

基于 ResNet-MLP 模型的车辆目标检测算法

王可栋^{1,2a}, 曲含章³, 马敏^{4,5,*}, 杨子奕^{2b}, 康爱平^{2a,6}

(1. 青岛黄海学院 智能制造学院, 青岛 266427; 2. 青岛理工大学 a. 机械与汽车工程学院; b. 土木工程学院, 青岛 266525;
3. 长安大学 长安都柏林国际交通学院, 西安 710064; 4. 青岛环球重工科技有限公司, 青岛 266400;
5. 山东科技大学 能源与矿业工程学院, 青岛 266590; 6. 兰州现代职业学院 财经商贸学院 兰州 730300)

摘要:为提高车辆目标检测精度, 针对传统视觉传感器光照敏感性、空间感知性差等缺点, 采用激光雷达传感器, 提出一种基于 ResNet-MLP 二阶段模型的车辆目标检测算法。该算法对点云鸟瞰图的映射方式进行改进, 使其保留点云高度特征, 并通过改进后的 ResNet 进行点云特征的提取, 最后使用并行多层感知机网络对车辆目标分类和位置回归。采用 KITTI 的 3D Object 数据集进行验证, 通过与 PointNet++ 和 VoxelNet 方法进行对比实验, 结果发现, 交并比 (IOU) 较高时 3 种方法的检测精度均有所下降, 但相对于其他 2 种算法, 本算法检测精度更高, 运行速度更快, 可为未来自动驾驶车辆的实时感知方面提供技术支撑。

关键词: 车辆目标检测; 激光雷达; 点云图像; 鸟瞰图映射

中图分类号: U491.1 文献标志码: A 文章编号: 1673-4602(2024)03-0142-09

Vehicle target detection algorithm based on ResNet-MLP modeling

WANG Kedong^{1,2a}, QU Hanzhang³, MA Min^{4,5,*}, YANG Ziyi^{2b}, KANG Aiping^{2a,6}

(1. Intelligent Manufacturing Institute, Qingdao Huanghai University, Qingdao 266427, China;
2. a. School of Mechanical and Automotive Engineering; b. School of Civil Engineering, Qingdao University of Technology, Qingdao 266525, China; 3. Chang'an Dublin International College of Transportation, Chang'an University, Xi'an 710064, China; 4. Qingdao Hicorp Heavy Industry Science & Technology Co. Ltd., Qingdao 266400, China; 5. College of Energy and Mining Engineering, Shandong University of Science and Technology, Qingdao 266590, China; 6. College of Business and Economics, Lanzhou Modern Vocational College, Lanzhou 730300, China)

Abstract: Aiming at the disadvantages of low sensitivity and poor spatial perception of traditional visual sensors, a vehicle target detection algorithm based on ResNet-MLP two-stage model is proposed to enhance the accuracy of vehicle target detection by using LiDAR sensor. The algorithm improves the mapping of point cloud bird's-eye view so that it retains the point cloud height features, and the point cloud features are extracted by the improved ResNet. Finally, a parallel multilayer perceptron network is used to classify vehicle targets and regress the position. KITTI's 3D Object dataset is used for validation, and through comparison experiments with PointNet++ and VoxelNet methods, it is found that the detection accuracy of the three methods decreases when the IOU is higher, but compared to the other

收稿日期: 2023-12-28

基金项目: 国家自然科学基金资助项目 (52272311)

作者简介: 王可栋 (1986—), 男, 山东潍坊人。博士, 副教授, 主要从事人机共驾理论及技术方面的研究。

E-mail: 18661831717@163.com

* 通信作者: 马敏 (1985—), 女, 山东济宁人。硕士, 高级工程师, 主要从事机电工程、建筑交通方向的研究。

E-mail: 18663981858@163.com

two algorithms, the proposed algorithm has better detection accuracy and higher running speed, which can provide technical support for the real-time perception of self-driving vehicles in the future.

Key words: vehicle target detection; LiDAR; point cloud images; bird's-eye view mapping

作为自动驾驶和智能交通研究的一项关键技术,车辆目标检测正受到越来越多的关注^[1]。现有检测算法主要依靠视觉传感器(如高清摄像头),但是视觉传感器由于其工作原理,缺乏三维空间感知能力,且易受到光照和天气情况的影响,鲁棒性差^[2]。近年来,具备远距离深度感知能力的激光雷达传感器^[3](Light Detection and Ranging, LiDAR)逐渐受到研究人员的关注。LiDAR 是一种主动视觉传感器,基本不受光照及天气情况的影响。它发射的激光脉冲在物体上反射,记录反射的光能(即物体的反射强度),同时根据飞行时间原理计算其与物体的距离,使其能够感知周围环境的结构。点云是 LiDAR 输出数据的代表,类似于图像是相机输出数据的代表。点云是大量点的集合,描述了 LiDAR 周围环境的 3D 结构^[4]。

基于点云数据的特点,国内外大量学者将 LiDAR 应用在三维目标检测上,提出诸多目标检测算法^[5]。传统的三维目标检测算法需要先经过背景过滤、点云聚类之后,根据目标的几何特征对其进行识别和分类,从而完成三维目标检测^[6]。这类传统算法的运行时间长,无法满足目标检测实时性要求,而且精度不高^[7]。随着计算机硬件的发展,大部分学者开始考虑使用深度学习的方法进行点云目标检测。QI 等^[8]于 2017 年首次提出了 PointNet 网络,利用深度学习技术检测点云中的 3D 目标。该网络处理点云分类任务,其基本思路是直接处理原始点云,利用 MLP 和 MaxPooling 提取点特征和全局对象,但这种方法在处理稀疏点云数据时效率相对较低。进一步提出了 PointNet 的改进版,即 PointNet++,解决了 PointNet 不能捕获点云局部特征的问题^[9]。

PointNet++ 的主要思想是通过聚类创建若干个点集(候选区域),使用 PointNet 在每个点集中提取点特征,并分层多次重复这一过程。每次聚类算法输出的点集在后续过程中被视为抽象点集,由此提取的点特征具有更广泛的领域含义,包括局部邻域内的上下文信息。这种方法忽略了点云空间的空区域,避免了无效计算,也不需要调整量化超参数,但它没有使用先进的基于二维卷积的目标识别算法,同时用 GPU 进行点云处理的效率较低,实际执行速度不高。为了加快检测速度,PIXOR 网络将三维点云投影到地面,转为鸟瞰图(Bird's-eye View, BEV),然后使用二维卷积网络提取 BEV 特征,进行目标检测^[10]。该方法虽然采取了二维卷积网络提升了算法的运行效率,但是将三维点云转为 BEV 过程中丢失了高度信息,导致其目标检测准确率较低。

为了同时兼顾运行效率和检测准确率,提出一个两阶段检测模型架构(ResNet-MLP)用以检测道路使用者。该架构对 BEV 映射方式进行改进,使其保留点云高度信息,经过卷积和最大池化后将点云输入到 ResNet 网络中进行特征提取,最后通过并行 MLP 网络对车辆目标进行分类,并回归出目标位置参数(目标中心点、目标长宽高、目标航向角)。

1 LiDAR 工作原理及点云预处理

1.1 LiDAR 工作原理

LiDAR 主要由激光、扫描仪和 GPS 接收器组成。而激光又由红外激光发射器和配对的红外激光接收器组成,安置在坚硬的外壳内。在 LiDAR 接通电源后,激光扫描仪开始 360°旋转扫描其周边环境。激光扫描仪发射出的脉冲在空气中传播,直至碰到反射能量的障碍物,反射的能量被配对的红外激光接收器捕获,生成相关的 3D 点云数据和反射强度数据等^[11]。其中 3D 点云数据由 d 、 ω 、 α 坐标构成,是激光束扫描到的每个点相对于 LiDAR 的球形坐标系坐标。在点云目标检测算法中,常用的坐标系为笛卡尔坐标系 x 、 y 、 z ,因此,需要首先将球形坐标系下的点云转换为笛卡尔坐标系下的点云(图 1),转换公式如式(1)所示。

$$\begin{cases} x = d \cdot \cos\omega \cdot \sin\alpha \\ y = d \cdot \cos\omega \cdot \cos\alpha \\ z = d \cdot \sin\omega \end{cases} \quad (1)$$

在 LiDAR 扫描过程中,决定云数据质量的因素为 LiDAR 本身的内置特征,主要有:通道数、最大检测范围、垂直视场、垂直视场分辨率、水平视场和水平视场分辨率^[12]。通常, LiDAR 通道数越多,可以收集到的点云就越密集。LiDAR 内置特征可以由图 2 进行解释。

1.2 点云数据预处理

在 LiDAR 采集数据的过程中,由于受到周围环境和测量设备的影响,其采集到的点云数据会引入噪声点^[13],进而影响算法的精度。为了解决这一问题,采用均值滤波法对原始点云进行降噪。假设 P 为一给定点,其邻域为 $\{P_{i,j}=(x_{i,j},y_{i,j},z_{i,j})|-n \leq i \leq n, -m \leq j \leq m\}$ 。其中, i, j 分别为给定点的左右邻域; $(x_{i,j},y_{i,j},z_{i,j})$ 为给定点邻域的空间三维坐标; n, m 分别为左右邻域阈值。那么,经均值滤波后, P 点变为

$$\bar{P}_{i,j} = \sum_{i=-n}^n \sum_{j=-m}^m h(P_{i,j}) \quad (2)$$

式中: h 为规范化加权系数。

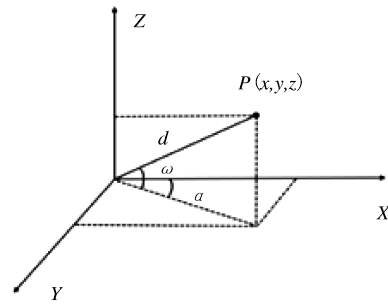


图1 球形坐标系转为笛卡尔空间坐标系

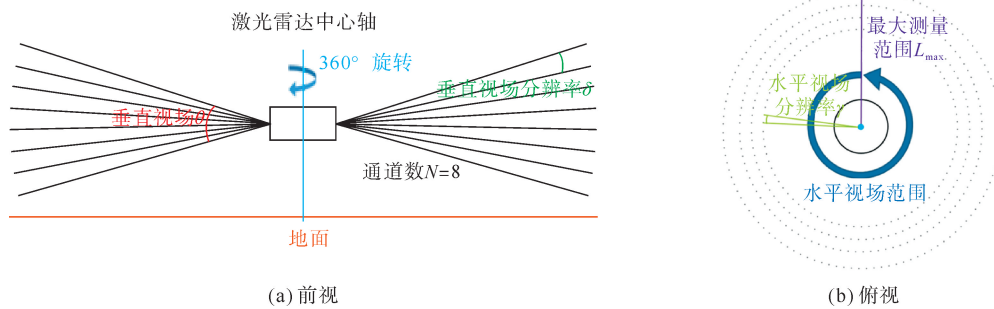


图2 LiDAR 内置特征在前视图和俯视图中的说明

2 基于二阶段模型的道路使用者检测算法

算法整体架构如图 3 所示,首先对 LiDAR 点云的 BEV 映射方式进行改进,使其保留点云的高度信息,然后对映射后的点云使用 3×3 卷积、批标准化和最大池化,最后通过两阶段检测模型对道路使用者进行检测。该模型的第一阶段使用改进的 ResNet 提取 BEV 映射后点云的特征,第二阶段将第一阶段输出的特征图经过并行 MLP 对其进行分类和位置回归。

2.1 改进后的 BEV 映射方式

在交通场景中,相比于摄像机采集到的正视图,点云 BEV 中的目标在同一平面的不同位置,目标彼此不会重叠,因而检测准确度较高。传统的点云 BEV 映射过程中,直接将三维点云映射到地面,丢失了点云高度信息。为了保留鸟瞰图映射过程中的高度信息,借鉴 VoxelNet 方法的思想^[14],将三维点云划分为一个一个立方体(体素),沿着 Z 轴方向将点云进行压缩,从而映射为带有高度信息的鸟瞰图。改进后的 BEV 映射方式如图 4 所示,具体步骤如下:

- 1) 输入原始三维点云,将点云以 LiDAR 坐标系原点为中心划分为 $l \times w \times h$ 个立方体;
- 2) 判断每一个小立方体内是否包含点云;
- 3) 如果小立方体包含点云,则该小立方体的值为 1,否则为 0;
- 4) 输出映射后的 BEV 点云图。

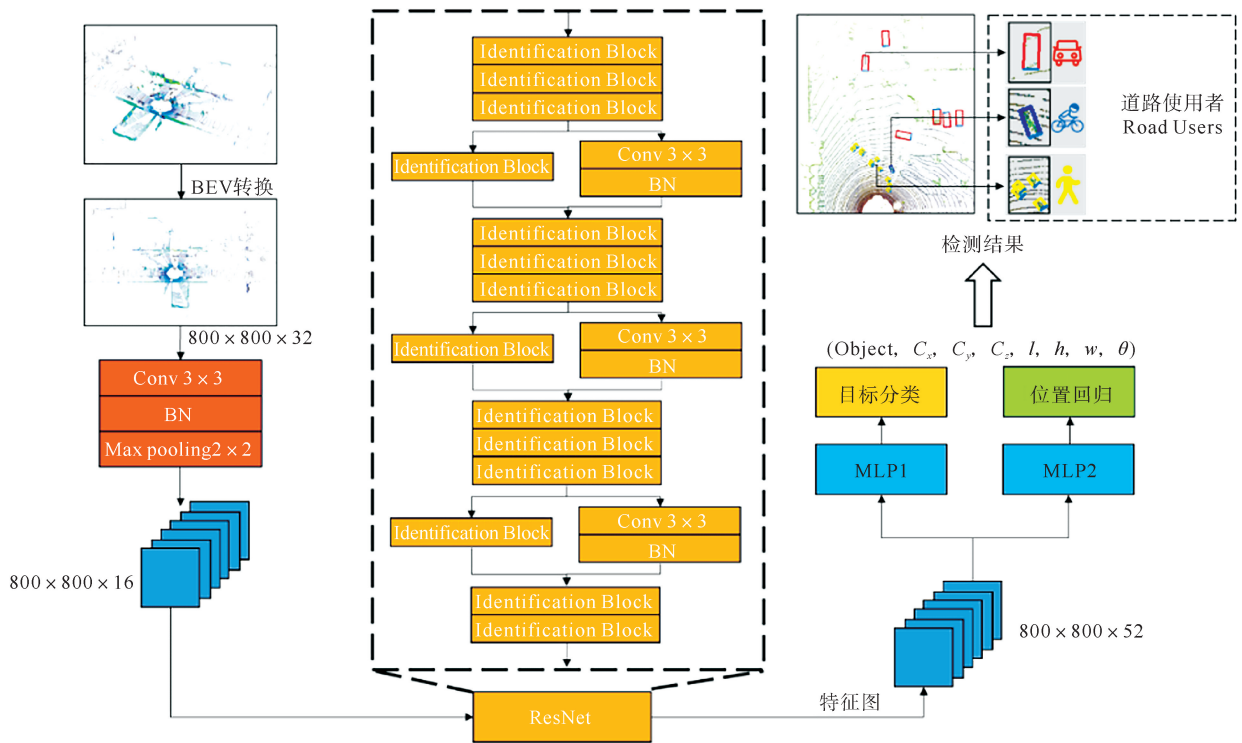


图 3 基于二阶段模型的车辆检测算法整体架构

为了确定 BEV 映射过程中 l, w 和 h 的值,本文参考了 KITTI 数据集的数据。该数据集使用 Velodyne HDL-64E 型号的 LiDAR,垂直视场范围为 $[-24.8^\circ, 2^\circ]$, LiDAR 的安装高度为 1.73 m,最远探测距离为 100 m,如图 5 所示。由于距离 LiDAR 较远的点云过于稀疏,利用价值不大,因此只考虑距 LiDAR 40 m 范围内的点云,通过计算可以得出最远处点云的最高点距离地面高度为 3.13 m,向上取整为 3.2 m。用分辨率为 0.1 m 的立方体对该范围内的点云进行切片,可得 BEV 尺寸为 $800 \times 800 \times 32$ 。

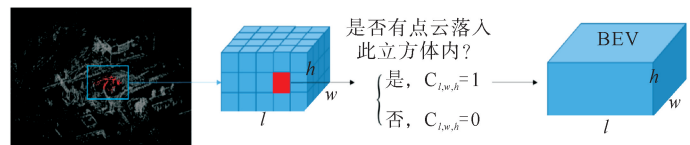


图 4 改进后的 BEV 映射方式

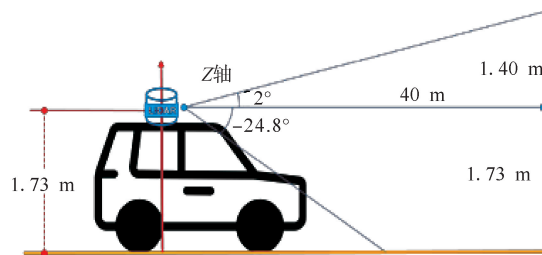


图 5 LiDAR 安装方式及探测范围

2.2 ResNet-MLP 模型结构

深度卷积神经网络(DCNN)融合了不同层次的特征,如局部特征和全局特征,因此可以通过增加网络的层数获得更加丰富的特征。然而,如果网络层次太深,传统的卷积神经网络(CNN)会存在退化问题,即当网络层数达到一定程度且过于复杂时,算法的准确率将达到饱和,然后迅速下降。ResNet 将原 DCNN 的若干层原有网络变为一个残差识别块,能够确保特征提取过程中输入和输出相同。残差识别块结构如图 6 所示。

其中, x 为残差识别块的输入,该输入经过一次线性变换和 ReLU 激活后再经过一次线性变换,变为残差 $F(x)$:

$$F(x) = W_2 * \Gamma(W_1 * x) \quad (3)$$

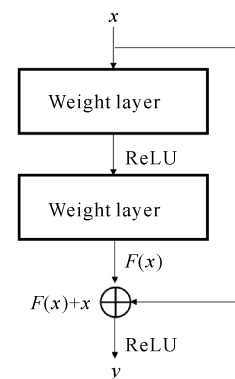


图 6 残差识别块结构

式中: W_1 、 W_2 为卷积核权重; Γ 为激活函数 ReLU; $*$ 为卷积运算符号。

残差 $F(x)$ 与最初的输入 x 相加, 然后经过 ReLU 激活后输出 y :

$$y = \Gamma(F(x) + x) \quad (4)$$

$F(x) + x$ 这条路径被称为捷径连接, 即在输入和输出之间建立一个直接相连的通道可以使参数优化层专注于学习从输入到输出的残差识别器。

采用 ResNet34 的结构, 并对其进行改进以适应 BEV 转换后的点云数据。原始的 ResNet34 由 4 部分组成, 每个部分分别有 3、4、6、3 个残差识别块。每个部分的每个残差识别块分别有 64、128、256 和 512 个过滤器。在本模型中, 由于经 BEV 转换后的点云尺寸较小 ($800 \times 800 \times 32$), 将 4 个部分的残差识别块数量减少为 3、3、3、2 个, 每个残差识别块使用的滤波器数量分别减少为 16、28、50 和 52 个, 并且不使用输出分类层, 最终输出特征图的尺寸为 $800 \times 800 \times 52$ 。修改后的残差识别块如图 7 所示, 由 2 个 3×3 卷积层 (Conv) 和 2 个批标准化层 (BN) 组成, 该模块可以提取更多的特征, 减少网络因处理的数据量较大而导致的性能不稳定性。

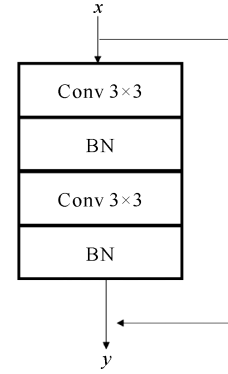


图 7 本文改进后的残差识别块结构

将 ResNet 输出的特征图输入到并行 MLP 中, 分别得到其目标分类结果和位置回归结果。MLP1 目标分类网络由 3 个卷积层 (卷积核尺寸为 3×3 , 卷积核数量分别为 52、36、18), 1 个分类层 (卷积核尺寸为 3×3 , 卷积核数量为 1) 组成。MLP2 位置回归网络也由 3 个卷积层 (卷积核尺寸为 3×3 , 卷积核数量分别为 52、36、18) 和 1 个回归层 (卷积核尺寸为 3×3 , 卷积核数量为 6) 组成, 如图 8 所示。

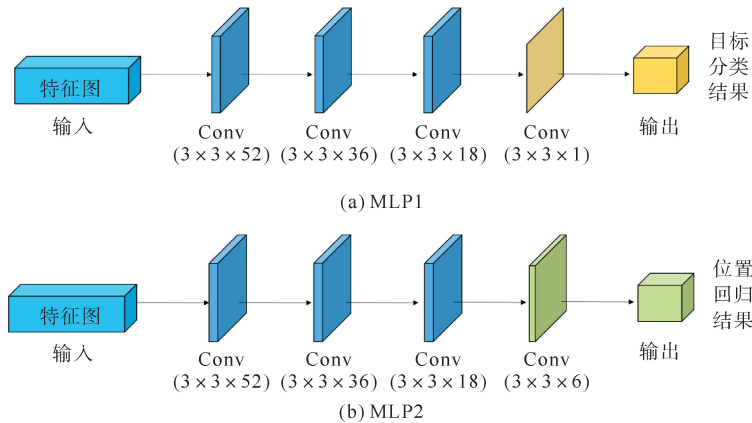


图 8 网络结构

2.3 目标优化及网络训练

对于构造好的 ResNet-MLP 两阶段目标检测模型, 使用随机梯度下降法 (SGD) 进行反向传播, 并对网络权重与偏置进行迭代更新:

$$v = \beta v - \alpha \nabla \omega \quad (5)$$

式中: v 为动量因子; β 为动量因子系数; α 为学习率; ω 为梯度; 当负梯度的方向与 v 的方向相同时, 表示更新的方向是正确的。

模型的损失函数主要包括分类损失 ζ_{object} 和 3D 包围框估计的损失 ζ_{box} , 对损失函数进行加权, 可得总的损失函数为

$$\zeta_{\text{total}} = \mu_1 \zeta_{\text{object}} + \mu_2 \zeta_{\text{box}} \quad (6)$$

式中: μ_1 和 μ_2 为权重系数。

对于目标分类损失,使用二进制交叉熵进行计算:

$$\zeta_{\text{object}} = N^p \times \log N^t - (1 - N^p) \log(1 - N^t) \quad (7)$$

式中: N^p 、 N^t 分别为目标类别的预测值和真实值。

交并比(Intersection over Union, IOU)损失函数是位置回归验证的经典损失函数,它反映了模型所预测的 3D 包围框与真实包围框间的交合程度,如图 9 所示。黄色的为真实框,绿色的为模型预测框,那么预测框 $P_{3\text{Dbox}}$ 和真实框 $T_{3\text{Dbox}}$ 的 IOU 可以由式(8)进行计算:

$$I_{\text{OU}} = \frac{|T_{3\text{Dbox}} \cap P_{3\text{Dbox}}|}{|T_{3\text{Dbox}} \cup P_{3\text{Dbox}}|} \quad (8)$$

然而,式(8)存在一些局限性,如果预测框和真实框没有交集,则 $I_{\text{OU}}=0$,这会导致梯度函数为零,网络无法学习和更新其参数。若 IOU 相等,则无法反映预测框方向、形状和大小的问题,如图 9 所示。

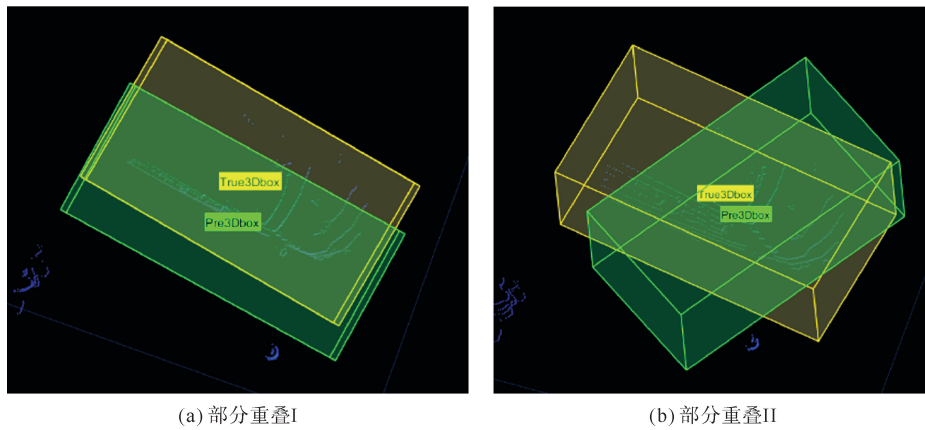


图 9 3D 包围框 IOU 示意($I_{\text{OU}}=0.56$)

因此,在本研究中,改进了损失函数,使其不仅考虑到预测框和真实框间的重叠量,还考虑到两框间的相似性。改进后的损失函数 IOU 公式为

$$\zeta_{\text{box}} = 1 + \lambda \Theta + \frac{\vartheta(C^p, C^t)}{l} - I_{\text{OU}} \quad (9)$$

式中: λ 为权重系数,计算公式如式(10)所示; Θ 为度量预测框和真实框的相似性指标,计算公式如式(11)所示; C^p 和 C^t 分别为预测的 3D 包围框和真实包围框的中心坐标; ϑ 为计算两点坐标点间的欧氏距离; l 为能够同时包含预测框和真实框的最小凸包区域的最长距离。

$$\lambda = \frac{\Theta}{\Theta + 1 - I_{\text{OU}}} \quad (10)$$

$$\Theta = \frac{4}{\pi^2} \left(\arctan \frac{\omega^p}{l^p} - \arctan \frac{\omega^t}{l^t} \right)^2 + \left(\arctan \frac{l^p}{h^p} - \arctan \frac{l^t}{h^t} \right)^2 + \left(\arctan \frac{h^p}{\omega^p} - \arctan \frac{h^t}{\omega^t} \right)^2 \quad (11)$$

式中: l^p 、 ω^p 、 h^p 分别为预测框的长、宽、高; l^t 、 ω^t 、 h^t 分别为真实框的长、宽、高。

3 实验验证与结果分析

3.1 实验准备

3.1.1 数据集

采用的点云数据集来源于 KITTI 3D object 数据集,该数据集由搭载了 Velodyne HDL-64E 型号 LiDAR 的车辆在真实的道路环境中采集而来。采集到的每一帧点云的数据格式为 .bin,是一个 n 行(n 个三维空间点)4 列(x 坐标、 y 坐标、 z 坐标、反射强度)的矩阵,该矩阵对应的标签为 m 行(m 个目标)8 列(目标类别,目标中心点的 x 、 y 、 z 坐标,目标包围框的长、宽、高及目标航向角),将训练样本按照 4 : 1 的比例拆分为训练集和验证集。

3.1.2 软硬件配置

采用的 PC 配置为: Intel(R) Core (TM) i5-9400 CPU@2.9GHz, 内存为 DDR4 16 G, 显卡为 GeForce GTX 1050Ti (4 G 显存); 操作系统为 Win10。

3.1.3 深度学习超参数配置

深度学习训练过程中的使用到的参数如表 1 所示。

3.1.4 评价指标

为了评估所提方法的性能, 本文采用三维目标检测常用的评价指标: 准确率 A_{3D} 、三维检测框平均准确率 m_{3D} 、检测速度。

$$A_{3D} = \frac{1}{11} \sum_{R=0}^1 \max P(c), R(c) \geq r \quad (12)$$

式中: $P(c)$ 为第 c 类目标检测的准确率; $R(c)$ 为第 c 类目标检测的召回率; R 为目标检测召回率, 取 $R \in \{0, 0.1, \dots, 1\}$; r 为召回率精度。

$$\begin{cases} P(c) = \frac{T(c)}{T(c) + F(c)} \\ R(c) = \frac{T(c)}{T(c) + F(v)} \end{cases} \quad (13)$$

式中: $T(c)$ 为第 c 类正确检测的样本; $F(c)$ 为第 c 类误检的样本; $F(v)$ 为第 c 类漏检的样本。

m_{3D} 为各类别 A 的平均值, 可以用式(14)计算:

$$m_{3D} = \frac{1}{K} \sum_{c=1}^K A_{3D} \quad (14)$$

式中: K 为目标类别总数, 取 $K = 3$ 。

3.2 实验结果与分析

经过 300 次迭代训练后, 模型验证集预测精度稳定在 92% 左右, 损失函数趋近于 0, 将训练后的模型权重应用在测试集上, 部分车辆目标检测的可视化结果如图 10 所示。

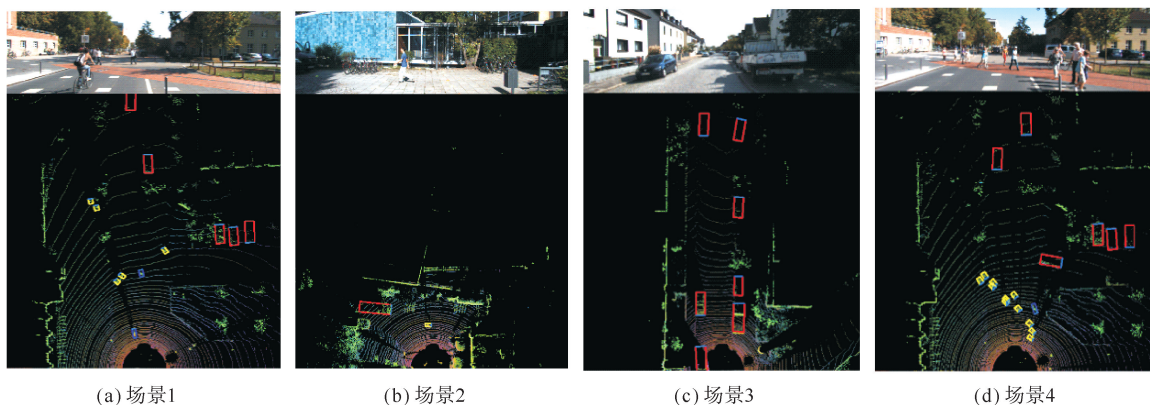


图 10 不同场景下的时空检测结果

从图 10 中 4 个场景点云鸟瞰图和视频图像的对比中可以发现, 车辆对象类别及位置均能较为正确地检测。

不同算法(PointNet++, VoxelNet 和基于 RseNet-MLP 二阶段检测模型)在不同 IOU 下的对比结

果如表 2 所示,量化结果如图 11 所示。

表 2 不同算法的对比结果

IOU 取值	算法	$A_{3D}(\text{car})/\%$	$A_{3D}(\text{ped})/\%$	$A_{3D}(\text{cyc})/\%$	$m_{3D}/\%$	检测速度/(帧·s ⁻¹)
0.5	PointNet++	71.73	60.50	62.64	64.96	0.8
	VoxelNet	80.19	67.03	76.18	74.47	2.2
	本文方法	93.17	89.24	90.96	91.12	15.2
0.7	PointNet++	60.23	49.15	56.91	55.43	0.8
	VoxelNet	69.83	54.75	67.93	64.17	2.0
	本文方法	81.65	66.19	72.84	73.56	13.4

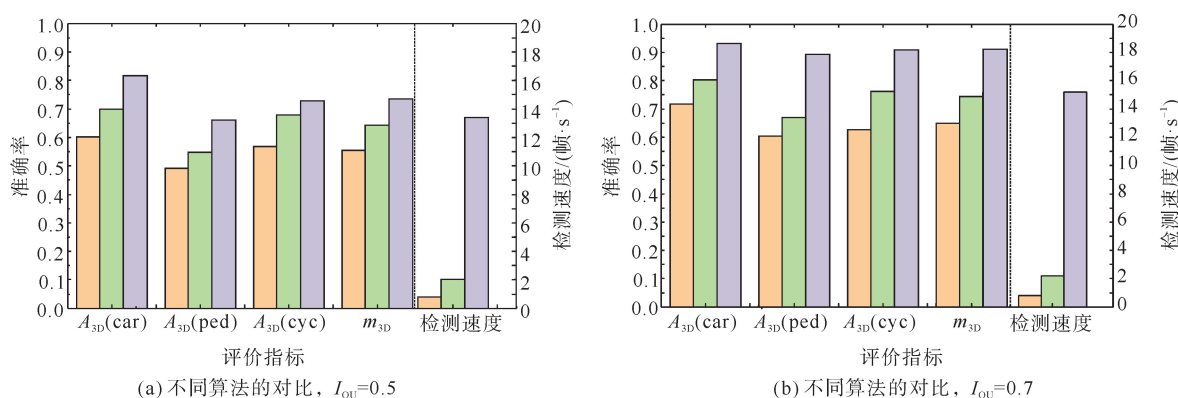


图 11 不同IOU下3种算法的检测性能对比

PointNet++; VoxelNet; 本文方法

通过对比分析可以发现,车辆目标检测方法识别准确率高 PointNet++和 VoxelNet。车辆、行人、自行车/电动车和所有目标的平均识别率在 $I_{OU} = 0.5$ 时分别达到了 93.17%、89.24%、90.96% 和 91.12%,检测速度达到了 15.2 帧/s,远远高于其他两种方法。在 $I_{OU} = 0.7$ 时,本文方法的识别准确率和检测速度较 $I_{OU} = 0.5$ 时虽然略有下降,但仍然高于其他两种方法。当 IOU 提高时,所有方法的识别准确率均略微下降,可能是 $I_{OU} = 0.7$ 时的检测条件过于苛刻,即当预测框与真实框的交并比大于 0.7 时,检测算法才认为该目标被正确检测,这在三维点云检测中是比较困难的。

综上所述,基于 ResNet-MLP 二阶段模型的车辆检测算法能够较好地完成目标检测任务,且相对于 PointNet++和 VoxelNet 方法检测准确率更高,运行速度更快。

4 结束语

通过提出了一个两阶段目标检测框架,首先将原始点云映射为具有高度信息的鸟瞰图,然后通过改进的 ResNet 提取点云鸟瞰图特征,最后使用并行 MLP 进行目标检测的分类和位置回归。通过与 PointNet++和 VoxelNet 两种方法进行对比,本文提出的方法提高了车辆检测准确率和运行速度,但是对于工程应用来说,该速度仍然难以满足检测实时性的要求。后续研究中,将采取新的研究思路,重点提高目标检测的效率,达到实时检测,并进行实车实地实验。

参考文献 (References):

[1] 彭育辉,郑玮鸿,张剑锋. 基于深度学习的三维目标检测方法综述[J]. 汽车技术, 2020(9): 1-7.
 PENG Yuhui, ZHENG Weihong, ZhANG Jianfeng. A review of 3D target detection methods based on deep learning[J]. Automotive Technology, 2020(9): 1-7.

[2] 张婷婷,章坚武,郭春生,等. 基于深度学习的图像目标检测算法综述[J]. 电信科学, 2020, 36(7): 92-106.

- ZHANG Tingting, ZHANG Jianwu, GUO Chunsheng, et al. A review of deep learning-based image target detection algorithms[J]. Telecommunications Science, 2020, 36(7): 92-106.
- [3] 曲大义, 杨建, 王进展, 等. 基于雷达多目标检测技术的交通信号智能控制系统[J]. 青岛理工大学学报, 2017, 38(2): 1-8.
QU Dayi, YANG Jian, WANG Jinzhan, et al. Intelligent traffic signal control system based on radar multi-target detection technology[J]. Journal of Qingdao University of Technology, 2017, 38(2): 1-8.
- [4] WU J Q. An automatic procedure for vehicle tracking with a roadside LiDAR sensor[J]. ITE Journal, 2018, 88(11): 32-37.
- [5] 张小玉. 基于 LiDAR 机器视觉的目标检测与分类[D]. 天津: 天津工业大学, 2019.
ZHANG Xiaoyu. LiDAR-based machine vision for target detection and classification[D]. Tianjin: Tianjin University of Technology, 2019.
- [6] 胡舒, 王树根, 王越, 等. 基于车载 LiDAR 点云的车辆目标自动检测[J]. 地理空间信息, 2021, 19(3): 10-13.
HU Shu, WANG Shugen, WANG Yue, et al. Automatic detection of vehicle targets based on vehicle LiDAR point cloud[J]. Geospatial Information, 2021, 19(3): 10-13.
- [7] 彭育辉, 郑玮鸿, 张剑锋. 基于深度学习的道路障碍物检测方法[J]. 计算机应用, 2020, 40(8): 2428-2433.
PENG Yuhui, ZHENG Weihong, ZHANG Jianfeng. A deep learning-based method for road obstacle detection[J]. Computer Applications, 2020, 40(8): 2428-2433.
- [8] QI C R, SU H, MO K, et al. PointNet: Deep learning on point sets for 3D classification and segmentation[C]//30th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017). Hawaii: 2017: 77-85.
- [9] QI C R, YI L, SU H, et al. PointNet ++: Deep hierarchical feature learning on point sets in a metric space[C]//31st Conference on Neural Information Processing Systems (NIPS). Los Angeles: 2017.
- [10] YANG B, LUO W, URTASUN R. PIXOR: Real-time 3D object detection from point clouds[C]//Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City: 2018: 7652-7660.
- [11] 王伟舒. 基于 LiDAR 感知增强的车辆避撞方法研究[D]. 北京: 北京交通大学, 2018.
WANG Weishu. Research on vehicle collision avoidance method based on LiDAR perception enhancement[D]. Beijing: Beijing Jiaotong University, 2018.
- [12] LIN C Y, SUN G H, TAN L D, et al. Mobile LiDAR deployment optimization: Towards application for pavement marking stained and worn detection[J]. IEEE Sensors Journal, 2022, 22(4): 3270-3280.
- [13] 李雨欣. 基于激光雷达的点云数据处理算法研究[D]. 长春: 长春理工大学, 2020.
LI Yuxin. Research on LiDAR-based point cloud data processing algorithm[D]. Changchun: Changchun University of Technology, 2020.
- [14] ZHOU Y, TUZEL O. VoxelNet: End-to-end learning for point cloud based 3D object detection[C]//Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Salt Lake City: 2018: 4490-4499.

(责任编辑 张晓靖; 英文校审 程文华)