特征图大小 $C_{in} \times H \times W \times L$ 和输出特征图大小 $C_{out} \times H \times W \times L$ 为例,对于标准 卷积 $D \times D$,其计算量为 $C_{in} \times C_{out} \times H \times W \times L \times D$ ×D;而对于深度卷积[3×3×3],其计算量为 $C_{in} \times$ $H \times W \times L \times D \times D$,逐点卷积[1×1×1]的计算量为 $C_{in} \times C_{out} \times H \times W \times L$;标准卷积核深度可分离卷积 的计算量比值为

$$\frac{C_{\rm in} \times H \times W \times L \times D \times D + C_{\rm in} \times C_{\rm out} \times H \times W \times L}{C_{\rm in} \times C_{\rm out} \times H \times W \times L \times D \times D}$$
$$= \frac{1}{C_{\rm out}} + \frac{1}{D^2}$$
(8)

当输出特征图的通道数 C_{out} 很大时,深度可分离卷 积可以大约降低 D² 倍的计算量,同时网络模型的参数量也会降低。

算法1是本节方法的具体实现过程。

算法1 轨道点云分割算法
输入:点云 $P \in \mathbb{R}^{N \times 3}$;提速网格大小 $[\Delta \alpha, \Delta \beta, \Delta r]$
输出:点云标签 $L \in \mathbb{R}^{N \times 1}$ 和轨道点云 P^L ;
1. for p_n , $p_n \in P$ do
2. 计算 p_n 极坐标 $[\alpha_n, \beta_n, r_n]$; //式(2)
3. 计算 p_n 的体素编号 $[l_n, w_n, h_n]$; //式(4)
4. end for
5. 体素降采样 $P^{v} = unique(P, [h, w, l]);$
6. for p_n , $p_n \in P^V$ do
7. 最近邻点集 $\{p_n^k\} = KNN(p_n, P);$
8. 相对位置编码 $F_n \leftarrow \{f_n^k\}$; //式(5)、(6)
9. 高维最大特征提取 $\hat{F}_n \leftarrow \max(MLP(F_n));$
10. end for
11. 体素标签计算 $\boldsymbol{L}^{V} = network(\boldsymbol{P}^{V}, \hat{\boldsymbol{F}});$
12. for p_n , $p_n \in P$ do
13. 映射 p_n 标签 L_n ← map(L^v , $[l_n, w_n, h_n]$);
14. $L \leftarrow L_n$; // 点的标签添加到 L 中
15. if $L_n = 轨道$ then
16. $P^{R} \leftarrow p_{n}$; //得到轨道点云
17. end if
18. end for //得到点云标签 $L \in \mathbb{R}^{N \times 1}$

2.2 障碍物检测

激光雷达实时采集环境点云,轨道部分点云在 每帧的点云中都是近距离密集和远距离稀疏的特 点,轨道形状包括直轨道、弯轨道和直轨道与弯轨道 相接等。离散的三维轨道点云需要拟合出轨道中心 线,基于中心线推算出列车前方的安全行驶空间。

面对直轨道和弯轨道场景,在轨道线拟合时忽略z轴,将轨道点云看作二维平面上的点。用直线 表示轨道线是一种简化的方法,但是在弯轨道的近 似结果与真实曲线偏差较大。二次函数可以同时表 示直线与弯曲曲线,因此本文基于最小二乘法拟合 二次函数来表示轨道的中心线。根据列车安全行车 区域的宽和高,平移曲线得到三维矩形盒体范围,即 列车前方安全行车区域。

在铁路环境中,障碍物大多数是未知物体,落在 轨道上时有可能被识别为轨道,所以要分割出列车 行驶区域内的路面点云,才能进一步对障碍物点云 聚类以及定位。本文基于随机采样一致(random sample consensus,RANSAC)^[17]算法对安全区域内 的点云进行去地面处理。RANSAC 是一种在包含离 群点的数据集合中,通过迭代的方式估计模型的参 数,例如三维空间中的平面。相比于最小二乘法,在 密度不均匀的轨道点云中,基于 RANSAC 的平面拟 合可以缓解点云分布不均匀的问题。假设铁路的地 平面模型为α,本文采用基于 RANSAC 拟合平面方 法的步骤如下。

(1) 在点云 P 中随机选择 3 个不共线的点计算出平面 α, 其方程为

 $Ax + By + Cz + D = 0 \tag{9}$

其中, $A \ B \ C \ D$ 是 3D 空间中确定平面 α 的参数, $x \ y, z$ 表示三维坐标系下的值。

(2)依次计算 P 中所有点到平面 α 的距离,把 距离小于阈值 d 的点归为内点,得到内点集合 In, 记录集合 In 中点的数量 N。

(3)迭代步骤(1)和(2),对应于 N 最大的平面 方程即为最优模型;当满足设定条件时退出迭代,例 如达到设定的迭代次数。

在列车安全行驶的 3D 区域内,去除地面点云后,剩下的都属于地面以上的点,然后对这些点进行 聚类,形成多类物体的集合。依次统计集合中点的 数量,若数量未超过设定的阈值,则认为无障碍物, 反之则判定为存在障碍物。

基于密度的聚类算法(density-based spatial clustering of applications with noise, DBSCAN)^[21], 与层次 和划分聚类方法不同, 它将类定义为密度相连点的 最大集合, 能够把具有足够高密度的点云划分为同 一类, 并且能抑制离群点的影响。本节基于 DB-SCAN 算法对点云聚类, 如图 13 所示, 根据 ε (在一 — 1083 — 个点周围邻近区域的半径)、k(邻近区域内至少包 含点的个数)2个参数,把样本点云中的点分成3 种:(1)核心点(A点):以当前点为球心半径为 ε 的 领域内含有不小于 k 个点云。(2)边缘点(B、C 点):当前点云为球心半径为 ε 的领域内含有小于 k 且大于0个点。(3)离群点(D点):不是核点也不 是边缘点。



图 13 基于 DBSCAN 算法的点云聚类

通过 DBSCAN 聚类后,再统计每个聚类结果中 的点云,将点的数量超过设定阈值的类别判定为障 碍物,并将该类中点的最大和最小坐标值作为障碍 物的具体位置,完成列车安全行车区间内障碍物的 定位。

算法2是障碍物判定算法的具体实现过程。

算法2 轨道障碍物判定算法
输入: 点云标签 $L \in \mathbb{R}^{N \times 1}$ 和轨道点云 P^L , 列车宽 W 高
H;平面距离阈值d;点云数量阈值k;
输出:障碍物位置;
1. 拟合二次函数
$l: y = ax^{2} + bx + c \leftarrow fit(P^{R}:(x, y));$
2. 平移 l 得到行车区域 V
$y_1 = y + W, y_2 = y - W, y_3 = y_1 + H, y_4 = y_2 + H;$
3. for p_n , $p_n \in P$ do
4. if $p_n \in V$ then
5. $P^{s} \leftarrow p_{n}; // 将 p_{n}$ 加到行车区域点云 $P^{s};$

- 8. 拟合平面 $\alpha \leftarrow fit(P^s:(x, y, z));$ 9. for $p_n, p_n \in P^s$ do 10. if $| p_n, \alpha | > d$ then 11. $P^o \leftarrow p_n; // \Re p_n$ 加到障碍物点云 $P^o;$ 12. end if 13. end for 14. 点云聚类得到点云标签 $L^o:$ DBSCAN $(P^o);$ 15. for $i, i \in \max(L^o)$ do 16. if $| P^o[L^o = i] | > k$ then // 判定为障碍物 17. // 计算障碍物的位置
 - 18. $\min(P^{0}(x, y, z)), \max(P^{0}(x, y, z));$
 - 19. end if
 - 20. end for

6.

end if

7. end for

3 实验

本文的实验首先使用设计的虚拟激光雷达从铁路仿真环境中采集点三维云数据并制作数据集,然后训练点云语义分割网络模型并评估模型准确率。 将障碍物随机地放置在轨道附近重新采集点云数据,使用训练的模型分割出轨道点云,再拟合连续的 列车安全行车区域并得到区域内的点云。最后对安 全区域内的点云进行地面分割和障碍物点云聚类等 步骤实现障碍物检测。

3.1 铁路环境三维点云采集及数据集标注

在仿真环境中采集点云数据,如图 14 所示,3 种轨道场景包括直轨道、左转和右转轨道,其中纹状 线条表示雷达射线与物体的交点,这些交点的集合 即点云。

如图 15 所示,在轨道点云横截面中,轨道枕木 和铁轨组成的区域具有明显的几何特征。因此将这 个区域作为轨道标记区域,然后手动标注数据中的



图 14 直轨道、左转轨道和右转轨道的点云采集



图 15 轨道标记区域

轨道点云,其余的点云标注为背景,标注完成后的可 视化如图 16 所示,其中虚线框内的点表示标注为轨 道点云。

在仿真环境中,一共采集 180 帧点云数据并划 分训练集和测试集,统计信息如表 2 所示,包括 60 帧直线轨道、60 帧左转弯轨道和 60 帧右转弯轨道。 数据集包含大约 452 万个点,其中轨道大约 31.2 万 个点。因为场景的空旷程度不一致,点云帧中点的 数量范围是2万~5万,每个点包含x、y和z值,表



图 16 标注结果可视化图

	中午来午	点的数	$ \mathbb{E}(10^3) $	特决不可
切京	帜剱	全部	轨道	- 籶坦尖型
1	50	1257	65	直轨道
2	50	137	95	左转轨道
3	50	1151	101	右转轨道
4	10	226	13	直轨道
5	10	288	19	左转轨道
6	10	220	19	右转轨道

表 2 轨道环境仿真数据集具体信息

示物体在 3D 坐标系下被激光雷达扫描到的坐标 值。根据铁路场景和轨道的排布类型,分为 3 类场 景,分别是直轨道,左转轨道和右转轨道。场景 1、 2、3 作为训练集,共 150 帧;场景 4、5、6 作为测试 集,共 30 帧。

3.2 轨道点云分割实验

本实验的硬件设备是一个 6 核 3.4 GHz 的 AMD CPU 与一张 Nvidia 2080Ti 显卡。深度学习环 境由 Ubuntu-18.04 系统, Pytorch-1.6, Spconv-1.2.1 等组成。锥形体素分割、相对位置编码和模型训练 的主要参数如表 3 所示。为了防止模型参数过拟 合,在±5°范围内将点云随机绕z轴旋转,并且在点 云上加入高斯噪声。模型训练过程中,动态学习速 率设定为每10个迭代(epoch)减小10倍,并使用交 叉熵损失函数来计算网络输出与真值标签的误差。

表 3 网络模型训练的参数

参数	值
体素网格大小	[0.3°, 0.5°, 0.5 m]
MLP 结构	[13, 64, 64, 16]
K	4
初始学习率	0.001
迭代次数	20

实验结果评估方法使用通用的点云语义分割网络评价指标,分别是交并比(intersection over union, IoU),精度(precision),召回率(recall),表达式为

$$IoU = \frac{TP}{TP + FP + FN}$$

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$
(10)

其中,TP 表示真阳性,即正确识别轨道点的数量; FP 表示假阳性,即真实背景点被识别为轨道点的数 量;FN 表示假阴性,即真实轨道点被识别为背景点的数量。

模型首先经过 20 个 epoch 训练,达到收敛;然 后在场景 4、5、6 中测试,测试结果如表 4 所示,轨道 识别的平均 *IoU* 达到 93.2%。

场景	IoU/%	Precision/%	Recall/%
4	92.1	96.3	95.5
5	93.3	95.5	97.6
6	94.2	96.6	97.5
平均值	93.2	96.1	96.9

表 4 测试集上的轨道分割结果

轨道点云分割结果的可视化样例如图 17 所示, 其中虚线框内的点代表预测的 TP 点,圆形框内的 点代表 FN 点,方形框内的点代表 FP 点。无论是在 近距离密度高轨道还是在远距离点云稀疏的轨道, 除了边缘部分有少许识别错误的点,带状的轨道点 云基本上被完整分割。





在对比实验中,使用 2 个有代表性的语义分割 网络模型 PointNet 和 RandLA-Net 作为对比网络模型,在同样的数据集上训练和调参。对比实验的结 果如表 5 所示,本文提出的网络模型具有最高的 *IoU*,并且远高于其他 2 个对比网络。

3.3 障碍物检测实验

使用仿真环境中的障碍物元素,在轨道附近放

表 5 轨道点云分割对比实验结果

场景	IoU/%	时间/ms	显存占用/MB
PointNet	72.7	18.5	7548
RandLA-Net	84.3	35.8	1353
本文网络模型	93.2	59.2	1677

置1至2个障碍物,然后采集有障碍物的点云数据, 如图18所示。为了检验方法的鲁棒性能,在直轨 道、左转和右转轨道上分别采样20帧带有随机放置 障碍物的点云,在60帧点云中手动标注出障碍物的 三维位置。



图 18 放置圆球障碍物

在标注障碍物时,障碍物的 3D 目标框位置使 用一个六维向量表示为

 $[x_{l}, y_{l}, z_{l}, x_{r}, y_{r}, z_{r}]$ (11) 其中, $[x_{l}, y_{l}, z_{l}]$ 和 $[x_{r}, y_{r}, z_{r}]$ 表示目标框的左 下角坐标和右上角坐标,若存在多个障碍物则在标 签中有多个对应的位置向量。

将采集的数据输入到训练好的轨道点云分割网 络模型中,得到轨道点云。忽略z轴,使用点云的x 和y轴坐标值进行二次函数拟合,轨道中心线拟合 结果如图19所示。

仿真环境列车的宽和高分别为3m和4m,以人 与列车的安全距离1m为限制,列车的安全行车区 域宽的为5m,向四周平移轨道中心线得到连续的 3D区域,即列车前方安全行车区域,如图20所示, 分别是直轨道、左转和右转轨道,其中安全行车区域 由实线框表示,虚线框内的点表示区域内的点云。

安全行车区域内的点云通常包括轨道、部分轨



图 19 轨道中心线拟合



图 20 列车安全行驶区域

道床和障碍物等。基于 RANSAC 算法分离地面,它 的迭代次数设置为 1000,距离阈值 d 取为 20 cm,如 图 21 所示,在 3 种轨道场景中均达到较好的分离效 果,在地面点云,框内的点包括障碍物与离散的地面 点(未被正确分离的地面点云)。

安全区域内的点云分离地面后,会有未被分离 的地面点和噪声点,它们都是稀疏地分布在边缘位 置,而障碍物则呈现密集的点云。在 DBSCAN 对剩 余点云聚类时,将邻近区域的半径 *ε* 取值为 50 cm, 邻近区域最小点个数 *k* 取值为 10 时取得最佳效果 且有较好的鲁棒性,过大和过小的邻域半径和不合 适的最小包含点取值都会导致障碍物无法被精准地 聚类或识别失败。图 21 直轨道的聚类结果如图 22 所示,其中三角框内的是无效点,方形和椭圆形分别 为 2 类障碍物点云。







图 22 直轨道障碍物点云聚类结果

障碍物判定的点云数量阈值为 30,即同类点云 数量大于 30 则被判定为障碍物,然后计算该类所有 点的边界值得到 3D 物体框,完成障碍物检测。如 图 23 所示,分别是直轨道、左转轨道和右转轨道的 障碍物检测结果,其中灰色的是背景点,虚线框是列 车前方安全行驶区域,实线框内的点云分别是障碍 物。

本文实验使用的主机配置包括1块6核3.4 GHz 的 CPU,1 张 NVIDIA 2080TI 显卡。图 24 是点的数 量与算法耗时的关系图,仿真环境中一帧点云包含 2.5~5.5 万个点,算法耗时与点的数量呈正比关 系,耗时大约在70~90 ms/帧。雷达采集点云的频 率为10 Hz,即100 ms/帧,本文算法平均耗时小于雷 达采集速度,因此达到实时性。

4 结论

针对铁路场景列车运行的安全性问题,本文设 计了一种基于虚幻引擎与三维点云的列车前方障碍 物检测仿真系统。首先,本文构建的铁路环境仿真 系统和虚拟激光雷达传感器弥补了真实场景采集障



图 23 直轨道、左转轨道和右转轨道障碍物检测结果



图 24 算法耗时

碍物数据困难的缺陷。然后,轨道分割方法中的锥 形体素点云分割方法实现了体素随距离增大而包含 更多的点,缓解了点云密度对局部特征聚合的影响。 最后,提出了一种基于三维点云数据的列车前方障 碍物判定方法用于检测障碍物。实验结果表明,点 云语义分割网络在仿真环境采集的数据中分割轨道 点云的平均 *IoU* 达到 93.2%,拟合的列车安全行驶 区域与铁轨方向一致,障碍物判定方法可以准确地 识别障碍物的位置。与传统轨道障碍物检测系统相 比,本系统可以高效地构建环境、采集数据和验证方 法的有效性。 参考文献

- [1] 铁道技术监督. 国家铁路局发布《2020 年铁道统计公报》[J]. 铁道技术监督, 2021,49(5):34.
- [2] SELVER A M, ATAÇ E, BELENLIOGLU B, et al. Visual and LIDAR data processing and fusion as an element of real time big data analysis for rail vehicle driver support systems [J]. Innovative Applications of Big Data in The Railway Industry, 2018(1):40-66.
- [3] KOH K H C. Developing an integrated rail simulator for SimMobility[D]. Massachusetts: Massachusetts Institute of Technology, 2017.
- [4] 郭彦宏. 基于城轨交通三维仿真系统的列车自动驾驶 研究[D]. 成都:西南交通大学, 2018.
- [5] TANG S. Rail platform obstacle detection using LabVIEW simulation [D]. Gävle: Sweden University of Gävle, 2015.
- [6] NASSU B T, UKAI M. A vision-based approach for rail extraction and its application in a camera pan-tilt control system[J]. IEEE Transactions on Intelligent Transportation Systems, 2012,13(4):1763-1771.
- [7] MANIKANDAN R, BALASUBRAMANIAN M, PALANIVEL S. Vision based obstacle detection on railway track [J]. International Journal of Pure and Applied Mathematics, 2017, 116:567-576.
- [8]周杏芳.基于多源图像融合的铁路异物侵限检测方法 研究[D].北京:北京交通大学,2018.
- [9] 郭双全. 基于雷达测量数据的列车运行前方障碍物检测方法研究[D]. 兰州:兰州交通大学, 2020.

- [10] 蔚增岩. 基于激光测量技术的铁轨障碍物检测系统研 究[D]. 长春:长春理工大学, 2020.
- [11] 杜华臻,张文安,杨旭升.基于激光雷达的移动机器 人人体目标跟随 [J].高技术通讯,2019,29(12): 1239-1246.
- [12] ZHU X X, TUIA D, MOU L, et al. Deep learning in remote sensing: a comprehensive review and list of resources[J]. IEEE Geoscience and Remote Sensing Magazine, 2017,5(4):8-36.
- [13] QI C R, SU H, MO K, et al. Pointnet: deep learning on point sets for 3d classification and segmentation [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017:652-660.
- [14] HU Q, YANG B, XIEL, et al. Randla-net: efficient semantic segmentation of large-scale point clouds [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 11108-11117.
- [15] MATURANA D, SCHERER S. Voxnet: a 3d convolutional neural network for real-time object recognition [C] //2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Hamburg: IEEE, 2015: 922-928.
- [16] ZHU X, ZHOU H, WANG T, et al. Cylindrical and asymmetrical 3d convolution networks for lidar segmenta-

tion[C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Virtual: IEEE, 2021:9939-9948.

- [17] FISCHLER M A, BOLLES R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography[J]. Communications of the ACM, 1981,24(6):381-395.
- [18] GRAHAM B, ENGELCKE M, VAN DER MAATENL. 3D semantic segmentation with submanifold sparse convolutional networks [C] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 9224-9232.
- [19] HOWARD A G, ZHU M, CHEN B, et al. Mobilenets: Efficient convolutional neural networks for mobile vision applications [EB/OL]. (2017-04-17) [2022-1-20]. http://arxiv.org/pdf/1704.04861.pdf.
- [20] RONNEBERGER O, FISCHER P, BROX T. U-net: Convolutional networks for biomedical image segmentation [C]//International Conference on Medical Image Computing and Computer-Assisted Intervention. Cham: Springer, 2015: 234-241.
- [21] ESTER M, KRIEGEL H P, SANDERJ, et al. A densitybased algorithm for discovering clusters in large spatial databases with noise[C] // Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining. Portland: AAAI Press, 1996: 226-231.

Design of railway obstacle detection simulation system based on unreal engine and 3D point cloud

YU Xinyi, HE Weiqi, CUI Zhufan, OU Linlin

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023)

Abstract

With the development of railway technology, active protection technology for train safety is extremely important in train running, such as a non-contact obstacle detection system. However, obstacle scenarios are difficult to be captured in a real train-running railroad environment. There exists some difficulty in collecting experimental data, such as image and point cloud data. In this paper, a railway simulation system and a virtual lidar are designed to provide a reliable test environment and point cloud acquisition sensors for railway non-contact obstacle detection research. A point cloud semantic segmentation network is proposed to accurately identify the rail located in front of the train based on voxel partition and sparse convolution. A continuous safe driving area is obtained by fitting discrete rail point clouds based on the least squares algorithm. A method for calculating the 3D position of obstacles is presented based on plane segmentation and point cloud clustering. By collecting the point clouds of three railway scenes in the simulation environment, the datasets are made for training the network model. Furthermore, point clouds with obstacle scenes are collected to test the obstacle detection method. The experimental results verify the feasibility of the simulation system and the effectiveness of the proposed obstacle detection method.

Key words: unreal engine, railway simulation, 3D point cloud, obstacle detection, neural network