DOI:10.19651/j.cnki.emt.2518100

基于注意力残差网络和混合池化的 3D 目标检测*

王 涛 薛庆水 王 栋 张 旭
(上海应用技术大学计算机科学与信息工程学院 上海 201418)

摘 要:针对 3D 目标检测任务中行人和骑行者的检测精度较低问题,以 Voxel-RCNN 为基准算法进行改进,提出了一种基于注意力残差网络和混合池化的 3D 目标检测算法来提升检测精度。首先,设计了一种融合残差网络和注意力机制的新型 2D 骨干网络,通过残差网络结构来增强模型对不同目标尺寸的适应性,同时引入注意力机制以聚焦于关键区域,提高特征表示能力;其次,提出了一种新型的 MLP 池化方法,同时设计了一种结合卷积的注意力池化方式,两种池化方法不仅能够有效保留小目标的局部几何细节信息,还能增强全局语义特征表达能力,从而进一步提升对复杂场景中多样性目标的捕捉能力。在公开数据集 KITTI 上的实验结果表明,Pedestrian 和 Cyclist 类别的平均精度(mAP_{3D})分别达到了 54.06%、76.85%,相比较于基准算法提升了 3.43%、3.03%。该实验结果证明了所提方法的有效性。

关键词:3D 目标检测;注意力残差网络;注意力机制;混合池化;小目标检测
中图分类号:TN958.98;TP391.4 **文献标识码**:A **国家标准学科分类代码**:510.4

3D object detection based on attention residual network and mixed pooling

Wang Tao Xue Qingshui Wang Dong Zhang Xu
(School of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai 201418, China)

Abstract: Aiming at the problem of low detection accuracy of pedestrians and cyclists in 3D object detection tasks, Voxel-RCNN is used as the baseline algorithm for improvement. A 3D object detection algorithm based on residual attention network and hybrid pooling is proposed to improve the detection accuracy. Firstly, a new 2D backbone network integrating residual network and attention mechanism is designed. The residual network structure is used to enhance the adaptability of the model to different object sizes. At the same time, the attention mechanism is introduced to focus on the key area and improve the feature representation ability. Secondly, a new MLP pooling method is proposed, and an attention pooling method combined with convolution is designed. The two pooling methods can not only effectively retain the local geometric details of small objects, but also enhance the expression ability of global semantic features, thereby further improving the ability to capture diverse objects in complex scenes. Experimental results on the public dataset KITTI show that the mean average precision (mAP_{3D}) of the Pedestrian and Cyclist categories reached 54.06% and 76.85%, respectively, which is 3.43% and 3.03% higher than the baseline algorithm. The experimental results demonstrate the effectiveness of the proposed method.

Keywords: 3D object detection; attention residual network; attention mechanism; mixed pooling; small object detection

0 引 言

随着无人驾驶技术的快速发展,研究者们对其相关应用技术的研究投入了越来越多的关注。要实现无人智能驾驶,车载环境感知是不可或缺的关键环节[1]。环境感知是自动驾驶技术的核心部分,通过传感器收集周围环境和自身状态信息,智能系统将其整合分析,为决策和控制提供依据,确保行车安全与智能性[2]。激光雷达技术是一种关键的环境感知工具,大多数 3D 目标检测依赖激光雷达提供深度信息[3],它能够直接获取深度信息,并精确捕捉目标的三维空间数据。因此,激光雷达在自动驾驶、无人机等领域应用非常广泛,成为目标检测和定位不可或缺的重要传

收稿日期:2025-02-18
* 基金项目:国家自然科学基金(61672350,61170227)、教育部基金(39120K178038,14YJA880033)、国家社会科学基金(16BGL003)项目资助

感器[4]。

三维目标检测包括基于体素、点和体素点融合的处理方法。Qi 等[5]提出了一种空间变换网络 PointNet,该网络自动学习点云的空间变换,为了获取全局特征直接使用最大池化处理,因此,PointNet 实质上没有局部的概念,这就会造成一些精细的特征缺失。Qi 等[6]再一次提出的 PointNet++在 PointNet 的基础上引入了局部特征提取机制和层次化特征学习,有效弥补了原模型在局部特征提取方面的缺陷。Shi 等[7]提出的 PointRCNN 网络在第一阶段生成初始边界框并筛选掉一些边界框数量,第二阶段对保留下来的边界框进行精细化调整。但 PointRCNN 对于点云数据的密度变化较为敏感,需要额外对数据进行归一化处理。Zhou 等[8]提出了 VoxelNet 将点云数据划分为一系列体素(Voxel),然后对每个非空体素进行局部特征编码,最后经过区域提议网络(region proposal network, RPN)模块对物体进行分类和位置回归。尽管体素化可以有效地处理点云数据,但当体素的大小较大时会丢失部分精细的几何信息。He 等[9]提出通过引入两个辅助任务来增强骨干网络的特征,从而实现更优的定位效果,并且通过能量模型进一步提高了性能。但在处理含有噪声的数据时可能会难以有效区分有用数据和噪声。Yang 等[10]提出的 3DSSD 先通过集合抽象(set abstraction, SA)进行下采样并提取关键特征,然后使用 特征传播(feature propagation, FP)机制对数据进行上采样来生成初步的边界候选框,最后对检测框进行精细调整。但该模型在目标密集的场景中可能会受到目标重叠的影响,从而导致漏检问题。Zhang 等[11]提出的网络采用了类别感知和质心感知采样策略,能够高效选择感兴趣点,适合大规模点云数据处理,这直接导致对小物体的检测效果较差,需要精细设计采样策略以避免信息损失。Yan 等[12]提出的 SECOND 采用稀疏 3D 卷积代替了传统的 3D 卷积,大幅度降低了计算复杂度,并且在保持较高精度的情况下提升了实时性检测,但对于密集区域的点云会造成精度损失。Lang 等[13]提出了一种新颖的点云编码策略,将处理后的点云送入 PointNet 网络提取特征,随后将这些特征转化为二维伪图像,借助 2D 卷积神经网络(CNN)对输入数据进行特征提取,最后由检测头输出物体回归框。由于小目标的特征较少,模型在生成二维伪图像过程中会无法有效地提取出充足的特征,这会导致小目标的检测精度下降。Zheng 等[14]通过结合软目标、硬目标以及特定约束来共同优化模型的方法,提升了车辆检测的性能。Shi 等[15]提出 PV-RCNN 采用 Voxel Set Abstraction 操作来聚合多尺度信息,并引入 Predicted Keypoint Weighting 模块动态调整关键点的权重,增强前景点的特征。此外,PV-RCNN 还设计了增强的 3D 兴趣区域池化(3D region of interest pooling, 3D ROI Pooling)操作,精确捕捉局部特征。Shi 等[16]进一步提出了 PV-RCNN++网络,它采取了体素到关键点场景编码和关键点到网格兴趣区域(region of interest, RoI)特征提取的方案,从而使检测速度比 PV-RCNN 提升两倍以上。当小目标的点云稀疏或者场景复杂时,模型依然可能在极端小物体的检测上存在较大挑战。

针对远距离道路上行人和骑行者等小目标的检测精度较低问题,在综合考虑现有方法的基础上,本文提出了一种基于 Voxel-RCNN 的改进算法。在保留原有 Voxel-RCNN 框架的同时,本文重点优化了算法的精度,尤其是小目标的识别能力。具体的改进措施包括以下 3 个方面:首先,本文重新设计了一种融合注意力机制的残差网络。与传统的 2D 骨干网络不同,新设计的 2D 网络结构能够在特征提取过程中更好地进行信息融合与特征增强。通过引入注意力机制,网络能够自动聚焦于重要区域,从而提升对关键信息的敏感度和捕捉能力,尤其是对小目标的识别。其次,为了直接聚合来自 3D 体素特征量的空间上下文,Voxel-RCNN 提出 Voxel ROI Pooling 模块,该模块可以划分为 Voxel Query 和 Voxel RoI Pooling Layer 两个子模块,针对第二步的 Voxel RoI Pooling Layer,本文设计了两种新型的池化方法:多层感知器(multilayer perceptron, MLP)池化和卷积注意力池化 (convolutional attention pooling,CAP)。结合了这两种池化方法的网络,能够提取出更加丰富、细致的三维 RoI 特征,从而增强了目标在空间位置和结构上的表达能力。

1　本文算法

Voxel-RCNN 是基于体素的两阶段算法,如图 1 所示为 Voxel-RCNN 的网络结构。

图中可以看出该算法可以分为 4 个部分组成,分别是 3D Backbone Network、2D Backbone Network、Voxel RoI pooling 和 Detect Head。在 Voxel-RCNN 算法中,点云数据经过体素化操作后被划分为规则的三维网格,然后经过 3D 骨干网络(3D backbone network)提取局部和全局的几何特征,生成稀疏的三维特征图,然后三维特征图通过沿高度方向(Z 轴)压缩转换为二维鸟瞰图(bird′s-eye-view, BEV)表示形式[17],二维骨干网络负责对生成的 BEV 特征做进一步的特征提取来增强目标的语义信息和空间分布信息。二维骨干网络的输出输入到 RPN 生成初步的三维候选框。RPN 网络通过预定义的锚框机制,结合点云特征,筛选出潜在的目标区域,这些目标区域输入到 Voxel RoI Pooling 中生成 3D RoI 特征,最后,生成的 3D RoI 特征被送入检测头,检测头由多层全连接网络组成,分别用于候选框的分类和边界框的回归优化。然而,该算法使用的是传统 2D 骨干网络,网络中通过多层卷积扩展了感受野,但对全局上下文信息的捕获仍然有限,不同尺度的特征分辨率和抽象程度差异较大,小目标可能仍会被忽略。同时 RoI aware Pooling 模块中的加速 PointNet 模块仅使用了单一的最大池化,最大池化操作通过选取每个池化窗口中的最

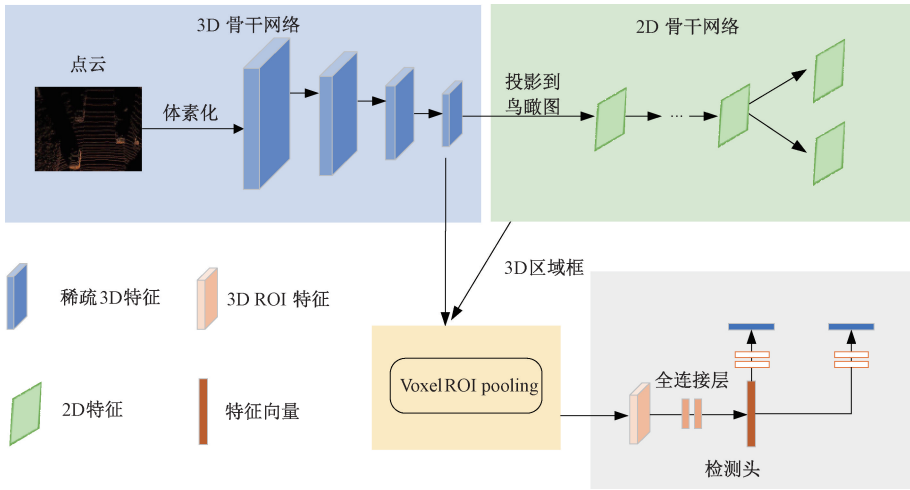

图 1　Voxel-RCNN 网络结构

Fig. 1　Network architecture of Voxel-RCNN

大值来代表该区域的特征，在处理小物体或细节时，最大池化可能无法有效保留小目标的特征，因为池化窗口可能覆盖到多个目标或背景区域，从而导致小目标的特征被弱化或丢失，从而导致小目标检测精度下降。

针对上述存在问题，从而提出了本文算法，主要涉及 3 个方面的改进。首先，原始点云数据经过体素化、3D 特征提取和高度压缩后，送入融合了注意力机制的残差网络中。在这一阶段，生成的特征继续通过 RPN 得到初步的区域候选框，随后输入到 Voxel ROI Pooling 模块中。接下来，Voxel ROI Pooling Layer 输出的特征将分别经过最大池化、MLP 池化和卷积注意力池化多个步骤分别进行处理，以提取更加丰富和多样化的 3D RoI 特征。最后，这些经过优化的特征将送入 Detect Head 进行进一步处理，从而得到最终的目标检测结果。网络结构如图 2 所示。

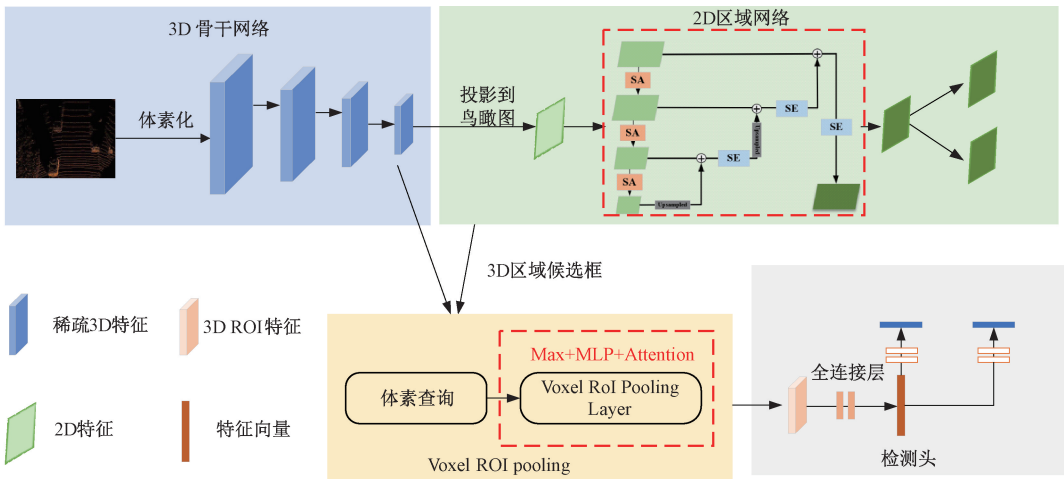

图 2　本文算法总体结构

Fig. 2　Overall structure of the algorithm in this paper

2　改进 2D Backbone Network

2.1　注意力残差网络

本文设计了一种改进的 ResNet-50 网络[18]，结合了 SA[19]模块和 SE[20]模块，显著提升了 2D 骨干网络的特征提取能力。本文将该注意力残差网络（attention reset network，ARN）替换 Voxel-RCNN 的传统 2D 骨干网络，来验证其在 3D 目标检测中的优势。ARN 网络如图 3 所示。

该网络由两部分组成，第 1 部分是基于 ResNet-50 的特征提取模块，下采样的每层添加一个 SA 模块以优化通道和空间特征交互；第 2 部分是多尺度融合模块，通过子像素卷积（SubPixelConv）[21]上采样恢复分辨率，并结合 SE 模块动态调整通道权重，最终通过 SE 模块对融合特征进行全局权重调整，并生成最终特征图。此外，本文在经典 ResNet-50 的基础上进行了以下改进：在每个残差层后引入了 SA 模块；在多尺度特征融合阶段，引入了 SE 模块；使用 SubPixelConv 模块替代传统的上采样方法，该模块通过重排

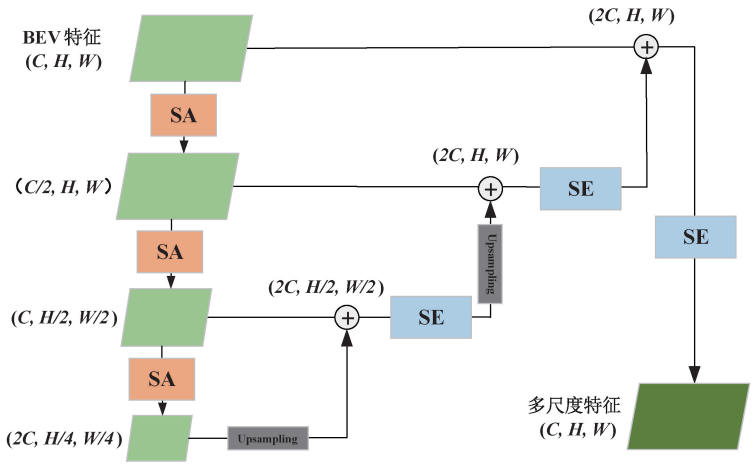

图 3　注意力残差网络

Fig. 3　Attention residual network

列实现上采样,避免了反卷积操作导致的棋盘效应。

原始 BEV 特征图大小为(C,H,W)。首先,第 1 层(Layer1)接收到的初始特征图大小为(C,H,W),通过步长为 2 的卷积将通道数调整为 $C/2$,使用 3 个 Bottleneck 模块提取初级特征,然后应用 Shuffle Attention 优化通道与空间维度特征,得到特征图大小为 ($C/2$,H,W);第 2 层(Layer2)输入来自 Layer1 输出的特征图 ($C/2$,H,W),通过步长为 2 的卷积将分辨率减半,通道数调整为 C,使用 4 个 Bottleneck 模块提取中级特征,继续应用 Shuffle Attention 进一步优化特征表达;第 3 层(Layer3)输入来自 Layer2 的输出特征图 (C,$H/2$,$W/2$),通过步长为 2 的卷积进一步减半分辨率,通道数提升为 $2C$,使用 6 个 Bottleneck 模块提取高级特征,最后应用 Shuffle Attention 对高层次特征进行增强,最终输出的特征图大小为($2C$,$H/4$,$W/4$)。接下来进行各层之间模块的融合,Layer3 特征图($2C$,$H/4$,$W/4$)通过子像素卷积(SubPixelConv)上采样至(C,$H/2$,$W/2$),与 Layer2 特征图 (C,$H/2$,$W/2$) 进行拼接,形成融合特征 Fusion1,Fusion1 通过卷积与 SE 模块校准,输出大小为(C,$H/2$,$W/2$);然后,Fusion1 特征图通过 SubPixelConv 上采样至($C/2$,H,W),与 Layer1 特征图($C/2$,H,W)拼接,形成融合特征 Fusion2,继续通过卷积与 SE 模块校准,输出大小为(C,H,W);最后,Fusion2 与初始输入特征图(C,H,W)拼接,然后通过 SE 模块优化通道权重,再经过卷积层调整为最终输出特征图(C,H,W)。该残差网络通过结合 Shuffle Attention、Squeeze-and-Excitation Networks 以及 SubPixelConv,显著增强了特征的提取与融合能力,尤其是在小目标物体的检测精度大幅提升。

2.2　SE 注意力机制

SE 注意力机制是一种轻量且高效的模块,用于增强卷积神经网络(CNN)的特征表示能力,SE 结构如图 4 所示。

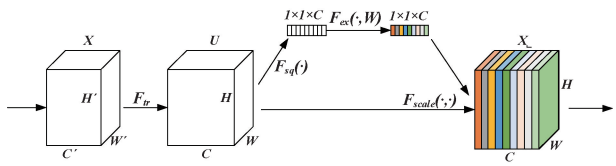

图 4　SE 注意力模块

Fig. 4　SE attention module

SE 模块通过显式建模特征通道之间的依赖关系,重新校准每个通道的响应,使网络能够更专注于关键的特征通道,并抑制冗余或不重要的通道。主要有以下 4 个步骤:

第 1 步转型,输入 $X \in C' \times H' \times W'$,经过 F_{tr} 输出 $U \in C \times H \times W$。$F_{tr}$ 表示可学习的卷积核集合。该过程公式如下:

$$U_c = V_c * X = \sum_{s=1}^{c'} V_c^s * X^s \tag{1}$$

其中,$*$ 表示卷积,V_c 表示第 c 个通道的卷积核,X 表示输入,V_c^s 表示一个 2D 的空间核。

该卷积算子公式表示,输入的特征图每一层都经过一个 2D 空间核的卷积最终得到 C 个输出的特征图,组成最终的特征图。

第 2 步全局信息嵌入,考虑到每一个卷积操作因为其感受野只能获取局部的上下文信息,为了进一步获取全局信息,SE 注意力将全局空间信息压缩到一个通道描述符中。这一操作通过全局平均池化来实现,将包含全局信息的特征图直接压缩成一个 $1\times1\times C$ 的特征向量 $\boldsymbol{Z}$,C 个特征图的通道特征都被压缩成了一个数值,这样使得生成的通道级统计数据 $\boldsymbol{Z}$ 就包含了上下文信息,缓解了通道依赖性的问题。公式如下:

$$z_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{W}^{H} \sum_{H}^{W} X_c(i,j), \quad c \in \{1,2,\cdots,C\} \tag{2}$$

其中，z_c 为 Z 的第 c 个元素，u_c 表示第 C 通道的特征图，大小为 $H\times W$，(i,j) 表示特征图上的坐标位置值。

第 3 步激励操作，为了利用压缩操作中汇聚的信息，继续执行 F_{ex} 操作，通过该操作来全面捕获通道依赖性。在该阶段，SE 模块通过一个轻量级的两层全连接网络对通道描述向量 $\boldsymbol{z}$ 进行建模，从而学习不同通道间的依赖关系。门控单元上 $\boldsymbol{s}$（即图 4 中 $1\times1\times C$ 的特征向量）的计算方式如下：

$$\boldsymbol{s}=F_{ex}(\boldsymbol{z},W)=\sigma(g(\boldsymbol{z},W))=\sigma(W_2\delta(W_1\boldsymbol{z})) \quad (3)$$

其中，δ 表示 ReLU 激活函数，σ 表示 Sigmoid 激活函数，$W_1\in R^{\frac{C}{r}\times C}$ 和 $W_2\in R^{C\times\frac{C}{r}}$ 两个全连接层的权值矩阵，r 是维度衰减因子。

最后是重新加权（Scale）操作，将前面得到的注意力权重加权到每个通道的特征上。在该阶段，SE 模块将学习到的通道权重重新作用于特征图 U，完成通道特征的增强，得到 SE 模块的最终输出，即图 4 中的 $F_{scale}(\cdot,\cdot)$ 操作：

$$\widetilde{X}_c=F_{scale}(u_c,s_c)=s_cu_c \quad (4)$$

其中，s_c 是门控单元 $\boldsymbol{s}$（向量）中的一个标量值。

本文将 SE 注意力模块嵌入到 Resnet-50 网络中，通过动态建模通道间的关系，SE 模块能够帮助网络更加聚焦于任务相关的关键通道。SE 模块通过结合压缩和激励操作，巧妙地在特征通道间引入了动态权重调整机制，从而提升了网络的表达能力。

2.3　SA 注意力模块

空间注意力机制和通道注意力机制分别用于捕获成对的像素级关系和通道间依赖关系的。同时使用两种注意力机制可以达到更好的效果，但是不可避免地增加了模型的计算量。SA 模块可以来解决这个问题，并且能高效地结合两种注意力机制，整个 SA 模块分为 3 个步骤，分别为特征分组、混合注意力和特征聚合，如图 5 所示。

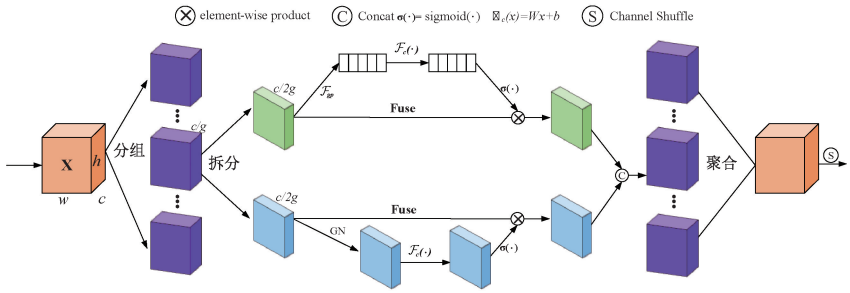

图 5　SA 注意力模块

Fig. 5　SA attention module

最后，所有的子特征会被汇集起来，为了达到特征通信目的，SA 使用通道混洗让不同组的特征进行融合，通道混洗通过重新排列通道来实现不同组之间的特征通信，从而提高网络的特征提取能力。

SA 首先对通道特征进行分组，它将输入特征图分为多组，每组为一个子特征。具体来看，输入特征图 $X\in\mathbb{R}^{C\times H\times W}$ 被沿着通道维度分为 G 组，表示为 $X'=[X_1,\cdots,X_G]$，每一个组的特征大小是 $\mathbb{R}^{\frac{C}{G}\times H\times W}$，这些子特征随着训练会逐渐捕获一种特定的语义信息，这部分对应图 5 中左侧的 Group 标注的部分。接下来通过混合注意力处理，X' 被分为两个分支，依然是沿着通道维度划分，两个子特征表示为 $X_{k1},X_{k2}\in\mathbb{R}^{\frac{C}{2G}\times H\times W}$，即图 5 中间拆分（Split）标注后的部分，该部分可分为上下两个分支，上分支实现通道注意力开采通道间的依赖，下分支则捕获特征之间的空间依赖生成空间注意力图。通道注意力分支过程变换公式如下：

$$s=\mathcal{F}_{gp}(X_{k1})=\frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W}X_{k1}(i,j) \quad (5)$$

$$X'_{k1}=\sigma(\mathcal{F}_c(s))\cdot X_{k1}=\sigma(W_1s+b_1)\cdot X_{k1} \quad (6)$$

上述公式只有两个变换参数，即 $W_1\in\mathbb{R}^{C/2G\times1\times1}$ 和 $b_1\in\mathbb{R}^{C/2G\times1\times1}$。空间注意力分支对输入特征图进行 Group Norm（GN）操作，通过一个变换 $\mathcal{F}_c(\cdot)$ 来增强输入的表示，具体公式如下：

$$X'_{k2}=\sigma(W_2\cdot GN(X_{k2})+b_2)\cdot X_{k2} \quad (7)$$

其中，$W_2,b_2\in\mathbb{R}^{\frac{C}{2G}\times1\times1}$。

然后，对结果进行拼接，最后的聚合操作使用一种轻量化网络结构 ShuffleNetV2[22] 中的通道混洗来保证各组子特征之间的交互，最后得到和输入 X 相同维度的注意力图。

3　改进 Accelerate PointNet Module

基准模型 Voxel-RCNN 在 Voxel ROI Pooling 中使用了加速 PointNet 模块（accelerated PointNet module），以进一步降低体素查询的计算复杂度。该结构如图 6 所示，总共有 M 个网格点，每个网格点需要查找 K 个体素，每个 voxel 特征向量为 $\boldsymbol{C}+3$，将 voxel 特征和相对坐标进行拆分，包括 $\boldsymbol{C}$ 维体素特征和 3 维相对坐标，由于特征向量和网格点是相互独立的，因此对每个 voxel 进行特征变换，融合后的特征向量为 $\boldsymbol{C}'$，最后对融合的特征向量做最大池化处理。最大池化作为一种常用的下采样操作，虽然能够有效提取特征并降低计算复杂度，但可能会导致重要的细节特征被丢弃，尤其是在小目标或细粒度特征的场景中。

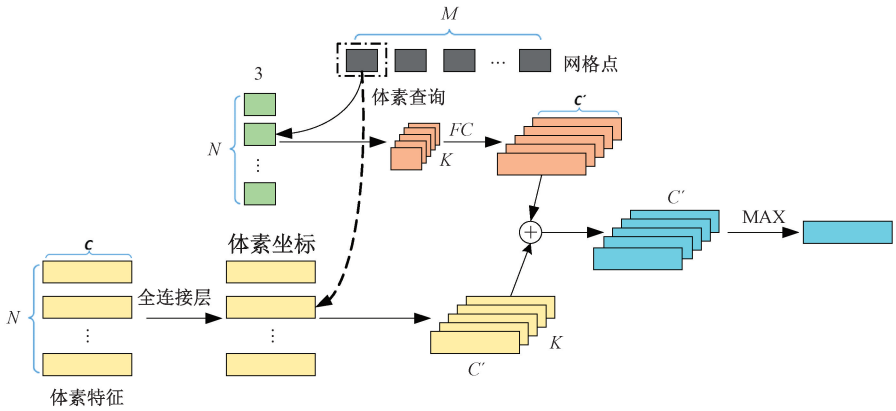

图 6　加速 PointNet 模块

Fig. 6　Accelerated PointNet module

本文算法以 Voxel-RCNN 为基础网络,进一步改进了模型中的加速 PointNet 模块,在该模块中引入了 MLP 池化和卷积注意力池化,将最大池化、MLP 池化和卷积注意力池化相结合,网络能够学习到更加细粒度的特征。最大池化确保保留最突出的特征,MLP 池化有助于学习复杂的特征关系,卷积注意力池化则通过关注最重要的特征进一步提高模型的判别能力。通常小物体在传统的池化操作中会因为信息丢失而变得难以识别,通过结合多种池化策略,提取的特征不仅能够保留小物体的细节,还能够突出重要的局部特征。如图 7 所示,改进的加速 PointNet 模块由 3 个分支组成:MLP 池化模块、卷积注意力池化模块 CAP 和最大池化模块(MAX Pooling)。

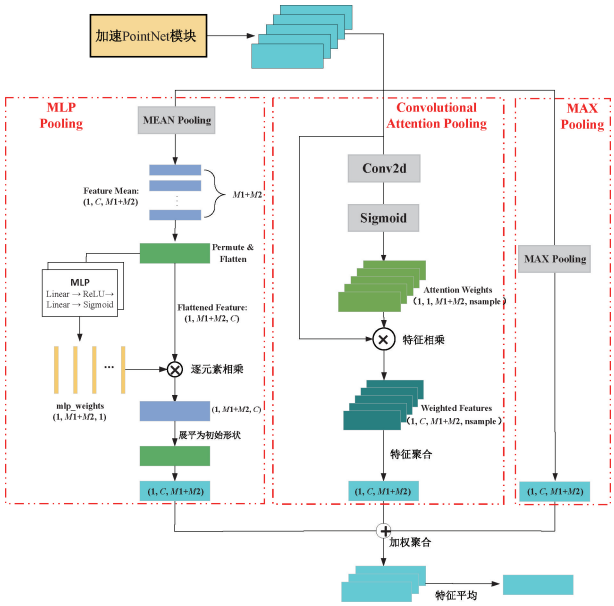

图 7　改进的加速 PointNet 模块

Fig. 7　Improved accelerated PointNet module

3.1　最大池化编码模块

在最大池化操作中,只有每个矩形区域内的最大值被选取并传递到下一层,其余元素则被忽略。通过这种方式,最大池化能够提取出特征图中响应最强的部分,从而将冗余信息剔除,帮助网络更高效地优化。然而,这种操作也有其局限性,它往往会丢失一些细节信息。基准模型中的 Voxel RoI Pooling 在加速 PointNet 模块的最后阶段,利用最大池化编码来提取具有较高分类辨识度的特征。该过程可以表示为:

$$f_m = \max(C'_i), i \in \{1,2,\cdots,n\} \tag{8}$$

3.2　MLP 池化模块

本文设计了一种 MLP 池化方法,旨在通过动态分配权重来增强特征聚合的灵活性和表达能力,如图 8 所示。

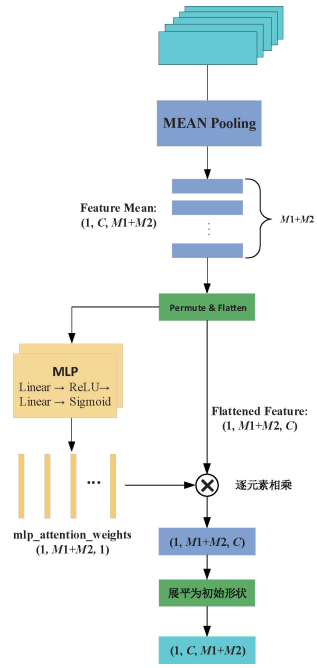

图 8　MLP 池化模块

Fig. 8　MLP pooling module

该方法在传统池化方式的基础上引入了多层感知(MLP),用于生成注意力权重,并通过自适应地调整各特

征对全局表示的贡献，突出关键信息。具体而言，输入特征首先沿邻域维度进行均值池化，提取全局描述后，通过 MLP 生成注意力权重，并与原始特征进行加权求和。

在模型中，池化前的特征张量形状为(1,C,$M1+M2$,$nsample$)，C 表示特征通道数；$M1+M2$ 是采样点的总数，$nsample$ 是每个采样点的邻域点数。该张量包含了每个采样点的局部邻域特征，目的是聚合这些特征，从而生成适用于后续任务的全局特征表示。在原始特征中，每个采样点的邻域特征是一个含 $nsample$ 点的子集。为了生成注意力权重，首先需要从邻域中提取全局简化表示。具体步骤如下：沿着邻域维度($nsample$)对特征进行均值池化，得到每个采样点的特征表示。该过程表示为：

$$f_{mean}(i)=\frac{1}{nsample}\sum_{j=1}^{nsample}X[:,:,i,j] \tag{9}$$

$$X_{flat}=permute(f_{mean}(i),(0,2,1)) \tag{10}$$

其中，$f_{mean}(i)$ 表示采样点 i 的均值特征，X_{flat} 表示将 $f_{mean}(i)$ 进行维度变换，$permute$ 是用于重新排列张量维度的操作。

为了适配 MLP 的输入，特征张量的形状由 ($1,C$,$M1+M2$) 变为($1,M1+M2,C$)，通过该操作，每个采样点的邻域特征被压缩为一个全局特征表示，为后续的注意力计算提供了基础。然后，多层感知机(MLP)用于生成注意力权重。每个采样点的全局特征 X_{flat} 输入到一个两层的 MLP 中，经过非线性激活函数($ReLU$)处理后，输出一个标量权重 $\omega(i)$，表示该特征的重要性，公式为：

$$\omega(i)=\sigma(\omega_2\cdot ReLU(\boldsymbol{W}_1\cdot X_{flat}+b_1)+b_2) \tag{11}$$

其中，$\boldsymbol{W}_1$ 和 $\boldsymbol{W}_2$ 是 MLP 的权重矩阵，b_1 和 b_2 是偏置项，$ReLU$ 是激活函数，用于引入非线性，σ 是 Sigmoid 激活函数，将权重限制在[0,1]范围内。

最后，生成的注意力权重被用来加权输入的特征，动态调整每个采样点邻域特征对全局特征表达的贡献，再将加权后的特征恢复到三维张量的形式。具体公式为：

$$F_{pool}(i)=\omega(i)\cdot X_{flat}(i) \tag{12}$$

$$F_{out}=permute(F_{pool},(0,2,1)) \tag{13}$$

式中：$\omega(i)$ 是采样点 i 的注意力权重；X_{flat} 是变换的特征。通过上述步骤，MLP 池化不仅动态调整了特征的重要性，还通过自适应的权重策略突出关键信息，同时抑制噪声或冗余特征。

3.3　卷积注意力池化模块

CAP 是一种通过引入空间和特征域注意力机制来增强点云特征表达能力的池化方法，旨在更有效地捕获局部上下文信息并提升关键点特征的权重分配，如图 9 所示。

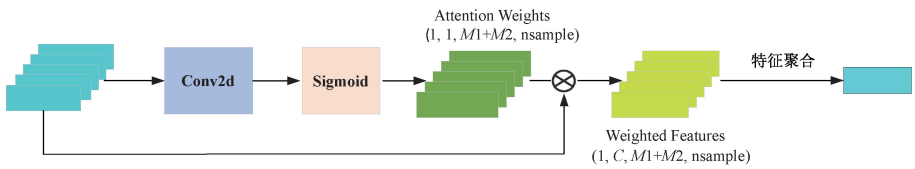

图 9　卷积注意力池化模块

Fig. 9　Convolutional attention pooling module

该方法的核心思想是结合卷积操作和注意力机制，通过动态调整特征的权重，以增强重要特征的表达能力。具体处理过程包括以下步骤：

1)权重生成：卷积注意力池化的核心在于通过一个 1×1 卷积操作生成注意力权重，用以衡量每个点的重要性，将输入特征 F_{input} 的每个通道投影到单通道注意力权重，生成的权重张量 $W_{attention}$ 形状为(1, 1, $M1+M2$, $nsample$)。该操作用于学习每个点在其局部邻域中的重要性。在卷积输出后，应用 Sigmoid 激活函数，将注意力权重值限制在[0,1] 范围，以确保数值稳定性并直观地表示每个点的重要性。具体公式如下：

$$W_{attention}=\sigma(Conv_{1\times1}(F_{input})) \tag{14}$$

其中，σ 是 Sigmoid 激活函数，F_{input} 是输入特征。

2)加权特征计算：得到注意力权重后，通过对输入特征进行逐元素相乘来计算加权特征。该操作通过抑制不重要的点并强化关键点特征的表达，使网络能够在局部范围内更加有效 地关注有意义的特征。加权特征计算公式为：

$$F_{attention}=W_{attention}\cdot F_{input} \tag{15}$$

式中：$W_{attention}$ 是生成的注意力权重；F_{input} 是初始的输入特征。

3)特征聚合：经过加权后的特征 $F_{attention}$ 仍然保留了分组维度 $nsample$。为进一步提取特征的全局表达需要沿着最后一维 $nsample$ 对加权特征进行求和聚合，聚合后的输出张量形状为 (1, C, $M1+M2$)，表示每个分组的全局特征：

$$F_{output}=\sum_{j=1}^{nsample}F_{attention}(i) \tag{16}$$

3.4　特征融合模块

为了充分发挥最大池化、MLP 注意力池化和卷积注意力池化的优势，可以将这 3 种池化方法的结果进行加权平均，从而获得更丰富的特征表示：

$$F_{final}=\frac{F_{\max}+F_{MLP}+F_{Conv}}{3} \tag{17}$$

通过这种融合策略，可以平衡不同池化方法的优势：最大池化提供了强大的局部特征提取能力，能够在降维时保留显著的特征；MLP 注意力池化能够根据特征的重要性动态地加权，提高对关键特征的聚焦能力；卷积注意力池化则在捕捉局部空间模式的同时，保持较高的计算效率。

4　损失函数

本文所使用的损失函数包括 RPN 损失和 Detect Head 损失。算法中的 RPN 损失包含分类损失(classification loss)和回归损失(box regression loss):

$$\mathcal{L}_{RPN}=\frac{1}{N_{fg}}[\sum_i \mathcal{L}_{RPN}(p_i^a,c_i^*)+1(c_i^*\geqslant 1)\sum_i \mathcal{L}_{reg}(\delta_i^a,t_i^*)] \tag{18}$$

其中,N_{fg} 为前景 anchors 数量,p_i^a 和 δ_i^a 分别是分类和回归输出,c_i^* 和 t_i^* 分别是分类标签和回归目标,$1(c_i^*\geqslant 1)$ 表示只计算前景 anchors 的回归损失。分类损失函数使用 Focal Loss,回归损失函数为 Huber Loss。

Detect Head 损失使用置信度函数如下:

$$\ell_i^*(IoU_i)=\begin{cases}0, IoU_i<\theta_L\\ \dfrac{IoU_i-\theta_L}{\theta_H-\theta_L}, \theta_L\leqslant IoU_i\leqslant\theta_H\\ 1, IoU_i>\theta_H\end{cases} \tag{19}$$

其中,IoU_i 是第 i 个方案和对应真值框的 IoU,θ_H 和 θ_L 分别是前景 IoU 和背景 IoU 的阈值。这里的置信度预测为二分类交叉熵函数,回归损失也使用 RPN 中的 Huber Loss,于是最终的检测头的损失函数为:

$$\mathcal{L}_{head}=\frac{1}{N_s}[\sum_i \mathcal{L}_{cls}(p_i,\ell_i^*(IoU_i))+1(IoU_i\geqslant\theta_{reg})\sum_i \mathcal{L}_{reg}(\delta_i,t_i^*)] \tag{20}$$

其中,N_s 是训练阶段的采样区域方案数量,$1(IoU_i\geqslant\theta_{reg})$ 表示只有 IoU 大于 θ_{reg} 得到区域方案。

5　实验与结果分析

5.1　数据集

在数据集的选择上,本文特别考虑了数据的多样性、标注的准确性以及该领域的代表性。本文选用了 KITTI 数据集作为实验的主要数据源。首先,KITTI 数据集涵盖了丰富多样的驾驶场景,包括不同类型的道路、行人、车辆等目标,这些多样化的场景为深度学习模型提供了很好的泛化能力,有助于模型在复杂环境下的应用。其次,KITTI 数据集的标注由人工精确完成,并经过严格的校验,确保了数据的准确性与可靠性。此外,KITTI 数据集在自动驾驶和 3D 目标检测领域具有广泛的应用和影响力,因此,作为一个标准数据集,它能有效验证本文所提模型的性能和实际效果。

KITTI 数据集包含 7 481 个训练样本和 7 518 个测试样本。在实验过程中,本文按照 Voxel-RCNN 算法的标准,从训练点云数据中随机选取 80%的样本用于模型训练,并将剩余的 20%样本用于验证。为了全面评估模型在不同情况下的表现,数据集分为简单、中等和困难 3 个难度等级。通过这种分级方式,可以更好地衡量模型在不同挑战下的检测能力与鲁棒性。

5.2　实验设置

本文实验基于 OpenPCDet 框架,实现对 KITTI 数据集和改进后的 Voxel-RCNN 算法的研究与实验。硬件环境包括 CPU Intel(R) Xeon(R) Platinum 8375C CPU @ 2.90 GHz、GeForce RTX 3090(24 GB)、72 GB 内存,搭载 Ubuntu 20.04 操作系统,软件环境为 Python 3.8、Cuda 11.8 和 PyTorch 2.0.0。实验将点云的范围裁剪为 X 轴的[0,70.4] m,Y 轴的[−40,40] m 和 Z 轴的[−3,1]m。输入体素大小设置为(0.05 m,0.05 m,0.1 m),使用 Adam_onecycle 优化器进行端到端训练,设置批次大小为 4,权重衰减值为 0.01,动量为 0.9,学习率衰减值为 0.1,最大迭代次数为 80。

5.3　实验结果及分析

本文算法在 KITTI 数据集上遵循评价指标对汽车、行人、骑行者 3 个类别分别进行简单、中等、困难 3 个难度级别下的性能评估,所有实验结果通过平均精度(AP)进行评估。汽车的交并比(IoU)阈值设为 0.7,行人和骑行者的 IoU 阈值设为 0.5。为了评估改进 Voxel-RCNN 算法的性能,在 KITTI 数据集上对所提算法与基于 LiDAR 的 3D 目标检测相关算法进行测试对比,如表 1 所示。

表 1　不同算法在 KITTI 验证集上的对比结果

Table 1　Comparison results of different algorithms on the KITTI validation set

算法	汽车(IoU=0.7)			行人(IoU=0.5)			骑行者(IoU=0.5)			3DmAP
	简单	中等	困难	简单	中等	困难	简单	中等	困难	
PointPillars	87.25	75.72	72.75	50.67	44.33	39.80	79.45	62.44	58.25	63.40
Second	88.76	79.10	75.96	50.76	45.86	41.07	81.06	66.17	62.23	65.66
PointRCNN	89.34	80.17	77.69	60.04	53.19	48.28	**90.08**	71.36	66.75	70.77
Part-A^2	91.94	83.32	79.94	60.13	52.51	46.87	85.73	70.63	66.12	70.80
PV-RCNN	92.09	82.64	80.07	56.67	48.99	44.40	83.59	68.92	64.64	69.11
PV-RCNN++	91.06	82.37	80.71	59.23	51.37	47.62	85.21	70.90	66.08	70.51
IA-SSD	88.45	80.09	77.89	56.39	52.78	47.99	87.73	71.98	67.33	70.07
Voxel-RCNN	92.15	82.56	79.64	58.09	49.03	44.76	87.26	69.45	65.07	69.78
Ours	**92.84**	**82.73**	**80.23**	**60.22**	**53.40**	**48.57**	89.38	**72.82**	**68.35**	**72.06**

5.4　消融实验

为了验证所提算法各模块的有效性，在 KITTI 数据集的 3 种类型目标检测方面进行了消融实验，以对提出的注意力残差网络、MLP 池化和卷积注意力池化进行消融研究，分析它们对模型检测性能的影响。消融实验共有 4 组对比实验，表 2 详细展示了每个所提模块对算法检测性能的影响。

表 2　消融实验结果

Table 2　Ablation experiment results

方法	ARN	MLP	CAP	mAP_{3D}/%		
				汽车	行人	骑行者
Baseline				84.79	50.63	73.82
Experiment1	√			84.91	53.94	76.13
Experiment2		√		85.06	53.01	75.51
Experiment3			√	84.83	52.88	75.67
Experiment4		√	√	85.23	53.51	76.59
Ours	√	√	√	**85.27**	**54.06**	**76.85**

基线模型 Voxel-RCNN 未使用本文所提出的注意力残差网络、MLP 池化及卷积注意力池化。实验 1 在基准算法上使用提出的 ARN 替换 2D Backbone，汽车，行人和骑行者类别的 mAP_{3D} 相比较于基准算法分别提升了 0.12%、3.31%、2.21%。实验 2 在基准算法的加速 PointNet 模块添加 MLP 池化模块，汽车，行人和骑行者类别的 mAP_{3D} 相比较于基准算法分别提升了 0.27%、2.38%、1.59%。实验 3 在基准算法的加速 PointNet 模块添加 CAP，汽车，行人和骑行者类别的 mAP_{3D} 相比较于基准算法分别提升了 0.04%、2.25%、1.75%。实验 4 将基准算法中加速 PointNet 模块的最大池化、MLP 池化和卷积注意力池化相结合，汽车，行人和骑行者类别的 mAP_{3D} 相比较于基准算法分别提升了 0.44%、2.88%、2.67%。

本文算法使用提出的 ARN 替换基准算法的 2D Backbone，在基准算法的加速 PointNet 模块添加 MLP 池化模块和 CAP 模块后，汽车，行人和骑行者类别的 mAP_{3D} 相比较于基准算法分别提升了 0.48%、3.43%、3.03%。结果表明，本文所提出的方法，小目标物体的检测效果得到了显著的提升，验证了本文所提方法的有效性。

5.5　可视化结果

在 KITTI 数据集上对本文算法的效果进行可视化，如图 10 所示。图 10 分场景一和场景二上下两部分，每个场景上半部分左侧是基准算法在该场景下的效果图，右侧是本文算法在该场景下的效果图，场景图的下半部分是真实场景下对应的相机图像。从图中虚线框区域可以看出本文算法相比于基准算法在目标距离较远的情况下有效的缓解了基准算法的漏检问题，说明基准算法在远距离较为复杂的路面下检测性能的不足。此外，在图 10 场景一中，距离行人目标较近的区域，基准算法也会出现漏检问题，本文算法可以较好的降低误检率，取得了较好的可视化效果。然而，从场景二中可以看出本文算法也存在着一定的误判情况，图中上方虚线标注的两个 3D 框是将路边广告牌附近的树木枝干误识别为目标物，说明本文的算法还存在着一定的提升空间。

(a) 场景一
(a) Scenario one

(b) 场景二
(b) Scenario two

图 10　可视化结果对比图

Fig. 10　Visualization result comparison

综上所述，本文所提算法在复杂场景的小目标检测和远距离检测上有着较好的检测精度，因此整体的效果有所提升。

6　结　　论

本文提出了一种结合注意力机制的多尺度网络以及融合 MLP 池化模块和卷积注意力池化模块的 3D 目标检测算法。通过引入多尺度网络结构，有效整合了不同分辨率下的空间特征，提升了模型对多样化场景中目标的鲁棒性和感知能力。MLP 池化模块通过高效提取全局特征，增强了目标特征的语义表达能力，而卷积注意力池化模块进一步优化了模型对关键特征的关注度，提高了分类和 3D 边界框回归的精度。实验结果显示，Car，Pedestrian 和 Cyclist 类别的 mAP_{3D} 相比较于基准算法分别提高了 0.48%、3.43%、3.03%。该结果表明本文所提出的方法在小目标物体的检测效果得到了显著的提升，验证了本文所提方法的有效性。

本文算法在 KITTI 数据集上的性能在行人和骑行者这类难检测目标的场景下表现突出，证明了其在复杂驾驶场景中的有效性。然而，本文算法在远距离目标检测的精度上仍有提升空间，同时在处理稀疏点云时存在一定的局限性。未来研究将进一步探索注意力机制在点云特征提取中的潜力，优化模型对稀疏场景和远距离目标的适应能力，从而提升整体检测性能。

参考文献

[1]　王庆林，李辉，谢礼志，等. 基于激光雷达点云的车辆目标检测算法改进研究[J]. 电子测量技术，2023，46(1)：120-126.
WANG Q L，LI H，XIE L ZH，et al. Research on improving vehicle target detection algorithm based on

lidar point cloud [J]. Electronic Measurement Technology, 2023, 46(1):120-126.

[2] 吴文涛,何赟泽,杜旭,等.融合相机与激光雷达的目标检测与尺寸测量[J].电子测量与仪器学报,2023,37(6):169-177.
WU W T, HE Y Z, DU X, et al. Target detection and size measurement based on the fusion of camera and lidar[J]. Journal of Electronic Measurement and Instrumentation, 2023, 37(6):169-177.

[3] 郑自立,徐健,刘秀平,等.联合多注意力和 C-ASPP 的单目 3D 目标检测[J].电子测量与仪器学报,2023,37(8):241-248.
ZHENG Z L, XU J, LIU X P, et al. Monocular 3D object detection with joint multi-attention and C-ASPP [J]. Journal of Electronic Measurement and Instrumentation, 2023, 37(8):241-248.

[4] 汤新华,代道文,陈熙源,等.基于 PointPillars 的改进三维目标检测算法[J].仪器仪表学报, 2024,45(9): 260-269.
TANG X H, DAI D W, CHEN X Y, et al. Improved 3D target detection algorithm based on PointPillars [J]. Chinese Journal of Scientific Instrument, 2024, 45(9): 260-269.

[5] QI C R, SU H, MO K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2017: 652-660.

[6] QI C R, YI L, SU H, et al. PointNet++: Deep hierarchical feature learning on point sets in a metric space[C]. 31st International Conference on Neural Information Processing Systems, 2017:5105-5114.

[7] SHI SH SH, WANG X G, LI H SH. Pointrcnn: 3D object proposal generation and detection from point cloud[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019:770-779.

[8] ZHOU Y, TUZEL O. Voxelnet: End-to-end learning for point cloud based 3D object detection[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2018: 4490-4499.

[9] HE CH H, ZENG H, HUANG J Q, et al. Structure aware single-stage 3D object detection from point cloud[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 11873-11882.

[10] YANG Z T, SUN Y N, LIU SH, et al. 3DSSD: Point-based 3D single stage object detector[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 11040-11048.

[11] ZHANG Y F, HU Q Y, XU G Q, et al. Not all points are equal: Learning highly efficient point-based detectors for 3D lidar point clouds[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022: 18953-18962.

[12] YAN Y, MAO Y X, LI B. Second: Sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10): 3337.

[13] LANG A H, VORA S, Caesar H, et al. Pointpillars: Fast encoders for object detection from point clouds[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 12697-12705.

[14] ZHENG W, TANG W L, JIANG L, et al. SE-SSD: Self-ensembling single-stage object detector from point cloud[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021:14494-14503.

[15] SHI SH SH, GUO CH X, JIANG L, et al. PV-RCNN: Point-voxel feature set abstraction for 3D object detection [C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 10529-10538.

[16] SHI SH SH, JIANG L, DENG J J, et al. PV-RCNN++: Point-voxel feature set abstraction with local vector representation for 3d object detection[J]. International Journal of Computer Vision, 2023, 131(2): 531-551.

[17] LI H Y, SIMA CH H, DAI J F, et al. Delving into the devils of bird′s-eye-view perception: A review, evaluation and recipe [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024, 46(4): 2151-2170.

[18] HE K M, ZHANG X Y, REN SH Q, et al. Deep residual learning for image recognition[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR), 2016: 770-778.

[19] ZHANG Q L, YANG Y B. SA-Net: Shuffle attention for deep convolutional neural networks [C]. 2021 IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP), 2021: 2235-2239.

[20] HU J, SHEN L, SUN G. Squeeze-and-excitation networks [C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018: 7132-7141.

[21] SHI W ZH, CABALLERO J, HUSZÁR F, et al. Real-Time single image and video super-resolution using an efficient Sub-Pixel convolutional neural network[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 1874-1883.

[22] MA N N, ZHANG X Y, ZHENG H T, et al. Shufflenet v2: Practical guidelines for efficient CNN architecture design [C]. European Conference on Computer Vision(ECCV), 2018: 116-131.

作者简介

王涛,硕士研究生,主要研究方向为三维目标检测。
E-mail:236142142@mail.sit.edu.cn

薛庆水(通信作者),博士,教授,主要研究方向为网络空间安全、人工智能安全等。
E-mail:xue-qsh@sit.edu.cn

王栋,博士,副教授,主要研究方向为新一代人工智能与深度学习技术、大数据技术等。
E-mail:dongwang@sit.edu.cn

张旭,硕士研究生,主要研究方向为三维目标检测。
E-mail:236142158@mail.sit.edu.cn