激光与光电子学进展 先进成像

基于多阶段信息增强的3D点云目标检测算法

袁善帅[1,2]，丁雷[1,2,3*]

[1]中国科学院上海技术物理研究所红外探测与成像技术重点实验室，上海 200083；

[2]上海科技大学信息科学与技术学院，上海 201210；

[3]中国科学院大学，北京 100049

摘要 自动驾驶场景中，通常会用基于体素化的算法来完成点云3D目标检测任务，因为该类方法拥有计算量少、耗时少等方面的优势。但是当下常用的方法往往会带来双重信息损失，其一是体素化带来的量化误差造成的，其二则是对体素化后的点云信息利用不充分造成的。设计一个三阶段的网络结构来解决信息损失大的问题。第一阶段使用基于体素化的优秀算法完成输出边界框的任务；第二阶段利用一阶段特征图上的信息精修边界框，以解决一阶段对输入信息利用不充分的问题；第三阶段利用了原始点的精确位置信息再次精修边界框，以弥补体素化带来的点云信息损失。在Waymo Open Dataset上，所提多阶段3D目标检测算法的检测精度超过了CenterPoint等受工业界青睐的优秀算法，且满足自动驾驶落地的时间要求。

关键词 机器视觉；3D目标检测；激光点云；多阶段；信息增强

中图分类号 TP391 **文献标志码** A **DOI**：10.3788/LOP223207

Three-Dimensional Object Detection Based on Multistage Information Enhancement in Point Clouds

Yuan Shanshuai[1,2], Ding Lei[1,2,3*]

[1]*Key Laboratory of Infrared System Detection and Imaging Technology, Shanghai Institute of Technical Physics, Chinese Academy of Sciences, Shanghai* 200083, *China*;

[2]*School of Information Science and Technology, ShanghaiTech University, Shanghai* 201210, *China*;

[3]*University of Chinese Academy of Sciences, Beijing* 100049, *China*

Abstract Voxel-based method is usually used in autonomous driving when conducting three-dimensional (3D) object detection based on a point cloud. This method is associated with small computational complexity and small latency. However, the current algorithms used in the industry often result in double information loss. Voxelization can bring information loss of point cloud. In addition, these algorithms do not entirely utilize the point cloud information after voxelization. Thus, this study designs a three-stage network to solve the problem of large information loss. In the first stage, an excellent voxel-based algorithm is used to output the proposal bounding box. In the second stage, the information on the feature map associated with the proposal is used to refine the bounding box, which aims to solve the problem of insufficient information utilization. The third stage uses the precise location of the original points, which make up for the information loss caused by voxelization. On the Waymo Open Dataset, the detection accuracy of the proposed multistage 3D object detection method is better than CenterPoint and other excellent algorithms favored by the industry. Meanwhile, it meets the requirement of latency for autonomous driving.

Key words machine vision; three-dimensional object detection; laser point cloud; multistage; information enhancement

1 引 言

3D目标检测是自动驾驶场景下一个非常重要的任务，对自车感知周围障碍物以及多目标跟踪、规划控制等下游任务具有重要意义。虽然图像包含丰富的语义信息和纹理信息，但是深度信息的缺失导致其在面对3D检测任务时表现不佳，而激光雷达获取的点云包含有高质量的深度信息，因此点云相比图像在3D检测任务具有天然优势。所以基于激光雷达点云的3D目标检测算法是目前工程实践中使用的主要方法，也是

收稿日期：2022-11-30；**修回日期**：2022-12-30；**录用日期**：2023-01-17；**网络首发日期**：2023-02-07

通信作者：*leiding@mail.sitp.ac.cn

当下研究的热点。

基于点云的3D目标检测需要以大场景点云作为输入，输出七维(7D)的目标3D边界框。这7维信息包括边界框中心点的位置(x,y,z)、边界框的长宽高、朝向角。点云具有稀疏性、无序性和分布不均匀性等特点。PointNet[1]使用多层感知机处理点云并使用最大池化算子应对点云的无序性。PointNet++[2]利用基于点云的层次化分组方法来处理点云数据。基于PointNet系列算法[1]，有一些直接对原始点云进行处理的3D目标检测算法，通常这类方法被称作基于原始点云的方法。PointRCNN[3]中涉及基于全局点云分割的无锚框3D候选框生成子网络。STD[4]中涉及由稀疏到密集的策略，以更好地精修候选框。Part A2[5]在PointRCNN[3]的基础上具有一个可以感知感兴趣区域(ROI)的点云池化操作。VoteNet[6]从另一个角度出发，基于激光雷达点云都分布在物体表面的现象，提出了一种基于深度霍夫投票的新的候选框生成机制。

卷积神经网络(CNN)已经在2D图像任务中展示出了强大的信息抽取能力，但是无法对分布不均匀的点云直接进行处理，因此在空间画小方格，将点云体素化以生成卷积神经网络可处理的特征，该方法成为了一种在工业界受欢迎的方法。通常这类方法被称作基于体素化的方法。VoxelNet[7]使用PointNet[1]，在体素内抽取特征得到每个体素的特征向量。SECOND[8]通过应用稀疏卷积算子，解决了密集3D卷积计算量过大的问题。PointPillars[9]用立柱对点云进行体素化，避免了3D卷积算子的使用，提高了速度，有利于部署。CenterPoint[10]使用VoxelNet[7]或PointPillars[9]对点云进行体素化，生成鸟瞰视角下的伪图，之后使用无锚框的检测器进行检测。PV-RCNN[11]将原始点云的特征聚合进体素化的算法框架中。3D目标检测使用两阶段的结构，将一阶段输出的边界框作为候选框，利用二阶段网络对候选框进行精修，得到更精准的边界框。CenterPoint[10]使用一阶段特征图上与候选框有关的特征精修目标边界框。PV-RCNN[11]在ROI内聚合原始点的信息。Pyramid-RCNN[12]在ROI内使用金字塔结构解决点云稀疏的问题。Voxel-RCNN[13]直接使用ROI相关的体素特征来增强信息。

出于计算资源、实时性和方便部署等方面的考虑，工程实践中通常会使用基于体素化的算法框架。自动驾驶场景下一帧点云中点的数量可以达100×10^3量级，直接使用PointNet[1]去处理挑战过大，实践中通常会使用类似PointNet++[2]的层次化分组方法，但该方法会带来难以承受的计算量。所以基于原始点云的方法在实践中存在计算量和时延方面的瓶颈。基于体素化的方法可以避免上述问题，能够满足工业应用中关于计算量和时延的要求，但是体素化会带来量化误差，相比于使用原始点精确位置信息的基于原始点云的方法，基于体素化的方法的检测精度会更低一些。为了平衡基于体素化的方法与基于原始点的方法的优缺点，提出了一阶段使用基于体素化的方法，二阶段使用基于原始点云的方法的LiDAR R-CNN[14]，在二阶段仅对候选框里的点云进行处理，既利用了点的精确位置信息，又解决了处理原始点时计算量过大的问题。

本文在基于体素化的算法框架的基础上，增加小型网络模块来构建多阶段网络，让后续模块弥补第一阶段网络的不足，以提升网络的性能。本文设计了两种类型的小型网络，可分别作为二阶段和三阶段网络来使用。第一类小型网络使用一阶段网络特征图中的信息来精修输出的边界框，本文称其为基于特征图的网络；第二类小型网络利用了原始点云的信息，本文称其为基于原始点云的网络。本文研究了两类网络的性能，并提出了两者级联使用的方案。本文的贡献：1)分别设计了轻量化的基于特征图的网络和基于原始点云的网络；2)在一阶段网络的基础上，先后使用基于特征图的网络和基于原始点云的网络，形成了多阶段的网络结构，特征图的信息和原始点云的信息是对一阶段网络的有力补充，使三阶段网络的性能相比于一阶段网络有显著提升；3)设计交并比损失(IoU Loss)提升一阶段网络学习边界框大小的能力，使用SiLU激活函数促进一阶段网络的收敛，使用IoU分支和优化后的置信度分数计算方法提高基于原始点云的网络的性能。

2 算法原理

2.1 算法框架

所提多阶段算法框架如图1所示。第一个阶段为基于体素化的方法，在当下优秀算法的基础上，使用IoU Loss增强网络对边界框大小的学习能力，并使用了SiLU激活函数，输出图1中的候选框(proposal)；第二个阶段，利用图1中第一阶段特征图(feature map)上的信息，对每个proposal进行信息增强，图1中的feature-based network通过学习增强后的信息，得到精修后的目标边界框(refined 3D box)，称这一阶段为基于特征图的网络；第三阶段，将图1中每个refined 3D box中的点云抠出来，送入points-based network，输出最终的目标边界框，称这一阶段为基于原始点云的网络。

2.2 第一阶段的网络

这一部分的网络结构与CenterPoint[10]一阶段的网络结构相似，首先使用VoxelNet[7]或PointPillars[9]作为3D backbone(3D骨干网络)，将点云编码成鸟瞰视角下的伪图；之后使用一个具有多尺度信息聚合能力的网络(图1中的neck)对该伪图进行信息增强和多尺度信息融合，得到特征图；最后将此特征图送入基于中心点的检测头，输出图1中的proposal。输出的proposal包含7D边界框信息和一个表征分类质量的分数。所提算法的一阶段使用SiLU激活函数替代CenterPoint[10]使用的ReLU激活函数。SiLU的计算公式为

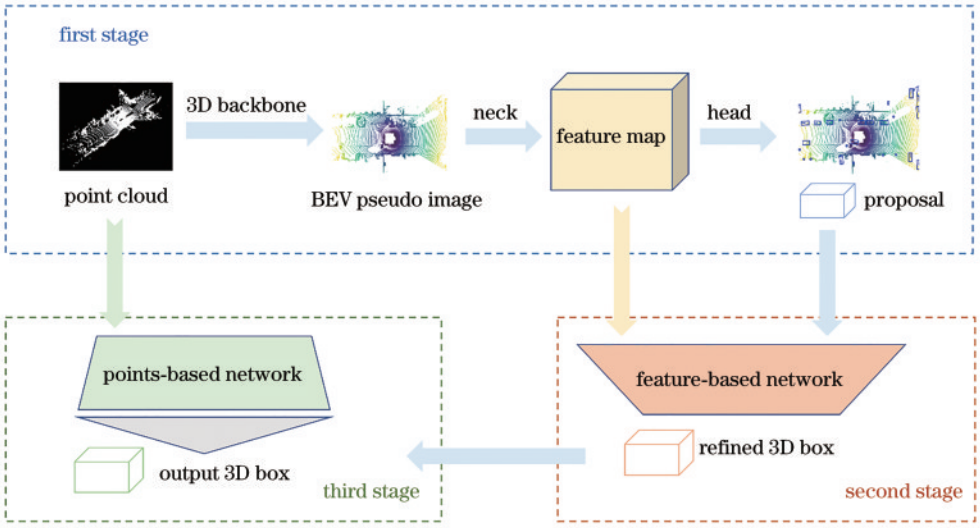

图 1　所提算法的总体框架

Fig. 1　Overall framework of the proposed algorithm

$$f_{\mathrm{SiLU}}(x)=x\times\mathrm{Sigmoid}(x)=\frac{x}{1+\mathrm{e}^{-x}},\qquad(1)$$

式中：x 指函数的输入。SiLU 激活函数在输入 x 接近 0 时，梯度会变小，输入等于 0 时不存在跳变，因此更有利于网络的收敛。

在计算输出边界框长宽高时，CenterPoint[10] 等优秀算法使用损失函数直接约束长宽高三个数值，本文对此方案进行了改进。进行坐标系转换，使预测框与对应真值框的中心点重合，朝向角一致，如图 2 所示。

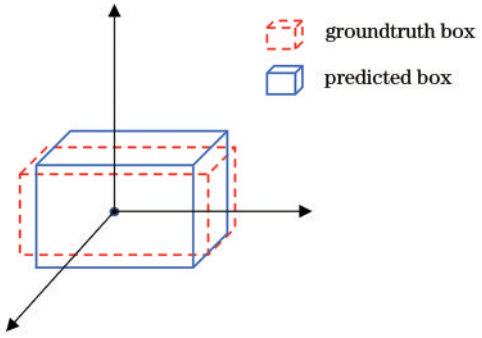

图 2　计算 IoU Loss 时坐标系转换后的结果

Fig. 2　Result after coordinate conversion when calculating IoU Loss

在此情况下计算两个三维框的 3D IoU，用 1-IoU 作为损失监督网络对边界框长宽高进行学习。由于网络学习边界框长宽高的最终目标是使得预测框与真值框的 3D IoU 尽可能小，所以此处使用 IoU Loss 约束网络更加直接高效，有利于网络预测出更精准的边界框大小。

2.3　基于特征图的网络

第一阶段的网络只是利用了特征图上目标中心点处的特征来回归边界框的 7D 信息，并没有充分利用与边界框有关的信息。一阶段的输出为图 1 中的 proposal。本文设计了一个轻量化的网络，利用一阶段特征图上与 proposal 有关但未被使用的信息，来精修 proposal，并称此网络为基于特征图的网络。基于特征图的网络如图 3 所示，首先将每个 proposal 投影到对应的特征图上，得到二维边界框，抽取特征图在二维框 4 个角点处的特征向量，将这 4 个特征向量与 proposal 中心点处的特征向量沿着通道维度拼接在一起，送入多层感知机（MLP）进行训练。CenterPoint[10] 提供的二阶段网络有个相似的做法，其将 proposal 6 个面的中点投影到对应特征图上，得到 5 个点，使用双线性插值方法对这 5 个点进行处理，生成特征图上的 2D 坐标，

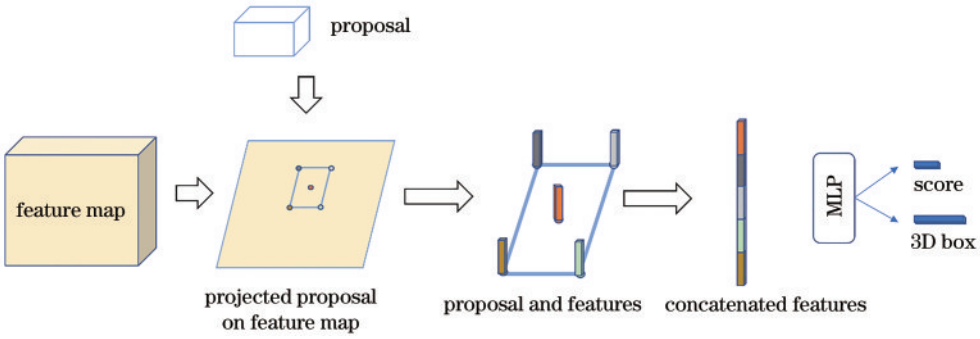

图 3　基于特征图的网络

Fig. 3　Feature-based network

并将这5个坐标处的特征向量抽取出来送入多层感知机进行训练。CenterPoint[10]的方法相对复杂。因为后面还需要添加一个基于原始点云的网络，所以本文这里选择了使用4个角点的简洁方案。该阶段网络的输出有边界框7D信息和分数两个部分，边界框的7D信息由L1损失函数约束。在计算分数时，本文参考了CenterPoint[10]的做法，使用I监督分数预测头的训练，I的计算方式为

$$I=\min\left[1,\max\left(0,2\times R_{\mathrm{IoU}n}-0.5\right)\right],\tag{2}$$

式中：$R_{\mathrm{IoU}n}$是第n个预测框与其对应真值框的IoU值。训练时，使用二分类交叉熵损失函数，表达式为

$$\mathcal{L}=-I_n\log\hat{I}_n-(1-I_n)\log(1-\hat{I}_n),\tag{3}$$

式中：$\hat{I}_n$是此阶段网络分数头预测出的置信度分数。推理时，对第一阶段的分数和此阶段的分数取几何平均值，得到最终的分数：

$$\hat{S}_n=\sqrt{\hat{C}_n\times\hat{I}_n},\tag{4}$$

式中：$\hat{C}_n=\max_{0\leqslant k\leqslant K}\hat{C}_{p,k}$是第$n$个目标对应的一阶段proposal的分数；$\hat{I}_n$是此阶段的分数预测头针对第$n$个目标输出的分数。

基于特征图的网络输出的最终分数同时包含分类信息和IoU信息，使得高分目标边界框不仅类别较为准确，还拥有较精确的位置、大小和朝向角，为后面的利用非极大值抑制判断正负样本提供了更好的分数值，有利于提高预测结果的平均精度(AP)。

2.4　基于原始点云的网络

所提算法的前两阶段均是在点云体素化的基础上进行的，体素化带来的量化误差始终存在。为了利用原始点的精确位置信息弥补体素化带来的信息损失，同时避免过大的计算复杂度，将在算法的最后阶段使用一个基于原始点云的网络，该网络只关注与图1中refined 3D box相关的原始点，忽略无关背景点，网络结构如图4所示。

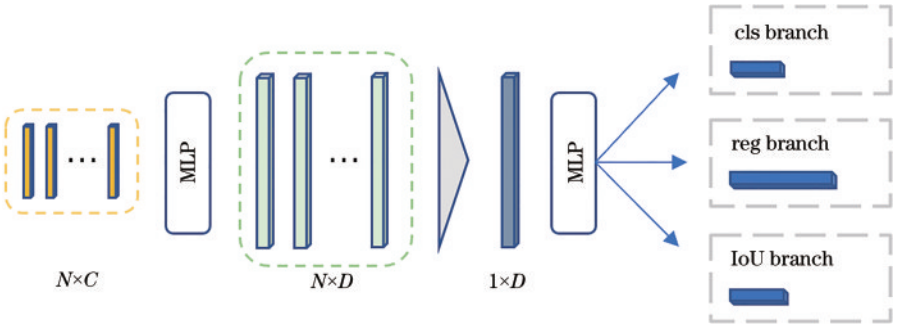

图4　基于原始点云的网络

Fig. 4　Point-based network

对于基于原始点云的网络，前序网络输出的边界框可以认为是候选框。在所提多阶段网络中，候选框对应图1中的refined 3D box。与LiDAR R-CNN[14]相似，基于原始点云的网络会将候选框适当扩大，形成ROI，之后将ROI里的点取出，送入图4所示的PointNet[1]。ROI里的这组点负责预测与候选框对应的更精确的边界框，这些更精确的边界框即是多阶段网络的最终输出，也就是图1中的output 3D box。本文借鉴了LiDAR R-CNN[14]处理尺寸歧义问题的方法，对于一组点云中的每个点，计算其到对应候选框6个面的距离，将这6个距离信息与点的特征向量拼接在一起，送入基于原始点云的网络，以使网络可以感知候选框的大小。图4中，N是一组点云中点的数量，C为9，分别代表点的位置(x,y,z)和点到所属框6个面的距离。经过多层感知机的处理，每个点的特征向量大小变为D，本文中D为512。使用最大池化得到一个长度为D的向量，将该向量作为该组点云的特征向量，将此特征向量送入后面的预测头。

LiDAR R-CNN[14]的预测头仅包含分类分支和回归分支。受CIA-SSD[15]和AFDetv2[16]的启发，本文增加了一个IoU预测分支，使用$2\times R_{\mathrm{IoU}}-1\in[-1,1]$作为真值来监督该分支的训练。特别区分于2.2节中的IoU Loss，此处的IoU分支预测出一个表征IoU信息的置信度分数，用于后面最终分数的计算，同时该分支为网络专门加了一个学习预测框与真值框IoU的约束，有利于网络预测出更精准的边界框。在此处优化了置信度分数的计算方法，每个输出边界框的分数f包含分类和IoU两部分，计算方式为

$$f=c_{\mathrm{cls}}{}^{\alpha}\times c_{\mathrm{IoU}}{}^{1-\alpha},\tag{5}$$

式中：c_{cls}是分类分支预测的分数；c_{IoU}是IoU分支预测的分数；α是数值在[0,1]范围内的超参数，用来控制分类分数和IoU分数对总分数f的贡献，本文中$\alpha=0.65$。在2.3节已经分析过，同时包含分类信息和IoU信息的置信度分数有利于提高预测结果的AP。网络中，分类分支使用交叉熵损失函数，回归分支和IoU分支使用Smooth L1损失函数，最终的损失函数$\mathcal{L}_{\mathrm{T}}$为

$$\mathcal{L}_{\mathrm{T}}=\lambda_{\mathrm{cls}}\mathcal{L}_{\mathrm{cls}}+\lambda_{\mathrm{IoU}}\mathcal{L}_{\mathrm{IoU}}+\lambda_{\mathrm{reg}}\mathcal{L}_{\mathrm{reg}},\tag{6}$$

式中：λ是每个子监督头的权重。

3 实验结果与分析

3.1 数据集与实验设置

Waymo Open Dataset(WOD)[17]是用于自动驾驶研究的大规模公开数据集,其点云数据由64线激光雷达采集得到。WOD的训练集有798个点云序列,包含158081帧点云,验证集有202个点云序列,包含39987帧点云,目标可分为3类,分别为车辆(vehicle)、行人(pedestrian)和非机动车(cyclist)。WOD中使用的评价指标是平均精度(AP)和按朝向角精度加权的平均精度(APH)。该数据集中的目标可以按难度分为LEVEL_1(L1)和LEVEL_2(L2)两类,若目标上有超过5个点并且没有被标记为LEVEL_2,则该目标被标记为LEVEL_1类别,若目标上的点数少于5或已被手动标记为LEVEL_2,则该目标归为LEVEL_2类别。

所用的硬件配置为Ubuntu 18.04系统和NVIDIA A100显卡。在第一阶段网络和基于特征图的网络中,使用AdamW优化器和one cycle学习率优化策略,最大学习率为3×10^{-3},权重衰减(weight decay)为0.01,动量(momentum)在0.85到0.95之间变化。在基于原始点云的网络中,参考LiDAR R-CNN[14]的设置,使用SGD优化器和poly学习率优化策略,动量(momentum)为0.9,学习率的初始值为0.02,权重衰减为1×10^{-5}。

3.2 算法性能

在WOD的验证集上检验所提多阶段3D目标检测算法的性能。表1展示了不同算法在WOD验证集车辆类别上的表现,其中一些数据来源于3D-MAN[18]。表1中,CenterPoint[10]的3D backbone为VoxelNet[7],使用单帧数据进行训练,网络为单阶段结构。Near指距离自车0到30 m的范围,Middle指距离自车30 m到50 m的范围,Far指距离自车50 m以上的范围,Overall指所有距离范围。每一个单项的最优以加粗形式表示。由表1可以得出:所提多阶段3D目标检测算法在除近距离L2之外的其他场景下均取得了最优效果,且相比于次优的算法,两个指标均提升了约2个百分点;在近距离L2难度的这个场景下,所提算法与最优的PV-RCNN[11]不相上下。近距离L2难度指目标距离自车0到30 m的范围内,且落在目标上的点云点数小于5的情况,所提算法的性能之所以在该场景下没能超越PV-RCNN[11],一个重要原因是PV-RCNN[11]在最开始就使用了原始点云的信息,而所提算法在第三阶段才使用原始点云的信息。在其他场景下,所提三阶段网络尽可能减小信息损失的优势就体现出来了,因此取得了非常优异的表现。

表1 在WOD验证集车辆类别上不同算法的检测结果

Table 1 Vehicle detection results of different algorithms on WOD validation set unit: %

Difficulty	Algorithm	3D AP				3D APH			
		Overall	Near	Middle	Far	Overall	Near	Middle	Far
L1	StarNet[19]	55.11	80.48	48.61	27.74	54.64	79.92	48.10	27.29
	MVF[20]	62.93	86.30	60.02	36.02				
	PointPillars[9]	63.27	84.90	59.18	35.79	62.72	84.35	58.57	35.16
	AFDet[21]	63.69	87.38	62.19	29.27				
	3D-MAN[18]	69.03	87.99	66.55	43.15	68.52	87.57	65.92	42.37
	PV-RCNN[11]	70.30	91.92	69.21	42.17	69.49	91.34	68.53	41.31
	CenterPoint[10]	74.63	90.93	72.90	51.32	74.12	90.50	72.34	50.62
	Proposed algorithm	**77.19**	**92.25**	**75.98**	**54.85**	**76.76**	**91.89**	**75.47**	**54.15**
L2	StarNet[19]	48.69	79.67	43.57	20.53	48.26	79.11	43.11	20.19
	PointPillars[9]	55.18	83.61	53.01	26.73	54.69	83.08	52.46	26.24
	3D-MAN[18]	60.16	87.10	59.27	32.69	59.71	86.68	58.71	32.08
	PV-RCNN[11]	65.36	**91.58**	65.13	36.46	64.79	**91.00**	64.49	35.70
	CenterPoint[10]	66.73	89.78	66.95	40.14	66.26	89.35	66.42	39.57
	Proposed algorithm	**68.70**	91.01	**69.47**	**42.56**	**68.30**	90.66	**69.00**	**41.99**

3.3 消融实验

由于WOD不同点云帧之间间隔较小,所以下面的实验中没有使用全量的数据,而是在WOD训练集中等间隔抽取20%的数据训练网络,以同样的方式在验证集中取20%的数据作为测试集,这样既提高了实验效率,又不会影响实验结果的有效性。所有结果均是在WOD验证集的车辆类别上进行测试得到的。由于在工程实践中PointPillars[9]运行速度快,方便部署,所以实验中一阶段网络的3D backbone均使用PointPillars[9]。

3.3.1 SiLU激活函数与IoU Loss

表2展示了在CenterPoint[10]一阶段网络上ReLU和SiLU的性能对比。在进行非极大值抑制(NMS)时,IoU的阈值设为0.1。由表2结果可知,SiLU激活函数比ReLU更有利于模型的收敛,得到的结果更好。

表3展示了使用SiLU激活函数的CenterPoint[10]

表 2 ReLU与SiLU的对比

Table 2 Comparison between ReLU and SiLU unit: %

Activation function	L1		L2	
	3D AP	3D APH	3D AP	3D APH
ReLU	65.56	65.05	60.81	60.33
SiLU	**66.21**	**65.71**	**61.41**	**60.95**

表 3 原版Loss与IoU Loss的对比

Table 3 Comparison between the original Loss and IoU Loss unit: %

Note	L1		L2	
	3D AP	3D APH	3D AP	3D APH
Original Loss	66.21	65.71	61.41	60.95
IoU Loss	**66.59**	**66.09**	**61.79**	**61.33**

在使用不同损失函数回归边界框长宽高时的结果。Original Loss为CenterPoint[10]原版使用的直接使用L1损失函数计算长宽高的损失。由表3可知，IoU Loss优于CenterPoint[10]使用的原版Loss。

3.3.2 基于特征图的网络和基于原始点云的网络性能探索

表4比较了基于特征图的网络和基于原始点云的网络在不同搭配方案下的性能，其中One stage指所提算法第一阶段的网络，Feature指基于特征图的网络，Point指基于原始点云的网络。本次实验中，基于原始点云的网络与LiDAR R-CNN[14]设置相同，没有添加IoU分支，输出边界框的分数为分类分数。相比于单独使用第一阶段的网络，增加基于特征图的网络和基于原始点云的网络均可以提升性能，且增加基于原始点云的网络后性能更好，这说明原始点的精确位置信息确实对提升性能有很大帮助。在One stage+Feature的基础上，使用Point，性能还可以进一步得到提升，且高于One stage+Point，说明基于特征图的网络和基于原始点云的网络提升性能的原理不同，可以结合使用，以综合两者的优势。还进行了一个在One stage之后级联两个Point的实验，其在WOD验证集车辆类别上检测的性能不如One stag+Feature+ Point，说明在一阶段之后先使用基于特征图的网络再使用基于原始点云的网络是最佳配置。

表 4 Feature和Point不同搭配方案的结果

Table 4 Results of different matching schemes of Feature and Point unit: %

Scheme	L1		L2	
	3D AP	3D APH	3D AP	3D APH
One stage	66.59	66.09	61.79	61.33
One stage+Feature	67.28	66.79	62.43	61.98
One stage+Point	68.60	68.16	63.61	63.19
One stage+Point+Point	68.95	68.50	63.92	63.51
One stage+Feature+Point	**69.14**	**68.69**	**64.12**	**63.70**

3.3.3 基于原始点云的网络中IoU分支的效果

表5中，No IoU Branch指One stage + Feature + Point结构，With IoU Branch指在此基础上对基于原始点云的网络添加IoU分支的结构。表5中的结果显示，尽管三阶段网络已经使用了很多手段提升性能，添加IoU分支后，算法性能仍有进一步提升。这说明，同时包含分类信息和IoU信息的边界框的置信度分数对提升3D目标检测网络的性能有很好的促进作用。

表 5 使用IoU分支的效果

Table 5 Effect of using the IoU branch unit: %

Note	L1		L2	
	3D AP	3D APH	3D AP	3D APH
No IoU Branch	69.14	68.69	64.12	63.70
With IoU Branch	**69.29**	**68.86**	**64.26**	**63.86**

3.4 算法耗时

在NVIDIA A100上测试所提算法的耗时。一阶段网络处理单帧点云的平均耗时为31.2 ms，基于特征图的网络处理单帧点云的平均耗时为5.8 ms。以预选框及其对应的原始点云为输入，基于原始点云的网络并行处理256个预选框的平均耗时为2.7 ms。目前自动驾驶任务中使用的激光雷达的帧率通常为10 Hz，即每秒产生10帧点云，则相邻两帧点云的时间间隔为100 ms。相关感知算法处理单帧点云的耗时小于100 ms，可认为算法符合自动驾驶落地的耗时需求。由于模型部署过程中会有模型加速等工程化手段的使用，算法在实际车载芯片上的耗时通常比在NVIDIA A100上更少，所以所提算法完全有能力在100 ms之内完成关于单帧点云的3D目标检测任务，因此也就满足了自动驾驶落地的耗时需求。

4 结 论

提出了一个基于多阶段信息增强的3D点云目标检测网络。对网络结构的设计进行了探索，在基于体素化的单阶段3D目标检测网络之后，增加基于特征图的网络和基于原始点云的网络，均可以得到更加精准的目标边界框，两种网络结合使用可以综合两者的优势。基于特征图的网络在前，基于原始点云的网络在后的方案是最优搭配，由此所提算法的三阶段网络结构便形成了。在算法的设计过程中，特别注意对IoU信息的利用，为基于原始点云的网络添加了IoU分支，利用二阶段和三阶段网络对预测框和真值框的IoU进行预测，并在计算预测框的分数时均结合了分类信息和IoU信息。还对一阶段网络进行了改进，用计算IoU Loss的方式预测一阶段输出边界框的长宽高，并使用SiLU激活函数促进网络的收敛。所提算法在Waymo Open Dataset[17]上有不错的表现，检测精度高于备受工业界青睐的CenterPoint[10]，且耗时满足自动驾驶落地需求，有很大的工业应用潜力。但是，所提算法使用的特征抽取器都

比较简单，所以后面可以探索使用更强大的CNN或Transformer[22]来增强算法中特征抽取器的能力。

致谢 本文的大部分工作是在博世智能驾驶与控制事业部（以下简称博世）实习期间完成的。博世为本文的工作提供了丰富的GPU资源，本文作者与博世的同事唐亚哲、成昌昊、毕研广等进行了多次沟通与讨论，感谢上述同事提供的灵感与建议。

参 考 文 献

[1] 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.

[2] Charles R Q, Yi L, Su H, et al. Pointnet++: deep hierarchical feature learning on point sets in a metric space [C]//Proceeding of Neural Information Processing Systems, December 4-9, 2017, Long Beach, California, USA. Red Hook: Curran Associates, 2017.

[3] Shi 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, 2020: 770-779.

[4] 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, Republic of Korea. New York: IEEE Press, 2020: 1951-1960.

[5] Shi S, Wang Z, Shi J P, et al. From points to parts: 3D object detection from a point cloud with part-aware and part-aggregation network[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 43(8): 2647-2664.

[6] Qi C R, Litany O, He K M, et al. Deep Hough voting for 3D object detection in point clouds[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV), October 27-November 2, 2019, Seoul, Republic of Korea. New York: IEEE Press, 2020: 9276-9285.

[7] 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.

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

[9] 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.

[10] Yin T W, Zhou X Y, Krähenbühl P. Center-based 3D object detection and tracking[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 20-25, 2021, Nashville, TN, USA. New York: IEEE Press, 2021: 11779-11788.

[11] 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.

[12] Mao J G, Niu M Z, Bai H Y, et al. Pyramid R-CNN: towards better performance and adaptability for 3D object detection[C]//2021 IEEE/CVF International Conference on Computer Vision (ICCV), October 10-17, 2021, Montreal, QC, Canada. New York: IEEE Press, 2022: 2703-2712.

[13] Deng J, Shi S S, Li P W, et al. Voxel R-CNN: towards high performance voxel-based 3D object detection[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(2): 1201-1209.

[14] Li Z C, Wang F, Wang N Y. LiDAR R-CNN: an efficient and universal 3D object detector[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 20-25, 2021, Nashville, TN, USA. New York: IEEE Press, 2021: 7542-7551.

[15] Zheng W, Tang W, Chen S, et al. Cia-ssd: confident iou-aware single-stage object detector from a point cloud [J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(4): 3555-3562.

[16] Hu Y H, Ding Z, Ge R Z, et al. AFDetV2: rethinking the necessity of the second stage for object detection from point clouds[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(1): 969-979.

[17] Sun P, Kretzschmar H, Dotiwalla X, et al. Scalability in perception for autonomous driving: waymo open dataset [C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 2443-2451.

[18] Yang Z T, Zhou Y, Chen Z F, et al. 3D-MAN: 3D multi-frame attention network for object detection[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 20-25, 2021, Nashville, TN, USA. New York: IEEE Press, 2021: 1863-1872.

[19] Ngiam J, Caine B, Han W, et al. StarNet: targeted computation for object detection in point clouds[EB/OL]. (2019-08-29)[2022-10-08]. https://arxiv.org/abs/1908.11069.

[20] Zhou Y, Sun P, Zhang Y, et al. End-to-end multi-view fusion for 3D object detection in LiDAR point clouds[EB/OL]. (2019-10-15)[2022-10-08]. https://arxiv. org/abs/1910.06528.

[21] Ge R Z, Ding Z, Hu Y H, et al. AFDet: anchor-free one-stage 3D object detection[EB/OL]. (2020-06-23)[2022-10-08]. https://arxiv.org/abs/2006.12671.

[22] Vaswani A, Shazeer N M, Parmar N, et al. Attention is all you need[C]//Proceeding of Neural Information Processing Systems, December 4-9, 2017, Long Beach, California, USA. Red Hook: Curran Associates, 2017.