文章编号:1674-2974(2025)08-0080-12 DOI:10.16339/j.cnki.hdxbzkb.2025285

基于双重注意力融合的三维目标检测方法

雷志勇†
(国家能源集团陕西神延煤炭有限责任公司,陕西 榆林 719000)

摘 要:针对Voxel-RCNN算法在检测远处小目标以及受到遮挡的目标时检测精度不足的问题,提出了一种改进的方法,命名为CS-Voxel-RCNN.首先,通过引入随机顺序、随机丢弃和随机噪声三项数据增强方法,丰富了训练样本的多样性,从而增强了模型的鲁棒性.其次,通过在2D骨干网络中集成CBAM模块,运用通道注意力机制和空间注意力机制,对多尺度特征进行更为细致的处理,优化了特征融合效果.最后,通过新增DIoU损失分支,对原损失函数进行改进,着重强调目标边界框之间的距离信息,从而提高了目标边界框回归任务的准确性.在KITTI数据集上与一些经典的3D目标检测算法进行对比实验.结果表明,新提出的算法对比原Voxel-RCNN算法,在骑车者类的简单和困难级别上分别提升了2.91个百分点和0.87个百分点,并通过消融实验验证了各改进模块的有效性,这一系列改进方法在提高三维目标检测在现实场景中的实用性和准确性方面取得了积极的成果.

关键词:三维目标检测;激光雷达点云;数据增强;注意力机制;特征融合
中图分类号:TP391.4 **文献标志码**:A

A 3D Object Detection Method Based on Dual Attention Fusion

LEI Zhiyong†
(National Energy Group Shaanxi Shenyan Coal Co., Ltd., Yulin 719000, China)

Abstract: An improved method called CS-Voxel-RCNN is proposed to address the issue of insufficient detection accuracy of Voxel-RCNN algorithm in detecting small distant targets and occluded targets. Firstly, by introducing three data augmentation methods: random order, random dropout, and random noise, the diversity of training samples is enriched, thereby enhancing the robustness of the model. Secondly, by integrating CBAM in the 2D backbone network and utilizing channel attention mechanism and spatial attention mechanism, multi-scale features are processed in more detail, optimizing the feature fusion effect. Finally, by adding a DIoU loss branch, the original loss function is improved, emphasizing the distance information between the target bounding boxes, thereby improving the accuracy of the target bounding box regression task. Comparative experiments with some classic 3D object detection algorithms on the KITTI dataset are conducted. The results show that the newly proposed algorithm has significantly improved performance, compared with the original Voxel RCNN algorithm, with improvements of 2.91 percentage and 0.87 percentage for pedestrians and cyclists, respectively. The effectiveness of

* **收稿日期**:2024-01-31
基金项目:国家重点研发计划资助项目(2021YFB2501800),National Key Research and Development Program of China(2021YFB2501800)
作者简介:雷志勇(1974—),男,陕西榆林人,高级工程师
† 通信联系人,E-mail:1637551917@qq.com

each improvement module is verified through ablation experiments. This series of improvement methods achieve positive results in improving the practicality and accuracy of 3D object detection in real scenes.

Key words: 3D object detection; LiDAR point cloud; data augmentation; attention module; feature fusion

近年来,目标检测在自动驾驶环境感知领域扮演着越来越重要的角色,直接关系到车辆在复杂环境中的安全性[1].虽然二维的计算机视觉任务已取得了显著成就,但在实际场景中光照变化、天气条件和深度缺失等因素限制了仅仅依靠二维视觉感知的效果[2].由于激光雷达获取的三维数据不受自然光等条件限制,因而弥补了二维视觉领域存在的一些不足.因此,三维目标检测在自动驾驶领域具有极其重要的价值,是实现自动驾驶路径规划和安全避障的核心研究内容.

点云数据难以直接应用于3D目标检测,主要因为其非结构化、非固定大小、稀疏性、无序性、不规则形状和深度信息缺失等特点.这使得传统的计算机视觉方法难以有效处理点云数据,为应对这些挑战,研究者提出了基于深度学习的3D目标检测算法.根据在3D目标检测中处理点云数据时采用的不同表达方式,3D目标检测算法可分为基于点和基于体素两种主要的方法[3].基于点的3D目标检测方法通过直接处理激光雷达采集的点云数据,以点为基本单元进行目标检测.PointNet[4]首次根据点云的无序性采用了端到端的学习,实现了对点云数据的直接处理,然而其全局特征建模受限,难以捕捉目标的局部结构.为了改进这一问题,PointNet++[5]进一步考虑点云中的关键点均匀分布,提出了层次化的架构,从而更好地捕捉点云的多尺度特征.PointCNN[6]采用卷积操作处理点云数据,通过局部结构表示和局部旋转不变性提高了对点云的特征学习能力.PointRCNN[7]专注于三维目标检测,将二维目标检测框架成功应用于点云数据.3DSSD[8]以单次前向传播为特点,通过多尺度锚框设计和3D卷积实现了高效的目标检测.PolarNet[9]将点云鸟瞰图放入极坐标进行表示,以平衡每个网格中的点数,重新分配网络的注意力,使其关注极坐标系中径向轴上的长尾点分布.KPConv[10]提出了一种空间可变形点卷积,相比固定网格卷积具有更大的灵活性.KVGCN[11]设计了一种新的图卷积网络架构,该架构在构建的图的边缘上执行卷积以提取代表性的局部特征,然后使用局部特征描述符聚合成全局向量的技术来聚集局部和全局上下文特征.GACNet[12]设计了一种图注意力卷积网络,根据不同几何相邻点的特征属性为其分配适当的注意力权重,并由此确定特定的卷积核形状,从而更好地捕捉点云的结构特征,进行细粒度分割.

基于点的3D目标识别算法直接利用了原始点云数据,算法能够更好地捕捉目标的全局形状和结构信息.然而,直接处理上万个点的数据使得网络结构复杂,计算量庞大,对硬件要求较高,目前仍处于初步研究阶段,准确率相对基于体素的方法略有不足.

基于体素的方法将激光雷达感知空间划分为有序的固定尺寸的三维小网格,使用三维卷积神经网络进行特征提取.该方法在解决点云稀疏性和规则化带来的信息丢失问题上取得了显著的进展.在基于体素的方法中,VoxelNet[13]将点云划分为等间距的规则体素,并使用VFE(voxel feature encoder,体素特征编码)层将体素内点的特征量化统一.SECOND[14]网络采用3D稀疏卷积提取特征,相比VoxelNet加快了点云特征的提取速度,并降低内存消耗.PointPillars[15]网络将点云立柱化后转化为伪图像,通过二维卷积提取高维特征,大幅提高了算法的运行速度.在此基础上,PillarNeXt[16]从分配计算资源的角度重新审视了局部点聚合器,采用柱状网格编码器,并对2Dbackbone网络进行改进,利用ASPP、BiFPN等neck模块从主干中聚集特征,以扩大感受野和融合多尺度环境.PointPillars+[17]在原PointPillars框架的主干网络中加入协调注意力(CA)机制,能让网络模型专注图像中有效特征信息.STD[18]模型通过将稀疏的点云映射到密集的表示来提高检测性能,有效地捕捉目标的细节信息.MVF-Net[19]专注于多视角信息的融合,利用体素表示,通过融合多个视角的体素表示,提高了对点云数据的建模能力.Voxel-FPN[20]提出了基于体素的特征金字塔结构,编码器以自下而上的方式提取并融合多尺度体素信

息,而解码器则以自上而下的方式通过特征金字塔网络融合来自各种尺度的多个特征图,从而得到来自多尺度的特征信息.3D ShapeNets[21]采用了降低分辨率细粒度的方法来减少内存使用,使用了一种层次式的体素化方法,将三维空间表示为多个分辨率的体素网格,并使用多个卷积神经网络对不同分辨率的体素网格进行处理,但是这反而导致了点云信息的损失.VPFNet[22]是一种创新的3D对象检测架构,通过引入虚拟点来解决激光雷达点云与立体图像之间的分辨率不匹配问题,实现了更高效的数据融合和更高的检测精度,同时在计算效率上也有显著表现,能够在单个GPU上达到15帧/s的处理速度.MonoLiG[23]框架通过结合半监督学习和主动学习技术,利用LiDAR指导的跨模态教师-学生模型来训练单目3D对象检测器,并通过提出基于数据噪声的加权机制和传感器一致性选择策略,有效地提高了模型性能,同时大幅减少了所需的标注工作量.

Voxel-RCNN[24]基于此问题引入了创新性的Voxel RoI Pooling模块,作为一种高效的区域池化方法.该模块采用体素聚合(voxel query)方法,允许网络更有效地检索感兴趣区域(region of interest, RoI)周围的体素信息.同时,算法结合了加速的Point-Net++网络,旨在提升目标识别优化阶段中的局部体素特征提取效率.这种双重的优化策略不仅有效地提高了局部特征的表达能力,同时也显著缩短了整体网络的运行时间,因此Voxel-RCNN在车辆识别任务中表现出色.然而,由于其缺乏对不完整点云信息的处理方法,因此在检测远处目标以及受到遮挡的目标时算法性能有待提升.为解决这一问题,本文从原模型的数据增强模块、2D骨干网络以及损失函数这三个方面进行了系统性的改进,以期在更广泛的场景中提高检测性能.在改进的过程中,充分考虑了点云数据的特殊性,力求优化模型对不同目标的鲁棒性和泛化能力,从而使得新模型能够更好地适应复杂环境下的目标检测任务.

1 Voxel-RCNN目标检测模型

1.1 算法概述

Voxel-RCNN认为:对于高性能的3D目标检测,对原始点进行精确定位并非是不可或缺的,粗体素粒度同样能够达到卓越的检测精度.基于这一观点,构建了一个简单而高效的基于体素的目标检测网络,即Voxel-RCNN.通过在两阶段方法中充分利用体素特征的优势,Voxel-RCNN最终实现了与当时最先进的基于点的模型(如PV-RCNN[25])相媲美的检测精度,同时计算开销大幅减少.

如图1所示,Voxel-RCNN由三部分组成,分别是3D骨干网络、2D骨干网络+RPN以及Voxel RoI Pooling(体素感兴趣区域池化)+检测头.其特别设计了一个名为voxel RoI pooling的模块,该模块能够直接从体素特征中提取RoI特征,以便进行进一步处理.实验结果表明,在KITTI数据集和Waymo数据集上,Voxel-RCNN相较于现有的基于体素的方法,不仅能保持实时帧处理速率(即在NVIDIA RTX 2080Ti GPU上达到25 FPS的速率),而且提供了更高的检测精度.这一研究成果为基于体素的目标检测方法的发展提供了有力的支持,具有显著的实际应用潜力.

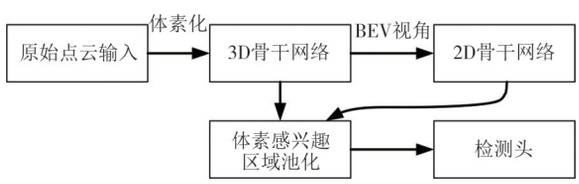


图1 Voxel-RCNN整体框架

Fig.1 Overview framework of Voxel-RCNN

1.2 3D骨干网络设计

Voxel-RCNN在构建其3D骨干网络时采用了由SECOND网络提出的创新性3D稀疏卷积方法.这一设计旨在应对直接进行3D卷积所带来的巨大计算量,通过充分利用点云特征的稀疏性,SECOND提出了一种高效的3D稀疏卷积方法,从而显著提升了3D卷积操作的执行速度.Voxel-RCNN利用SECOND网络提出的3D稀疏卷积作为其3D骨干网络.

对于3D点云数据,其通常以稀疏张量的形式进行存储,其中只有一小部分位置上存在非零值.具体而言,普通的卷积操作会对所有输入点进行响应,而稀疏卷积则仅考虑稀疏数据中的非零值.在进行卷积操作时,需要计算输入和输出位置的索引.如图2所示,稀疏卷积操作将输入数据转换为输入哈希表,该哈希表明确存储了输入点云中非零值的坐标.通过构建规则表和输出哈希表,能够将卷积计算前后的数据相互匹配.通过计算得到的索引,执行卷积计算,其中包括在输入和卷积核的非零位置上进行乘法运算,并将结果累加到输出的相应位置.由于涉及

的位置相对较少,这种高效的方式相对于普通卷积更加迅速.

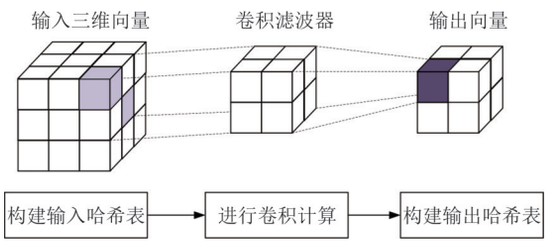


图 2　3D 稀疏卷积过程

Fig.2　3D sparse convolution process

这一创新性的3D稀疏卷积方法为Voxel-RCNN的性能提升提供了有效的技术手段.通过引入并行计算策略,Voxel-RCNN在3D目标检测任务中取得了令人瞩目的成果,在保持实时帧处理速率的同时,极大地提高了检测精度.

1.3　2D骨干网络和RPN设计

网络将经过稀疏卷积后的3D特征图在z轴上进行压缩,得到BEV(bird's eye view)特征图作为2D骨干网络的输入,在如图3所示的卷积神经网络(convolutional neural network, CNN)中进行进一步的特征提取.

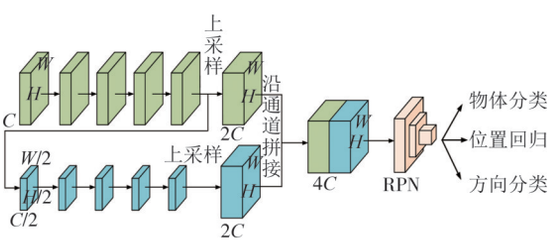


图 3　2D骨干网络和RPN过程

Fig.3　2D backbone and RPN process

具体来说,CNN分为两个分支进行下采样来得到两个不同尺度的特征图,从而提取输入数据在空间层面的不同细粒度信息.较低尺度的特征图具有较大的感受野,能够捕获全局信息,但对于局部细节的表示较差.相反,较高尺度的特征图的感受野较小,其更注重捕获局部细节,有助于提高对细微特征的敏感性.因此,在下采样之后通过反卷积操作将得到的特征图统一到相同维度,最后在通道方向进行拼接,得到全局特征图,从而实现多尺度的特征图的聚合.这样的多尺度特征图聚合策略在目标检测任务中综合考虑了全局和局部信息,以提高模型对不同尺度和结构目标的识别性能.具体的卷积操作见表1.

表 1　2D骨干网络的卷积操作

Tab.1　Convolution operations in 2D backbone networks

阶段	重复次数	卷积核大小	步长	输出通道
下采样层1	5	3	1	128
上采样层1	1	1	1	256
下采样层2	5	3	2	256
上采样层2	1	2	2	256

随后的区域生成网络使用了Faster RCNN[26]首先提出的区域生成网络(region proposal network, RPN).RPN网络通过在卷积特征图上进行滑动,生成多个候选区域,可能包含目标物体.每个候选区域都经过RPN评分计算,选择高置信度的候选框.滑动窗口生成一系列预定义的锚框,这些锚框具有多样的尺度和长宽比,以确保覆盖各种目标的形状和大小,从而提高RPN的通用性.这些锚框的固定位置和比例是通过在训练数据集上的分析和统计得到的.一个图像中的每个点通常对应多个锚框,这使得RPN能够有效地检测不同大小的目标,为后续目标检测提供了多样性的候选框.

RPN由小型卷积网络组成,包括卷积层和两个并行的全连接层,用于锚框分类和边界框回归.卷积层提取卷积特征图中的特征,全连接层生成每个锚框的得分和回归参数.

在每个滑动窗口位置,RPN对每个预定义的锚框执行二分类和回归操作,即目标存在的概率(objectness score)和修正Anchor Box的边界框偏移(bounding box regression).每个锚框生成两个分数,表示该锚框包含目标和不包含目标的概率,同时生成用于调整边界框位置的回归参数.RPN通过这一系列操作为后续目标检测提供了候选框的精选.生成的所有候选框(包括正样本和部分负样本)通过非极大值抑制(non-maximum suppression, NMS)进行筛选.NMS排除高度重叠的候选框,保留具有较高目标存在概率的框.

最终,RPN输出的是一系列生成的候选区域,每个区域都有一个与之关联的目标存在概率和边界框回归的信息.这些候选区域会被送入后继的RoI池化层,以便进一步进行目标分类和边界框回归.

1.4　Voxel RoI Pooling+检测头设计

Voxel RoI Pooling是算法的第二阶段,旨在对第一阶段提出的候选区域进行进一步的优化.一般的优化手段是采用RoI Pooling等技术,将所有候选区域分割成固定维度,以方便后续的并行计算.接着,

3D体素特征被映射到候选区域内,过滤掉非候选区域的特征,重新生成新的候选区域3D体素.每个候选区域被看作一个独立的体素空间,进行特征提取,最终微调第一阶段的初步候选区域.

Voxel-RCNN算法提出了Voxel RoI Pooling方法,其运用了voxel query方法和加速的PointNet++方法,以提高体素特征映射和优化的效率.其具体流程如下:

1)将初步候选区域划分为固定大小的小网格,计算每个网格的中心位置以作为特征采样点.

2)以特征采样点和非空体素的中心位置为基础计算它们之间的曼哈顿距离,选取至每个特征采样点曼哈顿距离最小的K个体素,如图4所示,使用voxel query方法进行体素选取,计算规定范围内的体素坐标并在非空体素坐标中进行索引.此方法相较于传统的球查询(ball query)方法,提高了体素查询的效率.

3)在体素特征的聚合过程中,采用了一种加速的PointNet++方法,以便快速地聚集从采样点邻域中选取的所有非空体素特征,对每个候选区域的所有网格均使用以上方法进行特征映射,有效提升了操作的效率.

4)完成特征映射后,针对候选区域网格的特征,利用两个全连接层进行进一步的二次特征提取.接着,采用两个由全连接层组成的分支,分别生成目标类别的置信度和优化回归的边界框的优化结果.这一多阶段的特征提取和优化过程有助于在保持高效性的同时,获取更准确的目标检测结果.

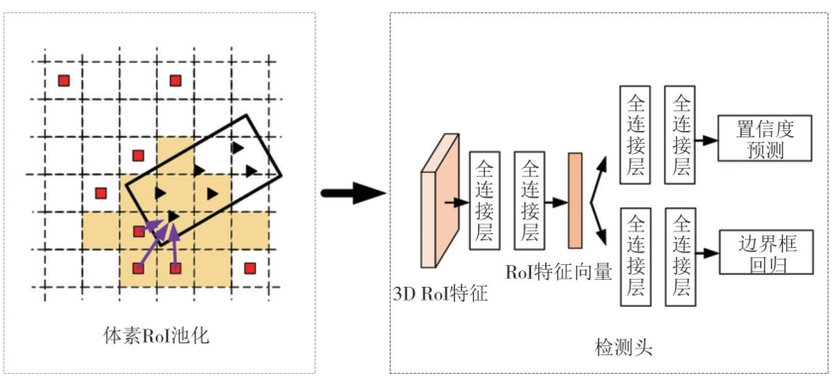


图4 体素RoI池化和检测头设计

Fig.4 The design of voxel RoI pooling and detect head

2 基于改进Voxel-RCNN算法的道路目标检测方法

2.1 数据增强方法

在对点云数据的增强策略方面,本文在原有的方法基础上引入了三项新的策略,分别是随机顺序、随机丢弃和随机噪声,以进一步丰富数据样本的多样性,提供更全面的信息采样和更广泛的变异性,为模型提供更具挑战性和真实性的训练样本,从而在处理小目标和受到遮挡目标的检测问题上取得更好的改进效果.

2.1.1 随机顺序

随机顺序在点云数据处理中的作用是通过随机打乱点云或者点云中的点的顺序,以增加训练样本的多样性,从而提高模型的鲁棒性和泛化能力.该操作涵盖两个主要方面:

首先,通过整体打乱点云批次及其对应标签的顺序,采用生成索引数组,并对其进行随机打乱的策略.接着,利用打乱后的索引,对原始数据和标签进行重新排列,从而实现整个批次的随机顺序.

其次,通过在单个点云中随机打乱点的顺序,同样采用生成点的索引数组并对其进行随机打乱的策略.然后,根据打乱后的索引,重新排列原始点云数据,达到在点云内部引入随机顺序的目的.这一过程旨在在训练中引入更多的数据变化,以使模型更好地适应不同的数据分布,提升其性能表现.

2.1.2 随机丢弃

随机丢弃是通过在点云中随机去除部分点,模拟点云数据的不完整性.首先,针对每个点云批次,随机生成一个丢弃概率,用于确定是否进行丢弃.其次,通过比较随机生成的值与设定的最大丢弃比例,确定需要丢弃的点的索引.如果存在需要丢弃的点,将这些点的数值设置为该点云中第一个点的数值,

实现伪丢弃.最后返回具有随机丢弃效果的点云批次.

2.1.3　随机噪声

随机噪声是通过对点云数据进行小范围扰动,以模拟真实场景中的噪声.该操作的主要作用是对每个点进行随机偏移,引入小范围内的随机扰动.该操作采用正态分布生成随机噪声,并将其应用于每个点,随后通过限制噪声值在预定的上下限范围内,将噪声添加到原始点云数据中.这样的处理有助于使模型更好地适应真实世界中的各种噪声情况,提升模型对不同噪声环境的处理能力.

通过引入上述三种数据增强方式,模型在解决小目标检测和被遮挡目标检测方面取得了显著改进效果.随机顺序的引入有助于模型更好地处理点云中小目标的信息.点云顺序被随机打乱,样本多样性得到提高,进而提高模型的泛化能力.随机丢弃策略通过模拟真实场景中的不完整点云,让模型能够更好地处理受遮挡目标的情况,提高了被遮挡目标检测的效果.而引入随机噪声则通过模拟真实噪声环境,增强了模型对噪声的鲁棒性,使其更适应复杂场景,进一步提升了小目标和被遮挡目标检测的性能.这三种方法的综合应用为解决小目标检测和被遮挡目标检测问题提供了有力的解决途径,为模型在复杂场景下的实际应用提供了可靠的支持.

2.2　CBAM模块

在CNN中,卷积运算将跨通道信息和空间信息相混合来提取特征,因此过程中有可能损失这两个维度的某些特征信息.CBAM(convolutional block attention module,卷积层注意力模块)[27]是一种注意力机制,旨在通过自适应地学习通道和空间注意力来提升卷积神经网络的性能.CBAM主要由两个模块组成:通道注意力模块(channel attention module,CAM)和空间注意力模块(spatial attention module, SAM).通道注意力模块通过学习各个通道之间的关联性,以提升有价值的通道特征;而空间注意力模块则旨在优化不同空间位置之间的关系,以提升空间特征的辨识度.这一综合的结构使得CBAM能通过自适应地调整通道和空间的关注度,进而使网络更加聚焦于关键特征,从而显著提升了卷积神经网络在图像识别、目标检测等领域的性能.

CBAM依次对输入特征图应用CAM和SAM模块,假设输入特征向量为:$\boldsymbol{F} \in \mathbf{R}^{C\times H\times W}$;经过CAM模块后,得到一维通道强化向量:$\boldsymbol{M}_{\mathrm{c}} \in \mathbf{R}^{C\times 1\times 1}$,该过程用公式表示为

$$\boldsymbol{F}' = \boldsymbol{M}_{\mathrm{c}}(\boldsymbol{F}) \otimes \boldsymbol{F} \tag{1}$$

经过SAM模块后,得到二维空间强化向量:$\boldsymbol{M}_{\mathrm{s}} \in \mathbf{R}^{1\times H\times W}$,过程表示为:

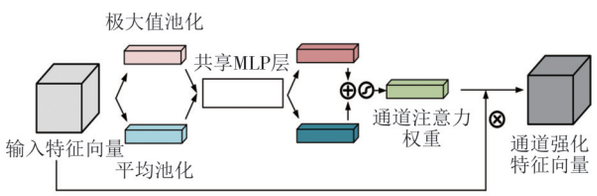
$$\boldsymbol{F}'' = \boldsymbol{M}_{\mathrm{s}}(\boldsymbol{F}') \otimes \boldsymbol{F}' \tag{2}$$

2.2.1　通道注意力模块

通道注意力模块旨在通过深入学习各个通道之间的内在关联性,从而提升具有价值的通道特征的辨识和表达.如图5所示,通道注意力模块首先对输入进行全局平均池化,以捕捉整体特征趋势.接下来,通过一系列全连接层运算,生成通道注意力权重,这些权重对各通道的重要性进行了准确而智能的评估.最终根据计算的权重,通道注意力模块对输入通道进行加权操作,以凸显对任务至关重要的通道特征.全过程可用公式表示为

$$\boldsymbol{M}_{\mathrm{c}}(\boldsymbol{F}) = \sigma(\mathrm{MLP}(\mathrm{AvgPool}(\boldsymbol{F})) + \mathrm{MLP}(\mathrm{MaxPool}(\boldsymbol{F}))) \tag{3}$$

式中:σ为Sigmoid函数.这种关注度机制使得网络更加聚焦于重要的通道信息,提升了对关键特征的感知和利用能力.



⊕: 拼接操作　⊘: Sigmoid函数激活操作　⊗: 矩阵相乘操作

图5　通道注意力模块设计

Fig.5　The design of channel attention module

2.2.2　空间注意力模块

空间注意力模块的目标在于提取学习不同空间位置之间的内在关系,从而提升空间特征的辨识和表达水平.如图6所示,首先利用两个分支的卷积操作,精准捕捉了数据在水平和垂直方向上的关联性,使模型能够更深刻地理解空间结构.在卷积操作的基础上,利用Sigmoid函数生成空间注意力权重.这些权重充分体现了各个空间位置的相对重要性,使得网络能够更智能地关注对任务至关重要的局部空间区域.最终,空间注意力模块通过将这些计算得到的权重应用到输入特征上,实现对空间特征的细致加权调整.全过程可用公式表示为

$$\boldsymbol{M}_{\mathrm{s}}(\boldsymbol{F}) = \sigma(f(\mathrm{AvgPool}(\boldsymbol{F});\ \mathrm{MaxPool}(\boldsymbol{F}))) \tag{4}$$

式中:σ为Sigmoid函数;f为卷积计算操作.这一过程不仅提升了对空间信息的关注度,同时也为网络提供了更好地捕捉和利用数据中的关键空间特征的手段.

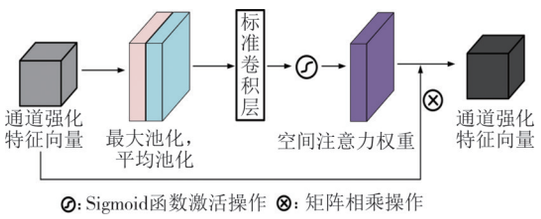


图6 空间注意力模块设计

Fig.6 The design of spatial attention module

综上所述,CBAM通过通道注意力模块和空间注意力模块的融合,在通道和空间两个维度上深入挖掘关联性,使得网络能够更全面、更敏锐地理解和利用输入数据.CBAM的引入为卷积神经网络注入了灵活性,使其能够智能地感知和适应关键特征,从而显著提升了神经网络的表征能力,进一步推动了深度学习在图像处理、计算机视觉等领域的性能提升.

2.3 CBAM模块与Voxel-RCNN算法的结合

在Voxel-RCNN算法的2D骨干网络中,经过两个尺度的卷积操作后,采用上采样到相同尺度后直接拼接的操作,然而,这种策略存在一些潜在的弊端.首先,直接拼接可能引起信息的冗余,因为不同尺度的特征图可能携带相似的信息,缺乏有效的筛选和融合机制.其次,这种简单的拼接操作或许难以充分挖掘不同尺度特征之间的复杂关系,进而限制了网络对目标的高级表达能力.再次,对于大型和小型物体同时存在的场景,不同层次的特征可能存在冲突,从而制约了特征的有效融合.这些弊端可能导致网络在处理多尺度特征时存在局限性,影响了对场景复杂性的全面理解.

为解决这些问题,本文提出通过CBAM模块对输出的特征图进行进一步操作.如图7所示,输入特征首先采用通道注意力模块,通过全局平均池化层获得每个通道的平均响应值,然后通过两个全连接层生成通道注意力权重.这些权重通过对原始特征图进行逐通道加权求和,从而使网络更集中地关注对任务更关键的通道,提高对不同通道之间关联性的感知能力,进而使得网络更加集中地关注重要的通道,以提高对不同通道间关联性的感知能力.随后再将通道强化后的特征输入空间注意力模块,通过对特征图进行全局最大池化和全局平均池化,获得每个通道的最大值和平均值.将其输入全连接层以生成空间注意力权重,然后将这些权重应用于原始特征图,使网络能够更好地捕捉不同空间位置的关键信息.通过这种方式,CBAM模块能够提升对目标的通道关联性和空间分布的感知,从而增强整体特征表达的能力.

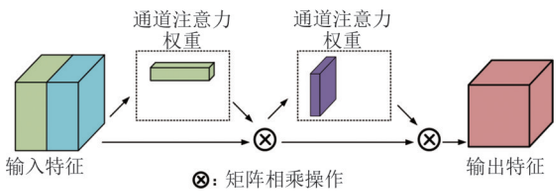


图7 CBAM模块的总体结构

Fig.7 The overview structure of CBAM

CBAM模块的引入使得Voxel-RCNN算法在特征聚合阶段更为精细地处理了多尺度特征,有效优化了多尺度特征的融合效果.通过引入通道注意力机制和空间注意力机制,CBAM模块能够更有针对性地提升对不同尺度特征的感知能力,使得网络更全面、准确地理解目标的结构和多尺度信息.这一优化在提升目标检测性能方面具有显著的作用,使Voxel-RCNN在处理复杂场景和多尺度目标时表现更为出色.

2.4 损失函数设计

2.4.1 原算法损失函数

Voxel-RCNN在定义损失函数时分为分类层损失(classification layer loss)、边界框回归损失(box regression layer loss)和角度分类损失(direction classification loss),分别用L_{cls}、L_{reg}和L_{dir}来表示.

分类损失L_{cls}采用了焦点损失(focal loss,FL)函数,Focal loss函数是一种用于解决类别不平衡问题的损失函数.在3D目标检测任务中,负样本通常远多于正样本,导致模型更容易过度偏向负样本,而无法有效学习正样本的特征.Focal loss函数通过降低易分类的负样本的权重,将关注点更集中在难以分类的样本上,从而提高对难样本的敏感性.其公式为

$$\mathrm{FL}(p_t) = -(1 - p_t)^{\gamma} \cdot \log p_t \tag{5}$$

式中:p_t是模型对样本的预测概率;γ是调节因子,用于调整难易样本的权重.当样本被错误分类时,$(1 - p_t)^{\gamma}$的值增大,使得损失更关注于难以分类的样本.这种损失函数在训练中能够有效缓解类别不平衡问题,提高模型对困难样本的识别能力,从而改善目标检测的性能.

边界框回归损失 L_{reg} 只针对前景的锚框进行损失计算,此处采用 Huber loss 函数,又名平滑 L1 损失(Smooth L1 Loss)函数,其主要特点在于对小误差具有鲁棒性,通过在接近零的区域采用二次函数,减缓损失函数的梯度.在面对大误差时,损失函数呈线性增长,相比其他损失函数更具抗干扰能力.其公式为

$$\text{Smooth L1 Loss}(x)=\begin{cases}0.5x^2, |x|<1\\ |x|-0.5, \text{其他}\end{cases} \tag{6}$$

式中:x 为模型预测值和真实值之间的差异.

角度分类损失 L_{dir} 用于优化模型物体在 3D 空间中的朝向进行分类,增强对目标在三维空间中的姿态理解.采用了交叉熵损失(cross-entropy loss)函数,其用于测量模型输出的概率分布与真实标签之间的差异,帮助优化模型以提高分类性能.公式为

$$L(x,y)=-\sum_{i=1}^{C}x_i\log y_i \tag{7}$$

式中:x_i 表示真实标签的第 i 个元素;y_i 表示模型预测 x 属于第 i 个类别的概率.

2.4.2　损失函数改进

为了提高算法对于小目标和受阻挡目标的检测效果,本文在原基础上引入了一个新分支,对预测边界框的位置回归进行进一步细化.采用了 DIoU(distance intersection over union,距离交并比)损失,该损失函数考虑了边界框之间的距离信息,通过提高 3D 预测边界框和真实边界框的交并比来提高回归任务的效果.

交并比(intersection over union,IoU)是目标检测中常用的一种性能评价指标.它衡量了预测边界框(bounding box, bbox)与真实边界框(ground truth box,gt box)之间的重叠程度,通常用于衡量检测算法的准确性.公式如下:

$$\text{IoU}(A,B)=\frac{\text{Area}(A\cap B)}{\text{Area}(A\cup B)} \tag{8}$$

式中:$A\cap B$ 表示预测边界框和真实边界框的交集;$A\cup B$ 表示它们的并集.IoU 的取值范围在 0~1 之间,其中 0 表示没有重叠,1 表示完全重叠.

然而,传统的 IoU 并没有考虑到边界框之间的空间位置关系,因此在处理较大的空间错位、较小目标或遮挡等情况下,IoU 的表现可能受到影响.文献[28]提出了 DIoU 损失函数,旨在解决 IoU 难以处理边界框间距离信息的问题,特别是当存在较大的空间错位或较小目标时.DIoU 损失函数被定义为

$$\text{DIoU}=\text{IoU}-\frac{\rho^2(b,b^{gt})}{c^2} \tag{9}$$

其各参数在 3D 目标检测时的含义如图 8 所示,红色框、黑色框和虚线框分别为真实边界框、预测边界框和覆盖预测框与目标框的最小封闭框.b, b^{gt} 分别代表了预测边界框和真实边界框的中心点.c 代表的是两个中心点间的欧式距离.ρ 代表的是能够同时包含预测边界框和真实边界框的最小外接矩形的对角线长度.

DIoU 损失函数的定义为:

$$L_{DIoU}=1-\text{IoU}+\frac{\rho^2(b,b^{gt})}{c^2} \tag{10}$$

DIoU 通过考虑边界框中心点距离和对角线长度,直接优化预测边界框和真实边界框之间的欧式距离,此过程可以为预测边界框提供移动方向,使得回归更加稳定,收敛更加快速.在处理小目标和缺失点云信息的目标检测场景中,DIoU 损失函数更全面地考虑了空间位置信息.由于小目标容易受到定位误差的影响,DIoU 的引入使得模型能够更准确地衡量目标边界框之间的位置关系,有助于提高目标检测算法在边界框回归方面的性能.在这些挑战性的情况下,DIoU 的综合性能优势能够帮助算法更好地适应各种目标形状和尺寸变化,进而提升检测精度.

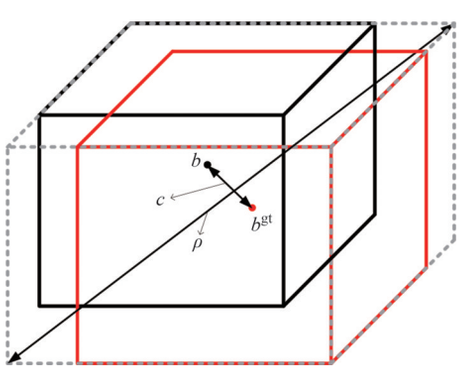


图 8　3D DIoU 示意图

Fig.8　3D DIoU schematic diagram

3　实验与结果分析

3.1　测试数据集

本实验数据集使用由德国卡尔斯鲁厄理工学院和丰田美国技术与创新研究院合作创建的 KITTI[29] 公开数据集,该数据集包含了丰富多样的城市驾驶

场景,其中包括高分辨率的图像、激光雷达点云、相机标定参数和车辆运动轨迹等信息.KITTI数据集主要用于评估和推动各种计算机视觉任务,如目标检测、语义分割、立体匹配以及道路线追踪等,尤其在自动驾驶领域发挥着重要作用.由于其真实的城市驾驶场景和详细的标注,KITTI数据集成为许多研究和算法验证的标准基准,共有7 481个训练样本和7 518个测试样本,通常训练样本会被进一步分为含3 712个样本的训练集和含3 769个样本的验证集.KITTI数据集包含车辆、行人和骑车者三类目标类别,且根据检测目标尺寸、被遮挡情况、截断程度、距离远近等指标分为简单、中等、困难三种级别.

3.2 实验设置

实验算法基于OpenPCDet目标检测框架实现,主板为Inspur YZMB-00882-10F,中央处理器(CPU)为Intel(R) Xeon(R) Silver 4210R,显卡使用了Geforce RTX 4090.操作系统为Ubuntu22.04,CUDA版本为11.7,深度学习框架为Pytorch1.10.0.训练超参数和优化相关的配置如下:epoch设置为80,batch-size设置为16,学习率设置为0.01,权重衰减为0.01,动量为0.9.

实验采用交并比(IoU)指标来评估目标是否被有效检测,IoU的取值范围在0~1之间,其中1表示完美匹配,0表示未匹配.其中车辆类别的IoU阈值设定为0.7,行人类别和自行车类别的IoU阈值均设定为0.5.

在体素划分的过程中,由于KITTI数据集只提供在相机观察到的场景范围中的目标的注释信息,故先对点云范围进行裁剪.具体来说,x轴的范围为[0, 70.4] m,y轴的范围为[-40, 40] m,z轴的范围为[-3, 1] m.输入体素的大小设置为(0.05 m, 0.05 m, 0.1 m),从而将原始点云输入规范化.

3.3 评价指标

为了更好地与原始算法进行对比,改进后的Voxel-RCNN算法选择了与原算法相同的评价策略和评价指标.这种一致性的选择有助于在性能评估过程中保持公平性和可比性,从而更准确地评估改进的算法在不同任务中的性能提升.

在目标检测的二分类问题中,通常使用真正例(true positive, TP)、真负例(true negative, TN)、假正例(false positive, FP)和假负例(false negative, FN)这些术语来描述模型的预测表现.TP表示模型正确地预测了正例,TN表示模型正确地预测了负例,FP表示模型错误地将负例预测为正例,而FN表示模型错误地将正例预测为负例.预测结果分类如表2所示.

表2 预测结果分类表

Tab.2 Classification table of prediction results

是否预测正确	Positive	Negative
True	TP	TN
False	FP	FN

在目标检测中,精度(precision, P)和召回率(recall, R)是两个关键的性能指标.P表示模型预测为正例中有多少是真正例,而R则表示所有真正例中有多少被模型成功预测为正例.计算公式分别为

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{11}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{12}$$

实验采用平均精度(mean average precision, mAP)来衡量模型效果.mAP是目标检测任务中常用的性能评估指标之一,综合考虑了模型在不同类别上的精度,并通过计算各类别的平均值来提供对整体性能的度量.通过在不同阈值下计算P和R,并绘制P-R曲线,可以得到每个类别的精度信息.mAP就是对这个曲线下面积的平均值,即平均精度.计算公式为

$$\mathrm{mAP} = \frac{1}{N}\sum_{i=1}^{N}\mathrm{AP}_i \tag{13}$$

在验证集上的测试实验中,使用了11点插值法对模型在单个分类上的目标识别AP进行估算;而在测试集上的评测实验中,采用了40点插值法来估算模型在单个分类上的目标识别AP.为了全面评估模型在多类别物体识别任务中的性能,使用各个类别上的平均预测精度mAP作为衡量标准.

3.4 不同算法对比实验

3.4.1 验证集结果

由于早期在验证集上的测试实验往往采用提出的11点插值法对模型进行评估,故为保证与其他算法形成公平的对照,使用了11点插值下3D视角的平均精度进行计算.

如表3所示,改进算法在骑车者(cyclist)类的简单和困难级别上相较原算法分别提升了2.91个百分点和0.87个百分点.

表 3　不同算法在 KITTI 验证集上的检测精度结果对比

Tab.3　Comparison of detection accuracy results of different algorithms on the KITTI validation set

方法	$P_{汽车}$/%			$P_{行人}$/%			$P_{骑车者}$/%		
	简单	中等	困难	简单	中等	困难	简单	中等	困难
SECOND[14]	90.55	81.61	78.61	55.94	51.14	46.16	82.96	66.74	62.78
PointPillars[15]	87.75	78.39	75.18	57.30	51.41	46.87	81.58	62.94	58.98
VPFNet[22]	91.82	80.57	78.08	61.69	54.97	48.47	91.64	73.26	68.78
MonoLiG[23]	92.10	84.36	82.48	62.17	54.49	49.88	89.10	70.38	66.02
PillarNeXt[16]	90.23	83.57	80.66	60.20	56.28	45.63	82.61	71.87	65.59
Voxel-RCNN[24]	89.22	79.08	78.36	62.75	56.03	50.76	82.05	71.09	67.14
本文算法	88.95	78.78	78.00	60.79	52.93	48.85	84.93	70.78	68.01

3.4.2　测试集结果

在测试集上的实验,采用了 40 点插值法进行评估,结果如表 4 所示.

图 9 展示了算法改进前后边界框可视化结果,可直观地看出改进后(右图)相比原算法(左图)更准确地检测到了点云信息缺失的车辆目标和两个行人目标.

表 4　不同算法在 KITTI 测试集上的检测精度结果对比

Tab.4　Comparison of detection accuracy results of different algorithms on the KITTI test set

方法	$P_{汽车}$/%			$P_{行人}$/%			$P_{骑车者}$/%		
	简单	中等	困难	简单	中等	困难	简单	中等	困难
SECOND[14]	88.64	79.00	75.81	51.77	46.20	41.12	75.83	60.82	53.67
PointPillars[15]	86.82	76.15	73.06	51.71	45.14	40.89	77.10	58.65	52.92
VPFNet[22]	91.82	80.57	78.08	61.69	54.97	48.47	91.64	73.26	68.78
MonoLiG[23]	90.25	81.43	76.82	61.80	53.37	48.63	78.60	63.71	57.65
PillarNeXt[16]	90.21	81.00	77.63	63.71	50.26	48.34	80.62	71.19	64.96
Voxel-RCNN[24]	92.06	82.84	80.25	62.52	54.61	49.06	84.46	71.32	67.05
本文算法	91.66	82.25	79.84	60.37	53.05	48.03	86.82	71.05	67.45

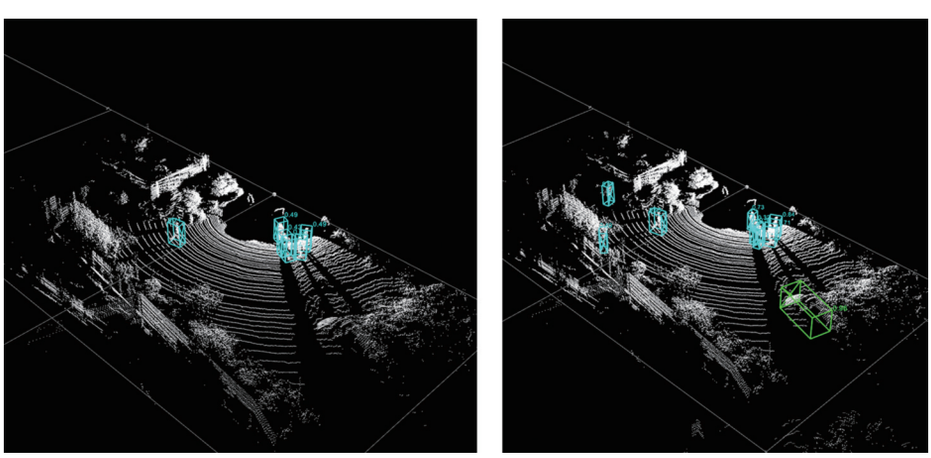


图 9　3D 可视化结果对比

Fig.9　Comparison of 3D visualization results

3.5　消融实验

在原 Voxel-RCNN 算法的基准上,通过结合本文提出的不同的改进模块,在 KITTI 验证集上测试各模块的有效性.表 5 展示了各模块对检测效果的影响,使用了在三个目标类别上的平均精度 mAP 来进行评价.

表 5　不同模块的测试结果表

Tab.5　Test results for different modules

CBAM	DIoU	mAP
		70.72
√		70.89
	√	71.22
√	√	71.56

4 结 论

本文通过对Voxel-RCNN算法进行深入改进,专注于解决其在小目标检测方面存在的不足之处.引入了随机顺序、随机丢弃和随机噪声三项数据增强方法,这些方法丰富了训练样本的多样性,提供了更全面的信息采样,增强了模型的鲁棒性.在2D骨干网络中嵌入CBAM模块,以更精细地处理多尺度特征.通过通道注意力机制和空间注意力机制,分别关注重要的通道和不同空间位置的关键信息,从而优化了特征融合的效果.特别是对于多尺度特征的处理,CBAM模块使得Voxel-RCNN算法在特征聚合阶段更为精细地处理多尺度特征,增强了对目标结构和多尺度信息的理解,从而提升了目标检测性能.同时,通过新增DIoU损失分支,对原损失函数进行改进,强调了目标边界框之间的距离信息,有助于提高模型在目标边界框回归任务中的准确性.

实验在KITTI数据集上进行评估,并与多个经典方法进行对比.结果表明,改进后的Voxel-RCNN算法在小目标检测方面表现优越,超过了一系列经典方法.消融实验也验证了引入的三种数据增强方法和DIoU损失分支的有效性.这一系列改进的算法和方法对于提高三维目标检测在现实场景中的实用性和准确性具有积极的推动作用,为相关领域的研究和应用贡献了有价值的成果.

参考文献

[1] 王亚东, 田永林,李国强,等. 基于卷积神经网络的三维目标检测研究综述[J]. 模式识别与人工智能, 2021, 34(12): 1103-1119.
WANG Y D, TIAN Y L, LI G Q, et al. 3D object detection based on convolutional neural networks: a survey [J]. Pattern Recognition and Artificial Intelligence, 2021, 34(12): 1103-1119.(in Chinese)

[2] 解则晓, 李美慧. 机器学习在基于点云的三维物体识别领域的研究综述[J]. 中国海洋大学学报(自然科学版), 2021, 51(6): 125-130.
XIE Z X, LI M H. A survey on machine learning in recognition of 3D object based on point cloud[J]. Periodical of Ocean University of China, 2021, 51(6): 125-130.(in Chinese)

[3] FERNANDES D, SILVA A, NÉVOA R, et al. Point-cloud based 3D object detection and classification methods for self-driving applications: a survey and taxonomy [J]. Information Fusion, 2021, 68: 161-191.

[4] 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). Honolulu, HI, USA. IEEE, 2017: 77-85.

[5] QI C R, YI L, SU H, et al. PointNet++ [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach, California, USA. ACM, 2017: 5105-5114.

[6] LI Y Y, BU R, SUN M C, et al. PointCNN: Convolution on x-transformed points [C]//NIPS' 18: Proceedings of the 32nd International Conference on Neural Information Processing Systems. Montréal, Canada. 2018.

[7] VORA S, LANG A H, HELOU B, et al. PointPainting: sequential fusion for 3D object detection[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA. IEEE, 2020: 4604-4612.

[8] YANG Z T, SUN Y N, LIU S, et al. 3DSSD: point-based 3D single stage object detector[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA. IEEE, 2020: 11040-11048.

[9] ZHANG Y, ZHOU Z X, DAVID P, et al. PolarNet: an improved grid representation for online LiDAR point clouds semantic segmentation [C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Seattle, WA, USA. IEEE, 2020: 9598-9607.

[10] THOMAS H, QI C R, DESCHAUD J E, et al. KPConv: flexible and deformable convolution for point clouds[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South). IEEE, 2019: 6411-6420.

[11] LUO N, YU H Q, HUO Z F, et al. KVGCN: a KNN searching and VLAD combined graph convolutional network for point cloud segmentation[J]. Remote Sensing, 2021, 13(5): 1003.

[12] WANG L, HUANG Y C, HOU Y L, et al. Graph attention convolution for point cloud semantic segmentation [C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA. IEEE, 2019: 10288-10297.

[13] 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. Salt Lake City, UT, USA. IEEE, 2018: 4490-4499.

[14] YAN Y, MAO Y X, LI B. SECOND: sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10): 3337.

[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). Long Beach, CA, USA. IEEE, 2019: 12689-12697.

[16] LI J Y, LUO C X, YANG X D. PillarNeXt: rethinking network designs for 3D object detection in LiDAR point clouds[C]//2023

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Vancouver, BC, Canada. IEEE, 2023: 17567-17576.

[17] SSINGH A R. PointPillars++ : An encoder for 3-D object detection and classification from point clouds[D]. North Carolina State University, 2021.

[18] 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). Seoul, Korea (South). IEEE, 2019: 1951-1960.

[19] WU F Z, BAO L C, CHEN Y J, et al. MVF-Net: multi-view 3D face morphable model regression[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, USA. IEEE, 2019: 959-968.

[20] 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.

[21] WU Z R, SONG S R, KHOSLA A, et al. 3D ShapeNets: a deep representation for volumetric shapes[C]//2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA. IEEE, 2015: 1912-1920.

[22] ZHU H Q, DENG J J, ZHANG Y, et al. VPFNet: improving 3D object detection with virtual point based LiDAR and stereo data fusion[J]. IEEE Transactions on Multimedia, 2022, 25: 5291-5304.

[23] HEKIMOGLU A, SCHMIDT M, MARCOS-RAMIRO A. Monocular 3D object detection with LiDAR guided semi supervised active learning [C]//2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). Waikoloa, HI, USA. IEEE, 2024: 2335-2344.

[24] DENG J 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.

[25] 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). Seattle, WA, USA. IEEE, 2020: 10529-10538.

[26] REN S Q, HE K M, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[27] WOO S, PARK J, LEE J Y, et al. CBAM: convolutional block attention module [C]// Computer Vision-ECCV 2018. Cham: Springer International Publishing, 2018: 3-19.

[28] REZATOFIGHI H, TSOI N, GWAK J, et al. Generalized intersection over union: a metric and a loss for bounding box regression[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA , USA.IEEE, 2019: 658-666.

[29] 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. Providence, RI, USA. IEEE, 2012: 3354-3361.