第 48 卷 第 17 期/2021 年 9 月/中国激光 研究论文

中国激光

基于语义分割的深度学习激光点云三维目标检测

赵亮[1,2,3,4]，胡杰[1,2,3,4*]，刘汉[1,2,3,4]，安永鹏[1,2,3,4]，熊宗权[1,2,3,4]，王宇[1,2,3,4]

[1]武汉理工大学汽车工程学院，湖北 武汉 430070；

[2]武汉理工大学现代汽车零部件技术湖北省重点实验室，湖北 武汉 430070；

[3]武汉理工大学汽车零部件技术湖北省协同创新中心，湖北 武汉 430070；

[4]武汉理工大学湖北省新能源与智能网联车工程技术研究中心，湖北 武汉 430070

摘要 最远点采样(FPS)算法可用于三维(3D)目标检测算法中关键点的采集，针对 FPS 采集的关键点中前景比例较低的问题，提出了一种基于语义分割特征的区域卷积神经网络(Seg-RCNN)3D 目标检测算法。用一种基于语义分割特征的最远点采样(SegFPS)算法预测点的语义分割类别，以提高采集关键点中前景点的比例，从而提升 Seg-RCNN 算法的检测精度。该算法以激光点云作为输入，在第一阶段，利用 3D 稀疏卷积网络和 2D 卷积生成候选框和前景点分割网络(SegNet)，得到每个点的分割类别；在第二阶段，基于 SegNet 输出的分割类别用 SegFPS 算法从原始点云中采集一小部分关键点，从而在降低算法时间复杂度和空间复杂度的同时保留一定比例的前景点和背景点。在 KITTI 测试集上的测试结果表明，相比现有的主流算法，Seg-RCNN 算法的检测精度高、运行时间短，对中等级、容易等级 Car 类的 3D 检测精度分别为 79.73%、89.16%，运算时间仅需 80 ms。此外，基于机器人操作系统实现了算法的在线检测，验证了算法的工程实用性。

关键词 遥感；自动驾驶；激光雷达；三维目标检测；深度学习

中图分类号 TN958.98；TN249 **文献标志码** A **doi**：10.3788/CJL202148.1710004

1 引 言

随着自动驾驶及机器人领域对环境感知需求的提高，三维(3D)目标检测逐渐受到了人们的关注。激光雷达(LiDAR)可将外界环境的 3D 空间信息记录为点云(点的集合)，从而为 3D 目标感知和场景理解提供重要的空间信息，因此，激光雷达在自动驾驶及机器人等领域中得到了广泛应用。

目前，基于深度学习理论的激光点云 3D 目标检测算法可大致分为两类，一类是基于体素(Voxel-based)的特征提取算法，另一类是基于原始点云(Point-based)的特征提取算法。基于体素的算法一般将非结构化数据的点云转化为 3D 体素[1-5]或转化到二维(2D)俯视图网格上[6-10]，通常用 3D 稀疏卷积或 2D 卷积提取特征，计算效率较高，但点云的体素化及 3D 稀疏卷积的降采样会损失原始点云的 3D 空间信息，导致基于体素的算法检测精度有所下降。基于原始点云的算法利用点云网络(PointNet)[11]或其变体[12]直接从原始点云中提取特征，未对点云进行体素化及卷积降采样，因此，保留了原始点云的高精度 3D 空间位置信息[12]；且基于原始点云的算法通过设置特征提取层(Point set abstraction)的聚合半径可以灵活设置感受野，从而实现较高的检测精度。基于原始点云的算法首先利用最远点采样(FPS)对点云提取一部分关键点(Keypoint)，但这种仅基于欧氏距离的采样算法会使有些目标框(Box)内的点(前景点)在采样过程中全部丢失。而前景点包含准确检测目标位置的重要信息，若采样后的关键点中前景点的比例较低，则会大大降低算法的检测精度。现有算法使用 FPS 时通常会采集多个关键点，以弥补 FPS 没有区别对待前景点和背景点的缺陷，但这并没有解决根本问题。因

收稿日期：2021-01-11；**修回日期**：2021-02-17；**录用日期**：2021-03-09

基金项目：湖北省技术创新专项(2019AEA169)、湖北省科技重大专项(2020AAA001)

通信作者：*auto_hj@163.com

此，为了在 FPS 中得到更多的前景点，需要考虑除欧氏距离外的其他采样标准。语义分割特征是指原始点云中每个点的类别，其中，前景点（Foreground point）定义为在标注框内部的点，其他点则被称为背景点（Background point）。需要注意的是，通常不区分是车辆的前景点还是行人或骑手的前景点，所有类别 3D 标注框内的点都被统称为前景点。

由于前景点能提供 3D 目标检测算法需要的目标位置信息，因此，本文提出了一种基于语义分割特征的最远点采样（SegFPS）算法，相比现有的 FPS 算法，可最大程度地保留关键点中的前景点。为了在 FPS 中高效得到每个点的分割类别（前景点或背景点），SegFPS 算法设计了一个语义分割网络 SegNet。SegNet 由数层一维（1D）卷积网络组成，可接收来自第一层 3D 稀疏卷积的输出，并利用 1D 卷积根据语义分割标签对点云进行监督学习。将点是否在 3D 真实标注框内作为判断依据，从而得到每个点的语义分割类别标签（前景点和背景点）。实验结果表明，该算法能解决现有 FPS 中前景点比例较低的问题，相比当前的主流算法，目标检测精度更高。

2 相关研究

2.1 基于体素算法的 3D 目标检测

激光点云数据不同于图像的 3 通道二维矩阵，它是杂乱无章的点的集合[13]。因此，人们提出了多种将无规律点云数据转化成类似图像矩阵数据的算法[14-22]，以适应卷积神经网络（CNN）。其中，大多数算法是将点云分配到 3D 空间网格或 2D 俯视图网格中，并用 3D 或 2D CNN 进行处理，这类算法也被统称为基于体素的算法。如文献[6]中提出的 MV3D（Multi-view 3D object detection network）算法将点云投影到 2D 俯视图网格中，并利用预定义的锚框（Anchor box）生成检测框。关于投影到俯视图 2D 网格的研究，文献[7,9]提出了更优的多传感器融合策略，文献[8,10]提出了基于俯视图的点云表示方法，文献[4,23]提出的负载可卸除网络也达到了较好的分割效果。此外，文献[2,24]将点云分配到 3D 网格中，并用 3D CNN 进行处理；文献[3,25]用一种高效的稀疏 3D CNN 对点云进行处理；文献[26-27]采用了多头检测（Detection head）机制，得到了较好的检测结果。上述基于体素的算法效率较高且能产生精准的 3D 候选框，但其感受野受 CNN 中卷积核大小的限制，不能充分利用空间信息。不同 FPS 算法的对比情况如图 1 所示，文中相关算法的命名规则如表 1 所示。

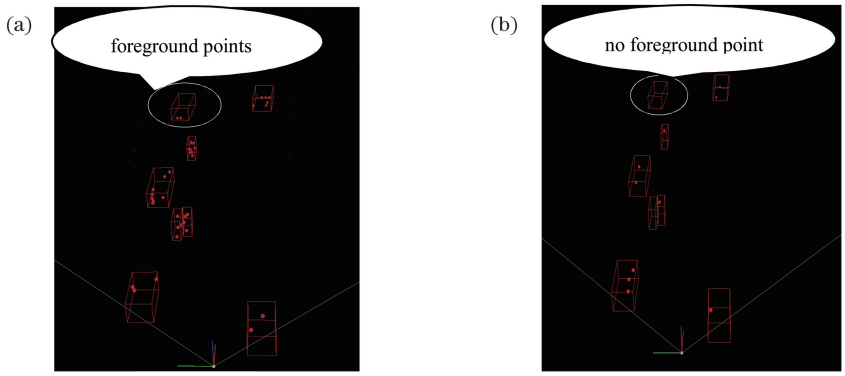

图 1 不同 FPS 算法的对比。(a)SegFPS 算法；(b)传统 FPS 算法

Fig. 1 Comparison of different FPS algorithms. (a) SegFPS algorithm; (b) traditional FPS algorithm

表 1 命名规则

Table 1 Nomenclature

Abbreviation	Explanation
Seg-RCNN	segmentation based region-convolution neural networks
SegFPS	segmentation classes based further point sampling
SegNet	semantic segmentation network for foreground points
NMS	non-maximum-suppression
Grouping	using keypoints to group features
Bev	bird's eye view
FPS	further point sampling

2.2 基于原始点云算法的 3D 目标检测

F-PointNet(Frustum PointNet)[28]将 PointNet[11-12]应用于 3D 目标检测中，该算法将基于图像检测的 2D 检测框通过几何约束投影到 3D 点云中对点云进行裁剪，并用 PointNet 处理裁剪的点云，得到检测结果。刘训华等[16]对基于原始点云的算法进行了改进，但这类算法受图像 2D 预检测精度的限制，不能充分利用点云的 3D 空间信息。PointRCNN[29]直接将原始点云作为输入进行目标检测，STD(Sparse-to-dense)[30]在第二阶段使用体素化算法对 Box 进行细化回归。将原始点云作为网络输入的算法通过设置聚合半径可得到灵活的感受野，且没有经过体素化和 3D 卷积降采样，产生的候选框(Box proposals)有着较高的召回率(Recall)。但相比基于体素的算法，基于原始点云的算法在推理速度上略慢，而基于体素的 RCNN(PV-RCNN)[31]通过融合这两种算法实现了高效和精准的目标检测。

3 算法框架

Seg-RCNN 为基于点云的双阶段(Two-stage)3D 目标检测算法，其结构如图 2 所示。其中，FC 为全连接层，Conv 为卷积。第一阶段对原始点云进行体素化，利用 3D 稀疏 CNN 提取体素特征，将 3D 体素特征压缩为 2D 俯视图特征，利用 2D 卷积在 2D 俯视特征图上生成候选框及其类别，与此同时，SegNet 检测每个点的分割类别；第二阶段中，SegFPS 根据 SegNet 检测的点类别对原始点云进行采样，得到关键点，利用 PointNet[11]中的聚合操作将多尺度的体素特征聚合到关键点上形成关键点特征。将第一阶段生成的候选框映射到关键点聚合的特征上，以提取感兴趣区域(ROI)特征，并利用该特征细化和筛选候选框，进而输出最终的检测结果。在 SegFPS 算法采集关键点的过程中结合分割特征着重采集了前景点，以提升关键点中前景点的比例。基于 PointNet 将多层不同尺度的 3D 体素特征进行聚合(Grouping)，如图 3 所示，Grouping 中的箭头表示聚合的特征向量。聚合后可得到关键点的聚合特征(Key-features)，实现特征压缩。多层不同尺度的 3D 体素特征可保留原本的空间坐标值，但尺度会发生变化，因此，需要将不同尺度的体素特征根据 3D 稀疏卷积降采样的倍数缩放到原始点云的尺度。与图像领域的 2D 目标检测不同，3D 目标检测是对物体空间位置的检测，在识别目标位置的同时也对目标进行了分类，即对目标同时进行定位与分类。此外，还增加了绕 z 轴的旋转角度，以区别于无旋转角度的 2D 图像目标检测。

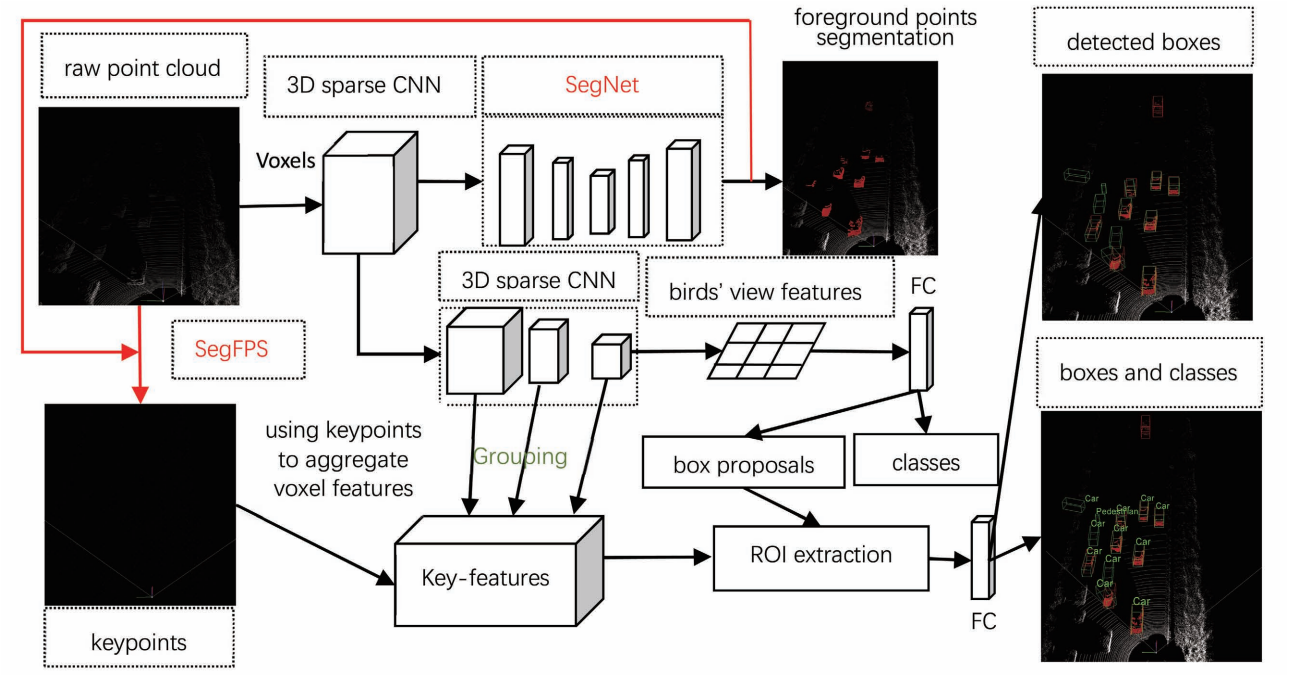

图 2 Seg-RCNN 的框架

Fig. 2 Framework of the Seg-RCNN

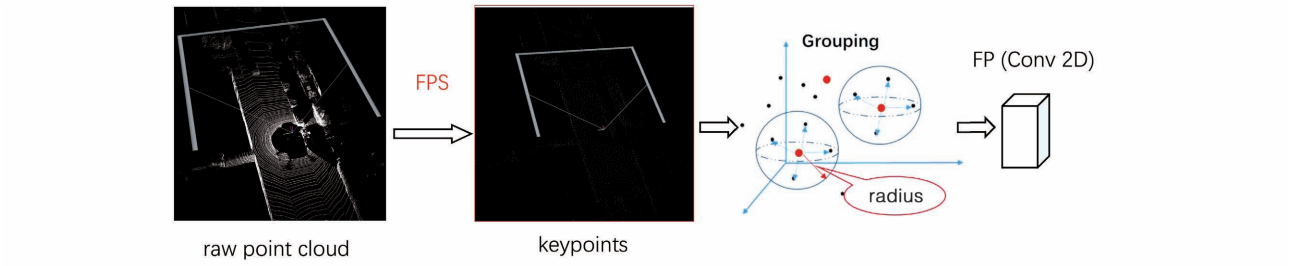

图 3 基于原始点云算法的网络结构

Fig. 3 Network structure based on original point cloud algorithm

3.1 第一阶段

目前，主流的 3D 目标检测算法[1,3,31-32]大都采用 3D 稀疏 CNN(3D sparse CNN)进行特征提取，原因是该算法的运行效率高且能生成比较准确的候选框。

3.1.1 体素化及 3D 稀疏 CNN

基于体素的算法需要将点云划分到 3D 空间网格中，以实现数据的结构化。若输入点云(所有点的集合)为 P，P 中的每个点将被划分到一些小的体素网格中，这些空间体素网格的分辨率为 $L\times W\times H$(L、W、H 分别为体素网格长、宽、高方向的网格数量)。通过计算非空体素网格内点的均值得到该体素网格的特征(Voxel wise feature)，从而将逐点特征(Point wise feature)转化为逐体素特征。逐点特征通常为该点的三维坐标 x、y、z 以及反射强度 i。该体素网格使用一系列卷积核为 3×3×3 的 3D 稀疏 CNN，逐层将点云的体素空间网格降采样为原始点云空间网格的 1，1/2，1/4，1/8 倍。中间体素特征在第二阶段被关键点聚合，以进行候选框的细化回归。

3.1.2 3D 候选框的生成

首先，将降采样 8 倍的 3D 体素特征压缩为俯视图 2D 特征，再利用 2D CNN 结合基于锚(Anchor-based)的策略[3,10]得到比较精准的候选框(3D box proposals)。俯视特征图的长、宽分别为 $L/8$、$W/8$，对于俯视特征图(Bev. feature map)，每个位置都有两个互相垂直的锚框，即每一类待识别的目标都有 $2\times(L/8)\times(W/8)$ 个预定义的固定尺寸锚框。在候选框生成网络(RPN)中，相比 PointNet[29-30]，3D 体素稀疏卷积的召回率更高[31]。

3.1.3 前景点分割网络

SegNet 可预测点云中每个点的语义类别(前景点与背景点两类)，其由一系列 1D 卷积组成，如图 4 所示。其中，C 为卷积层的通道数，DeConv 为反卷积，输出层(Out Conv.)的卷积核为 1，其余网络层的卷积核为 3，第一层 3D 稀疏卷积的输出为 SegNet 的输入。与单分支网络不同，SegNet 可提取多尺度中间特征进行监督学习，监督学习的标签为点的语义分割类别。通过判断每个点是否在 3D 标注框内，就能得到点的语义分割类别标签。SegNet 为现有的 3D 目标检测提供了一个新思路，即充分利用语义分割特征可进一步提升目标检测的精度。

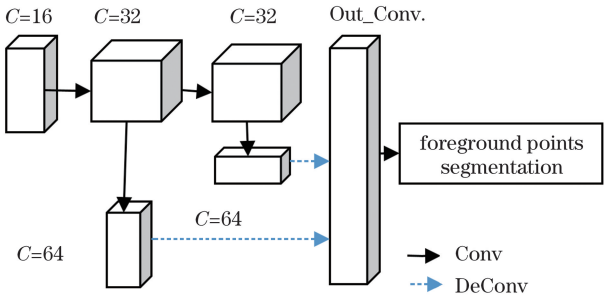

图 4 SegNet 的结构

Fig. 4 Structure of the SegNet

3.2 第二阶段

FPS 采集得到的关键点中前景点的比例较低，部分目标框内的前景点会在 FPS 过程中全部丢失，进而影响算法的检测精度。因此，提出了一种结合语义分割特征的 SegFPS 算法。

3.2.1 SegFPS 算法

SegFPS 算法可从原始点云中采集一小部分关键点，与 PointNet 中的 FPS 不同，SegFPS 结合了每个点的分割类别，给予前景点更大的权重，提升了关键点中前景点的比例。SegFPS 的目的是从原始点云集合 P 中采集出 n 个关键点 $K=\{p_1,\cdots,p_n\}$，具体步骤如下。

1) 随机在点云 P 中选择出第一个点 p_1，计算点云 P 中所有点到 p_1 点的欧氏距离，记与 p_1 点距离最大的点为 p_2，并将 p_1 和 p_2 作为初始的关键点集 K。

2) 计算集合 P 中除 p_1 和 p_2 外其余点到关键点集 K 的距离(K 中只包含 p_1 和 p_2)，点云 P 中

的任意一个点 p_m 到关键点集 K 的距离 D_{p_m-K} 定义为点 p_m 到 K 集合中所有点距离的最小值。

3) 将点云 P 内每个点的 D_{p_m-K} 根据 SegNet 输出的分割类别进行加权处理，权重值为 0～1 之间的归一化值。越靠近 1，表明该点为前景点的可能性越大；越靠近 0，表明该点为背景点的可能性越大。如 SegNet 输出某个点的分割结果是 One-hot 类型的值[b，f]，其中，b 处于索引为 0 的位置，即背景点；f 处于索引为 1 的位置，即前景点。对比 b 与 f 的大小，就能知道该点为前景点还是背景点。若 f 较大，表明该点为前景点；反之，则表明该点为背景点。但 SegNet 输出的 b 和 f 并不是一个归一化值，还需用 Sigmoid 函数将其缩放至[0，1]之间。设权重 b_w 为一个较小值，加权后前景点的 D_{p_m-K} 较大，背景点的 D_{p_m-K} 较小。因此，根据每个点的 D_{p_m-K} 判断该点是否为最远点采样时，前景点被采样的几率更大。

4) 在点云 P 中选择与关键点集 K 距离 D_{p_m-K} 最大的点作为本次迭代得到的关键点 p_3，并将其添加到关键点集 K 中，此时，$K=\{p_1, p_2, p_3\}$。

5) 重复计算点云 P 中剩余点到点集 K 的距离，并根据语义分割类别调整权重，根据加权后的 D_{p_m-K} 将选出的最远点 $p_{\max}$ 添加到关键点集 K 中，当关键点集 K 中的元素达到预设值时，停止迭代。

SegFPS 算法使用 CUDA (Compute unified device architecture)以及 C++实现并行计算，运行效率较高，SegFPS 算法的计算时间约为 0.001 s。SegFPS 被封装为 Python 可以调用的模块，从而实现接口的统一。

3.2.2 基于关键点聚合的多尺度体素特征

基于原始点云的算法一般包含 3 步：1) FPS 操作；2) 聚合操作；3) 特征传播(FP)操作。其中，关键点的数量可人为设置，聚合操作是根据聚合半径聚合关键点周围的原始点云，即计算以关键点为球心，聚合半径内所有原始点到关键点的相对距离，并将该相对距离作为每个关键点聚合的特征。FP 操作利用 2D 卷积对关键点聚合的特征进行传播。Seg-RCNN 算法基于原始点云将多尺度的中间体素特征聚合到采集的关键点上，实现了特征压缩，得到关键点聚合的体素特征。

3.2.3 感兴趣区域池化

ROI 池化先将候选框划分为 6×6×6 的网格，然后将每个候选框中的 216 个网格中心点作为 ROI 的关键点(ROI-grid)。该关键点基于 PointNet 提取关键点特征，从而学习 ROI 的特征，实现对候选框的二次细化回归。

3.3 训练的损失函数

Seg-RCNN 的损失函数由 RPN 的损失 L_{RPN}、SegNet 的类别损失 L_{Seg} 和 ROI 的细化回归损失 L_{ROI} 组成。L_{RPN} 损失与文献[3]一致，可表示为

$$L_{\text{RPN}}=L_{\text{cls}}+\beta\sum_{r\in\{x,y,z,l,h,w,\theta\}}L_{\text{smooth-l1}}(\Delta\hat{r}^{\text{a}},\Delta r^{\text{a}}),\tag{1}$$

式中，L_{cls} 为目标框的类别损失，可由 Focal loss[33] 函数计算，其超参数为文献[33]的默认值。Δr^{a} 为真实框(GT-box)与锚框(Anchor box)的残差，w、l、h 分别为检测框的长、宽、高，θ 为目标框绕 z 轴的旋转角度，$\Delta\hat{r}^{\text{a}}$ 为预测的残差，$L_{\text{smooth-l1}}$ 为候选框位置及尺寸的损失函数。SegNet 的类别损失 L_{Seg} 也采用 Focal loss[33] 函数，L_{ROI} 由目标框的置信度 $L_{\text{confidence}}$ 和候选框的位置损失组成，可表示为

$$L_{\text{ROI}}=L_{\text{confidence}}+\sum_{r\in\{x,y,z,l,h,w,\theta\}}L_{\text{smooth-l1}}(\Delta\hat{r}^{\text{p}},\Delta r^{\text{p}}),\tag{2}$$

式中，$L_{\text{confidence}}$ 可由交叉熵计算得到，其监督学习的目标是候选框与真实 Box 之间的 3D 交并比(3D IOU)。Δr^{p} 为候选框和标签 Box 编码得到的残差，编码方式与 RPN 中基于 Anchor 策略的编码方式相同，$\Delta\hat{r}^{\text{p}}$ 为预测的残差。本算法的总损失函数可表示为

$$L_{\text{loss}}=L_{\text{RPN}}+L_{\text{Seg}}+L_{\text{ROI}}。\tag{3}$$

4 实验结果及分析

4.1 数据集

实验采用的数据集为 KITTI 数据集[34]，该数据集是目前公认的公开数据集之一。KITTI 数据集包含 7481 帧训练数据(train split 和 val split)和 7518 帧测试数据(test split)。实验参考文献[2]的方式将 7481 帧训练数据分为 3712 个训练分集(train split)和 3769 个验证分集(val split)，并将 Seg-RCNN 算法与现有算法在 KITTI 在线排行榜的 test split 上进行了对比。

4.2 Seg-RCNN 架构的参数

SegNet 由两部分 1D 卷积组成，将两部分 1D 卷积的结果进行拼接可得到多尺度感受野的特征。

图 2 中 3D 体素 CNN 的通道数分别为 16,32,64,64,关键点用 PointNet 聚合体素特征,三个聚合网络对应的聚合半径(Radius)分别为(0.4,0.8),(0.8,1.2),(1.2,2.4),其中,(A,B)为两次不同的聚合半径。在 ROI 池化中将每个候选框区域划分为 6×6×6 个网格,候选框网格点用 PointNet 聚合多尺度体素特征,其聚合半径为(0.8,1.6)。在 KITTI 数据集中的检测范围参考文献[2]设置:沿 x 方向为[0 m,70.4 m];沿 y 方向为[−40 m,40 m];沿 z 方向为[−1 m,3 m];体素化网格的大小参考文献[4]设置为[0.05 m,0.05 m,0.1 m].

4.3 训练细节

Seg-RCNN 以端到端的训练方式进行,使用的优化器为 Adam,训练在 4 个 2080Ti 显卡上进行,batch-size 为 8,学习率为 0.01,整个数据集迭代 80 轮(80 epochs),在 train split 上的训练过程约为 8 h。在第二阶段进行候选框的细化回归时,选取了 128 个候选框,其中,前景候选框与背景候选框的比例为 1∶1。前景候选框的选择标准是候选框和真实目标框的 3D IOU 大于 0.55。训练阶段算法采用的数据增强技术:1)绕 x 轴的对称处理;2)整体缩放(缩放因子为 0.95~1.05)处理;3)全局点云绕 z 轴旋转(旋转角度为[−π/4,π/4])处理;4)将其他帧包含目标的点云拷贝到当前帧,以模拟不同的场景;5)使用路面标注将所有标签的真实框放置在同一地平面高度上。在推理阶段,根据每个候选框的 IOU 置信度选择出前 100 个候选框。候选框的置信度为 3D IOU 值,若置信度大于 0.7,表明检测结果有效。针对选出的有效候选框进行第二阶段的细化回归,并用非极大值抑制(NMS)去除冗余的候选框,从而得到最终的检测结果,其中,NMS 操作中的 3D IOU 阈值为 0.01。

4.4 KITTI 数据集的 3D 目标检测排行榜

在 val split 数据上测试时,算法使用 train split 进行训练;在 KITTI 官方测试数据 test split 上测试时,则使用整个 train+val 数据进行训练。为了保证实验的公平性,这两种测试方式与对比文献完全一致。

4.4.1 评价指标

实验使用的评价指标为平均预测精度(mAP),在 KITTI 官方测试集 test split 上,mAP 为 40 个不同召回率位置上的平均精度,val split 验证集的 mAP 则为 11 个不同召回率位置的平均精度。

4.4.2 与现有算法的对比

表 2 为 Seg-RCNN 算法与现有算法在 KITTI 数据集上的检测结果,其中,Bev. 为俯视图下的 2D 框检测精度,Easy、Mod、Hard 表示目标框的检测难度为容易、中等、困难,RGB(Red,Green,Blue)为图像数据,LiDAR 为点云数据。可以发现,相比现有算法,Seg-RCNN 算法的检测精度较高。如对于 Car 类目标的 3D 检测中,Seg-RCNN 算法在 Easy、Mod、Hard 难度障碍物上的 mAP 至少提高了 3.22,3.97,3.29 个百分点,在 Car 类的俯视图检测中,Seg-RCNN 在 Mod 难度障碍物上的 mAP 至少提高了 2.00 个百分点。此外,Seg-RCNN 算法可以同时检测 Car、Pedestrian 和 Cyclist 三类目标,而文献[3, 10]中的算法对每个类的目标需要单独的模型。不同算法在 val split 上的测试结果如表 3 所示,可以发现,Seg-RCNN 算法的性能优于现有的主流算法。

表 2 不同算法在 KITTI 测试集上的 mAP

Table 2 mAP of different algorithms on the KITTI test set unit: %

Algorithm	Reference	Type	Car-3D			Car-Bev.			Cyclist-3D			Pedestrian-3D		
			Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard	Easy	Mod	Hard
MV3D[6]	CVPR 2017	RGB+LiDAR	74.97	63.63	54.00	86.62	78.93	69.80	–	–	–	–	–	–
F-PointNet[28]	CVPR 2018	RGB+LiDAR	82.19	69.79	60.59	91.17	84.67	74.77	72.27	56.12	49.01	–	–	–
ContFuse[9]	ECCV 2018	RGB+LiDAR	83.68	68.78	61.67	**94.07**	85.35	75.88				–	–	–
AVOD-FPN[7]	IROS 2018	RGB+LiDAR	83.07	71.76	65.73	90.99	84.82	79.62	63.76	50.55	44.93	–	–	–
PointRCNN[29]	CVPR2019	LiDAR	85.94	75.76	68.32	92.13	87.39	82.72	73.93	59.60	53.59			
SECOND[3]	Sensors 2018	LiDAR	83.34	72.55	65.82	89.39	83.77	78.59	71.33	52.08	45.83	–	–	–
PointPillars[10]	CVPR 2019	LiDAR	82.58	74.31	68.99	90.07	86.56	**82.81**	**77.10**	58.65	51.92	–	–	–
VoxelNet[2]	arXiv 2017	LiDAR	77.47	65.11	57.73	–	–	–	61.22	48.36	44.37	–	–	–
Ours	–	LiDAR	**89.16**	**79.73**	**72.28**	93.36	**89.39**	81.93	76.23	**60.05**	**54.37**	78.17	63.89	56.73

表 3　不同算法在 val split 上的 mAP

Table 3　mAP of different algorithms on the val split　　unit：%

Algorithm	Reference	Type	Mod	Easy	Hard
MV3D	CVPR 2017	RGB+LiDAR	62.68	–	–
ContFuse	ECCV 2018	RGB+LiDAR	73.25	–	–
F-PointNet	CVPR 2018	RGB+LiDAR	70.92	–	–
AVOD-FPN[7]	IROS 2018	RGB+LiDAR	74.44	–	–
PointRCNN[29]	CVPR 2019	LiDAR	78.63	–	–
STD[30]	ICCV 2019	LiDAR	79.80	–	–
Ours	–	LiDAR	81.11	91.33	77.49

为了更直观地显示本算法的检测效果，展示了部分 val split 上的检测结果，如图 5 所示，其中，视角为俯视，白色点为背景点，可以发现，本算法能精准检测出障碍物。

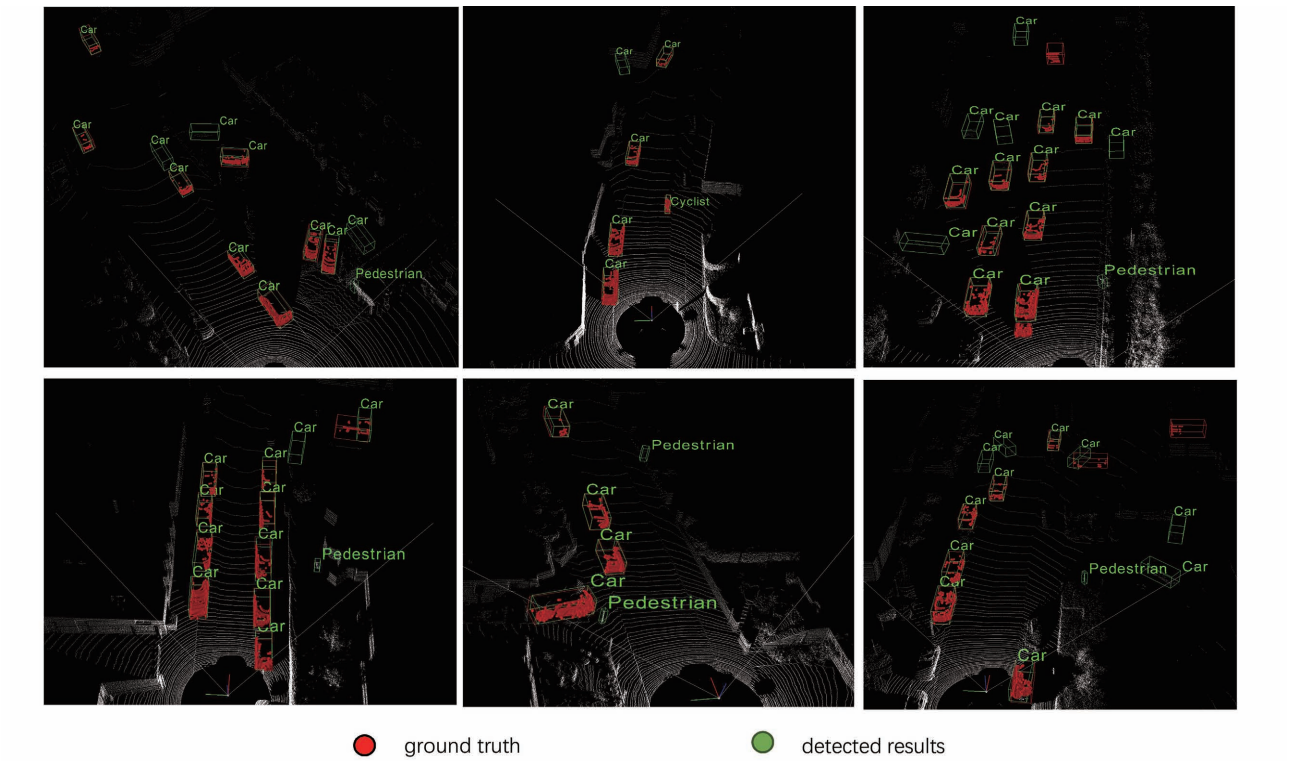

图 5　本算法在 val split 上的可视化检测结果

Fig. 5　Visual detection results of our algorithm on the val split

4.5　子模块分析

为了验证本算法中 SegNet 及 SegFPS 的有效性，进行了子模块测试实验。测试实验均在 train split 上进行训练，在 val split 上进行验证。评判标准参考文献[3，26]，即以 Car 类 Mod 难度级别的 3D mAP 作为评判标准。

4.5.1　SegNet 的不同策略

前景点的输出选择有两种，一种是输出两类（One-hot 类型），按照较大值确定前景点或背景点，将该策略称为 SegNet 1；另一种是将 SegNet 的输出设置为 1D 数据，经过 Sigmoid 函数处理后，按照预先设定的前景点阈值（Foreground threshold）判断该点是前景点还是背景点，将该策略称为 SegNet 2。不同策略的验证结果如表 4 所示，可以发现，SegNet 1 表现出的性能更好，原因是 SegNet 2 中使用手动阈值，难以达到最优效果。

表 4　不同策略 SegNet 的 3D mAP

Table 4　3D mAP of SegNet with different strategies

SegNet 1	SegNet 2	3D mAP/ %
√		78.23
	√	81.11

4.5.2　SegFPS 的有效性分析

FPS 是基于欧氏距离的最远点采样[11-12]，当选取的关键点较少时，会丢失某些障碍物的全部前景点。因此，本算法用 SegFPS 解决该问题，在 KITTI val split 上的测试结果如表 5 所示。可以发现，FPS 算法在测试集中的 3D mAP 为 78.21%，而 SegFPS 算法在测试集中的 3D mAP 为 81.11%，这表明 SegFPS 可以有效提高算法的检测精度。

表 5　SegFPS 的有效性验证

Table 5　Validation of the SegFPS

FPS	SegFPS and FPS fusion sampling strategy	SegFPS	3D mAP /%
√			78.21
	√		79.01
		√	81.11

4.6　检测精度分析

4.6.1　KITTI 数据集中漏标注对算法检测精度的影响

将 KITTI 数据集按照相机的视角标注，即只有图像范围内的障碍物标签，而处于相机盲区边缘的障碍物可能未被标注，如图 6 所示。可以发现，本算法能识别出未被标注的目标框。观察点云发现，这个目标是真实存在的，算法可以识别这部分被截断且处于盲区边缘的障碍物，但由于数据集未标注该目标，会被评价指标判为误检测，导致算法的检测精度较低。由于识别出盲区目标的算法得分较低，未识别出盲区目标的算法检测精度得分反而较高，对模型的选择会产生误导，这表明是否准确进行数据标注以及是否存在漏标注对算法精度的影响较大。

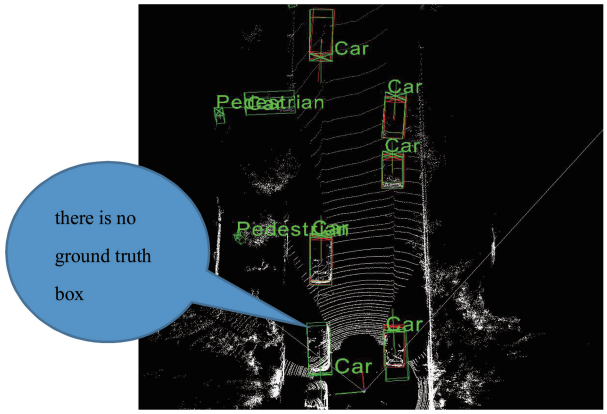

图 6　KITTI 数据集中没有标签的目标

Fig. 6　Unlabeled targets in the KITTI dataset

4.6.2　行人类检测精度的分析

站立人体的点云形状与树杆、电线杆相似，因此，本算法容易将树杆误检测为行人，如图 7 所示。评价指标会将这类误识别计算在内，导致 Pedestrian 类别的检测精度较低，而 Car 类和 Cyclist 类形状特殊、与其形状相似的物体也较少，因此，本算法对这类目标的检测精度较高。

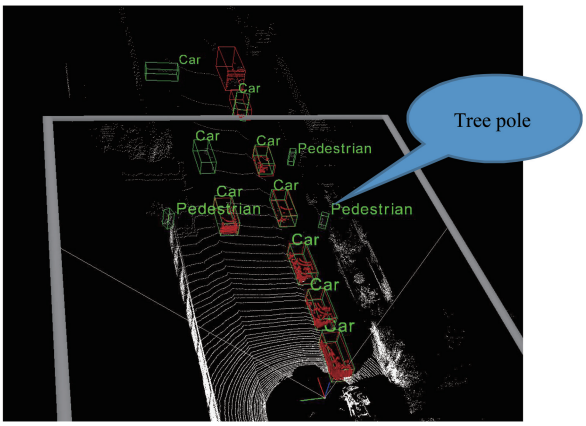

图 7　Pedestrian 类别的检测结果

Fig. 7　Detected result of the Pedestrian category

4.7　算法的在线检测

利用机器人操作系统(ROS)实现算法的在线检测，测试结果表明，本算法可以稳定识别场景中的障碍物。图 8 为 ROS 的节点分布，包含自动驾驶中常用的传感器数据，如图像、激光点云和毫米波雷达。

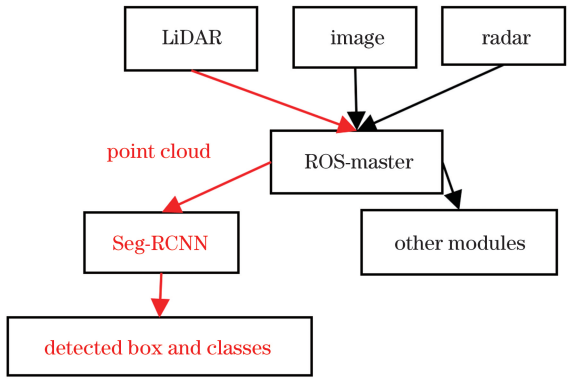

图 8　基于 ROS 实现的 Seg-RCNN 在线检测

Fig. 8　Seg-RCNN online detection based on ROS

4.8　算法的运行时间

Seg-RCNN 算法中 Point-based 部分的运行时间如表 6 所示，可以发现，聚合操作的耗时较长，为 32.7 ms；SegFPS 操作的耗时最短，从数量为 37595 的点云中采集 4096 个关键点仅需 0.141 ms；FP 操作也相对高效，运行时间为 5.2 ms。Point-based 部分的网络耗时共计约 38 ms，其中，SegFPS 只操作了 1 次，Grouping 和 FP 操作均有 6 次，在聚合三个稀疏卷积块时设置了两个不同的聚合半径，以提取不同感受野的特征。

表 6 本算法中 Point-based 部分的运行时间

Table 6 Running time of the Point-based part of our algorithm

Name of operation	SegFPS	Grouping	FP
Number of operation	1	6	6
Running time /ms	0.141	32.7	5.2

Voxel-based 算法的核心思想是将点云从非结构化数据转化为结构化数据，类似图像的 RGB 三通道矩阵，Voxel-based 思想的本质就是图像二维像素的拓展，即三维像素——体素（Voxel）。第一步是将点云分配到空间网格中，如图 9 所示，但每个网格中往往不止分配一个点，因此需要第二步的处理，即将逐点特征转化为逐体素特征。VFE（Voxel features encoder）可以将逐点特征转化为逐体素特征，已有研究表明[1]，直接对网格内点的坐标进行平均，将网格内点的均值作为该网格的体素特征也可以得到较好的结果，因此，均值法是将逐点特征转化为逐体素特征的有效算法之一。

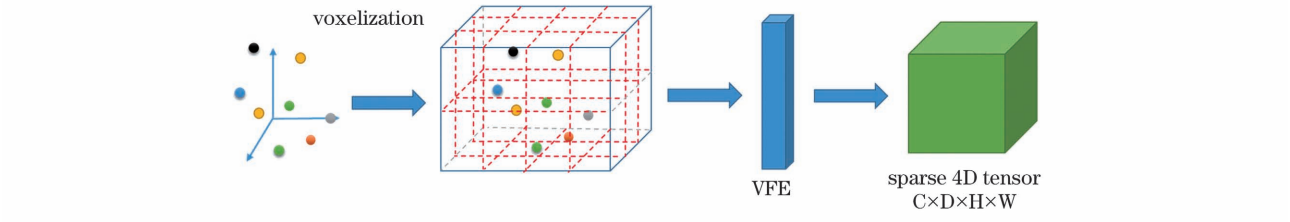

图 9 Voxel-based 算法的原理

Fig. 9 Principle of the Voxel-based algorithm

Voxel-based 算法部分的稀疏卷积耗时如表 7 所示，其中，中括号中的数字为该稀疏卷积块内稀疏卷积网络的层数，每个卷积层后都有批归一化（BN）及线性整流函数（ReLU）。可以发现，4 块稀疏卷积（Conv 1～Conv 4）总共消耗约 26.5 ms，这表明 Voxel-based 算法比 Point-based 算法更高效。

表 7 Voxel-based 算法的运行时间

Table 7 Running time of the Voxel-based algorithm

Conv	Running time /ms
Conv 1 (16,16) [1]	1.12
Conv 2 (16,32) [3]	6.40
Conv 3 (32,64) [3]	8.62
Conv 4 (64,128) [3]	10.40

SegNet 包含 8 层神经网络（6 层 1D 卷积、2 层 1D 反卷积），其中，1D 卷积运行十分高效，在 SegNet 部分的耗时约 1.24 ms。SegFPS 使用 CUDA 进行了并行加速，耗时约 1 ms。NMS 操作的运行时间为 6.5 ms，本算法各模块的运行时间如表 8 所示，图 10 为本算法在 KITTI 验证集上的运行时间，可以发现，本算法的平均运行时间约为 80 ms。

表 8 Seg-RCNN 中各个模块的运行时间

Table 8 Running time of each module in Seg-RCNN

Module	Running time /ms
Voxel-based	26.54
Point-based	38.04
SegNet	1.24
NMS	6.50
Others data transfer	～8

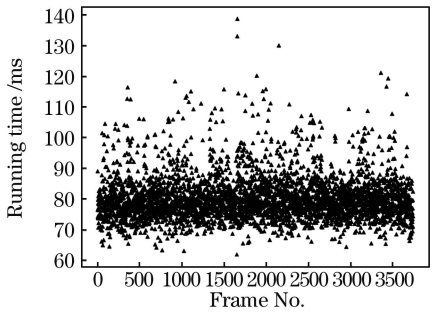

图 10 本算法在 val split 上的运行时间

Fig. 10 Running time of our algorithm on the val split

5 结 论

针对自动驾驶场景下激光点云的感知需求，提出了一种基于语义分割特征的激光雷达 3D 目标检测算法 Seg-RCNN。在 KITTI 官方测试集的测试结果表明，本算法在 Car 类目标简单等级上的检测精度可达到 89.16%。其中，SegFPS 算法在采集关键点时结合了语义分割类别，提高了关键点中前景点的比例。此外，数据标注的质量会影响目标检测的精度，如相机视角边缘的漏标注；行人点云形状与树干点云形状相似，也会导致一定的误检测，降低该类别的检测精度。最后，基于 ROS 实现了 Seg-RCNN 算法的在线检测，验证了该算法的工程应用价值。

参 考 文 献

[1] Shi S S, Wang Z, Shi J P, et al. From points to parts: 3D object detection from point cloud with part-

aware and part-aggregation network [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(8): 2647-2664.

[2] Zhou Y, Tuzel O. VoxelNet: end-to-end learning for point cloud based 3D object detection [C] // 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 4490-4499.

[3] Yan Y, Mao Y X, Li B. Second: sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10): 3337.

[4] He C H, Zeng H, Huang J Q, et al. Structure aware single-stage 3D object detection from point cloud [C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 11870-11879.

[5] Chen Y L, Liu S, Shen X Y, et al. Fast point R-CNN[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV), October 27-November 2, 2019, Seoul, Korea (South). New York: IEEE Press, 2019: 9774-9783.

[6] Chen X Z, Ma H M, Wan J, et al. Multi-view 3D object detection network for autonomous driving[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 6526-6534.

[7] Ku J, Mozifian M, Lee J, et al. Joint 3D proposal generation and object detection from view aggregation [C] // 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), October 1-5, 2018, Madrid, Spain. New York: IEEE Press, 2018: 1-8.

[8] Yang B, Luo W J, Urtasun R. PIXOR: real-time 3D object detection from point clouds[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 7652-7660.

[9] Liang M, Yang B, Chen Y, et al. Multi-task multi-sensor fusion for 3D object detection[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE Press, 2019: 7337-7345.

[10] 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, 2019: 12689-12697.

[11] 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.

[12] Qi C R, Yi L, Su H, et al. PointNet++: deep hierarchical feature learning on point sets in a metric space[EB/OL]. (2017-06-07)[2021-01-05]. https://arxiv.org/abs/1706.02413.

[13] Yang B, Liang M, Urtasun R. HDNET: exploiting HD maps for 3D object detection[EB/OL]. (2020-12-21)[2021-01-05]. https://arxiv.org/abs/2012.11704.

[14] Liu Z J, Tang H T, Lin Y J, et al. Point-voxel CNN for efficient 3D deep learning[EB/OL]. (2019-07-08)[2021-01-05]. https://arxiv.org/abs/1907.03739.

[15] Yi H W, Shi S S, Ding M Y, et al. SegVoxelNet: exploring semantic context and depth-aware features for 3D vehicle detection from point cloud[C]//2020 IEEE International Conference on Robotics and Automation (ICRA), May 31-August 31, 2020, Paris, France. New York: IEEE Press, 2020: 2274-2280.

[16] Liu X H, Sun S Y, Gu L P, et al. 3D object detection based on improved Frustum PointNet[J]. Laser & Optoelectronics Progress, 2020, 57(20): 201508.
刘训华, 孙韶媛, 顾立鹏, 等. 基于改进 Frustum PointNet 的 3D 目标检测[J]. 激光与光电子学进展, 2020, 57(20): 201508.

[17] Lei X D, Wang H T, Zhao Z Z. Small sample airborne LiDAR point cloud classification based on transfer learning [J]. Chinese Journal of Lasers, 2020, 47(11): 1110002.
雷相达, 王宏涛, 赵宗泽. 基于迁移学习的小样本机载激光雷达点云分类[J]. 中国激光, 2020, 47(11): 1110002.

[18] Huang G, Liu X L. Automatic extraction and classification of road markings based on deep learning [J]. Chinese Journal of Lasers, 2019, 46(8): 0804002.
黄刚, 刘先林. 基于深度学习的道路标线自动提取与分类算法[J]. 中国激光, 2019, 46(8): 0804002.

[19] Ma J J, Pan Q, Liang Y, et al. Object detection based on improved Grassberger entropy random forest classifier[J]. Chinese Journal of Lasers, 2019, 46(7): 0704011.
马娟娟, 潘泉, 梁彦, 等. 基于改进 Grassberger 熵随机森林分类器的目标检测[J]. 中国激光, 2019, 46

(7): 0704011.

[20] Zhang A W, Liu L L, Zhang X Z. Multi-feature 3D road point cloud semantic segmentation method based on convolutional neural network[J]. Chinese Journal of Lasers, 2020, 47(4): 0410001.
张爱武, 刘路路, 张希珍. 道路三维点云多特征卷积神经网络语义分割算法[J]. 中国激光, 2020, 47(4): 0410001.

[21] Lu X Y, Yun T, Xue L F, et al. Effective feature extraction and identification method based on tree laser point cloud[J]. Chinese Journal of Lasers, 2019, 46(5): 0510002.
卢晓艺, 云挺, 薛联凤, 等. 基于树木激光点云的有效特征抽取与识别算法[J]. 中国激光, 2019, 46(5): 0510002.

[22] Gu S T, Wang L, Ma Y X, et al. Local feature description of LiDAR point cloud data based on hierarchical Mercator projection[J]. Acta Optica Sinica, 2020, 40(20): 2015001.
顾尚泰, 王玲, 马燕新, 等. 基于分层墨卡托投影的激光雷达点云数据局部特征描述[J]. 光学学报, 2020, 40(20): 2015001.

[23] Du L, Ye X Q, Tan X, et al. Associate-3Ddet: perceptual-to-conceptual association for 3D point cloud object detection[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 13326-13335.

[24] Song S R, Xiao J X. Deep sliding shapes for amodal 3D object detection in RGB-D images[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 808-816.

[25] Graham B, Engelcke M, Maaten L V D. 3D semantic segmentation with submanifold sparse convolutional networks [C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 9224-9232.

[26] Kuang H W, Wang B, An J P, et al. Voxel-FPN: multi-scale voxel feature aggregation for 3D object detection from LiDAR point clouds[J]. Sensors, 2020, 20(3): 704.

[27] Zhu B J, Jiang Z K, Zhou X X, et al. Class-balanced grouping and sampling for point cloud 3D object detection[EB/OL]. (2019-08-26)[2021-01-05]. https://arxiv.org/abs/1908.09492.

[28] Qi C R, Liu W, Wu C X, et al. Frustum PointNets for 3D object detection from RGB-D data[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 918-927.

[29] Shi S S, Wang X G, Li H S. PointRCNN: 3D object proposal generation and detection from point cloud [C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE Press, 2019: 770-779.

[30] Yang Z T, Sun Y N, Liu S, et al. STD: sparse-to-dense 3D object detector for point cloud[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV), October 27-November 2, 2019, Seoul, Korea (South). New York: IEEE Press, 2019: 1951-1960.

[31] Shi S S, Guo C X, Jiang L, et al. PV-RCNN: point-voxel feature set abstraction for 3D object detection [C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 10526-10535.

[32] Graham B, van der Maaten L. Submanifold sparse convolutional networks [EB/OL]. (2017-06-05) [2021-01-05]. https://arxiv.org/abs/1706.01307.

[33] Lin T Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 42(2): 318-327.

[34] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite[C]//2012 IEEE Conference on Computer Vision and Pattern Recognition, June 16-21, 2012, Providence, RI, USA. New York: IEEE Press, 2012: 3354-3361.

Deep Learning Based on Semantic Segmentation for Three-Dimensional Object Detection from Point Clouds

Zhao Liang[1,2,3,4], Hu Jie[1,2,3,4*], Liu Han[1,2,3,4], An Yongpeng[1,2,3,4], Xiong Zongquan[1,2,3,4], Wang Yu[1,2,3,4]

[1] *School of Automotive Engineering, Wuhan University of Technology, Wuhan, Hubei* 430070, *China*;

[2] *Hubei Key Laboratory of Advanced Technology for Automotive Components, Wuhan University of Technology, Wuhan, Hubei* 430070, *China*;

[3] *Hubei Collaborative Innovation Center for Automotive Components Technology, Wuhan University of Technology, Wuhan, Hubei* 430070, *China*;

[4] *Hubei Research Center for New Energy & Intelligent Connected Vehicle, Wuhan University of Technology, Wuhan, Hubei* 430070, *China*

Abstract

Objective The low detection accuracy of the perception system in autonomous vehicles will seriously affect the reliability of autonomous vehicle and the safety of passengers. The traditional LiDAR-based three-dimensional (3D) object detection algorithms, such as the rule-based clustering method highly relies on hand-designed features probably be sub-optimal. Following the great advantages in deep learning for image field, a large body of literature to explore the application of this technology for 3D LiDAR point clouds. Among them, point-based methods directly use raw point clouds as the input of the detection model, and the further point sampling(FPS) algorithm is applied to sample a set of keypoints from raw point clouds, keypoints groups neighbor raw points to extract the feature for object detection. However, the proportion of foreground points (points in 3D bounding box) in keypoints collected through FPS algorithm are relatively low, especially for the remote object, foreground points almost totally lost in FPS (Fig. 1). Foreground points contain the important 3D space location information of objects, a low proportion of foreground points in keypoints will hurt the detection accuracy. To this end, we propose a semantic segmentation based two-stage 3D object detection algorithm named Seg-RCNN (segmentation based region-convolution neural networks), in which we propose a novel further point sampling strategy (segFPS) for sampling keypoints, and a segmentation network (SegNet) for semantic segmentation of foreground points and background points (Fig. 4).

Methods Seg-RCNN is a two-stage 3D object detector (Fig. 2), in the first stage, the raw points are first voxelized as voxel-wise features, then the sparse 3D CNN is adopted to extract voxel features, the output of sparse 3D CNN squeeze into 2D CNN for further feature propagation, and then box proposals are generated in 2D bird's eye view feature map through anchor-based strategy. The SegNet output the foreground and background points segmentation results of point clouds. In the second stage, the SegFPS is adopted to sample the keypoints according to the segmentation results obtained from SegNet. SegFPS, distinguished with previous FPS, uses both segmentation classes (foreground points and background points) and Euclidean distance as sampling criteria, which can improve the proportion of foreground points in keypoints (Fig. 1), and then can improve the detection accuracy by 2.90 percentage points in the KITTI dataset. Using keypoints to represent the whole point clouds not only reduces the complexity in time and space but also retains a certain proportion of foreground points and background points. Multi-scales 3D voxel features of different layers are aggregated into a set of keypoints through PointNet backbone to obtain the features aggregated by the keypoints (named key-features), so that achieve feature compression (grouping, as shown in Fig. 3, calculates the relative distance between the keypoints and neighbor raw points). Then, 2D CNN is adopted to further propagate the key-features. Project the box proposals onto the key-features map to extract the region of interests, finally the detection heads output the final perception results.

Results and Discussions Extensive experiments on KITTI dataset are conducted to demonstrate the higher performance of our framework as compared with previous mainstream methods, and the detection accuracy of the car class in moderate and easy level are 79.73%, 89.16%, respectively(Table 2). The mean average precision (mAP) of Seg-RCNN, on car objects easy, moderate and hard levels, increased by at least 3.22, 3.97 and 3.29 percentage points, respectively. There are two output strategies in SegNet, experiment results suggest that SegNet 1 is better

than SegNet 2 (Table 4). Adopting SegFPS to sample the keypoints indeed can improve the detection accuracy by 2.90 percentage points as compared with FPS (Table 5). The accuracy of annotation in dataset affect the detection performance of algorithm (Fig. 6), since the correct detection box will be judged as false positive due to the missing of data annotation. The similarity of the shape between different objects, such as the shape of tree pole and telephone pole are very similar to the pedestrian in point clouds, would decrease the classification accuracy, and then decrease the detection accuracy(Fig. 7). Runtime of the proposed method is 80 ms (Table 8). To further facilities the application of engineering, we achieve online real-time detection through robot operating system, which has great values for engineering projects.

Conclusions In this paper, we consider the problem of low proportion of foreground points in keypoints after FPS sampling, and introduce Seg-RCNN, a novel 3D object detection algorithm that has potential values of application for autonomous vehicle projects. Extensive experiments on KITTI dataset are conducted, the experiment results suggest that our algorithm has higher detection accuracy as compared with previous mainstream methods, specifically, the mAP of car class on easy, moderate and hard levels increase to at least 3.22, 3.97 and 3.29 percentage points, respectively. The runtime of our algorithm is only 80 ms. Our results suggest that Seg-RCNN is an effective architecture for 3D object detection on point clouds.

Key words remote sensing; autonomous vehicle; LiDAR; three-dimensional object detection; deep learning

OCIS codes 280.3640; 100.4996; 150.1135; 150.6910