智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

基于多通道交叉注意力融合的三维目标检测算法

鲁斌, 杨振宇, 孙洋, 刘亚伟, 王明晗

引用本文:
鲁斌, 杨振宇, 孙洋, 刘亚伟, 王明晗. 基于多通道交叉注意力融合的三维目标检测算法[J]. 智能系统学报, 2024, 19(4): 885-897.
LU Bin, YANG Zhenyu, SUN Yang, et al. 3D object detection algorithm with multi-channel cross attention fusion[J]. *CAAI Transactions on Intelligent Systems*, 2024, 19(4): 885-897.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202305029

DOI: 10.11992/tis.202305029
网络出版地址: https://link.cnki.net/urlid/23.1538.TP.20240604.0923.004

基于多通道交叉注意力融合的三维目标检测算法

鲁斌[1,2], 杨振宇[1,2], 孙洋[1,2], 刘亚伟[1,2], 王明晗[1,2]
（1. 华北电力大学 控制与计算机工程学院, 河北 保定 071000; 2. 华北电力大学 河北省能源电力知识计算重点实验室, 河北 保定 071000）

摘　要: 针对现有单阶段三维目标检测算法对点云下采样特征利用方式单一、特征对长程上下文信息的聚合程度无法满足算法性能提升需求的问题, 本文提出了基于多通道交叉注意力融合的单阶段三维目标检测算法。首先, 设计通道交叉注意力模块用于融合下采样特征, 可基于交叉注意力机制在通道层面上增强多尺度特征对不同感受野下长程空间信息的表达能力; 然后, 提出级联特征激励模块, 结合原始下采样特征对通道交叉注意力加权特征进行级联激励, 提升算法对关键空间特征的学习能力。在公共自动驾驶数据集 KITTI 上进行了大量实验并与主流算法对比, 本文算法作为单阶段目标检测算法, 在车辆类别 3 个难度级别上的检测准确率分别为 91.34%、79.85% 和 75.98%, 较基线算法分别提升了 4.83%、3.26% 和 3.32%。实验结果证明了本文算法及所提模块在三维目标检测任务上的有效性和先进性。

关键词: 三维点云; 自动驾驶; 激光雷达; 深度学习; 三维目标检测; 柱体素; 交叉注意力; 单阶段算法
中图分类号: TP391　　**文献标志码:** A　　**文章编号:** 1673−4785(2024)04−0885−13

中文引用格式: 鲁斌, 杨振宇, 孙洋, 等. 基于多通道交叉注意力融合的三维目标检测算法 [J]. 智能系统学报, 2024, 19(4): 885–897.
英文引用格式: LU Bin, YANG Zhenyu, SUN Yang, et al. 3D object detection algorithm with multi-channel cross attention fusion[J]. CAAI transactions on intelligent systems, 2024, 19(4): 885–897.

3D object detection algorithm with multi-channel cross attention fusion

LU Bin[1,2], YANG Zhenyu[1,2], SUN Yang[1,2], LIU Yawei[1,2], WANG Minghan[1,2]
(1. School of Control and Compute Engineering, North China Electric Power University, Baoding 071000 China; 2. Hebei Key Laboratory of Knowledge Computing for Energy & Power, North China Electric Power University, Baoding 071000, China)

Abstract: To solve the problems that the existing single-stage 3D object detection algorithm utilizes point cloud downsampling features in a single way and the degree of aggregation of features for the long-range contextual information cannot meet the requirement of enhancing the algorithm performance, we propose a single-stage 3D object detection algorithm based on multi-channel cross attention fusion. First, the channel-wise cross attention module is designed to fuse the down sampled features, which can enhance the expression ability of multi-scale features for the long-range spatial information under different receptive field based on the cross attention mechanism. Then, a cascade feature excitation module is proposed to combine the original downsampling features to cascade channel-wise cross attention weighted features to enhance the algorithm's learning ability for key spatial features. Extensive experiments were conducted on the public autonomous driving dataset KITTI and compared with mainstream algorithms. As a single-stage algorithm, the detection accuracy was 91.34%, 79.85% and 75.98% for the three difficulty levels of car categories, which were 4.83%, 3.26% and 3.32% better than the baseline algorithm. The experimental results demonstrate the effectiveness and advancement of the algorithm and the proposed modules for 3D object detection task.

Keywords: 3D point cloud; autonomous driving; LiDAR; deep learning; 3D object detection; pillar; cross attention; single-stage algorithm

三维目标检测是机器视觉的重要任务之一, 可以为自动驾驶、机器人导航和虚拟现实等应用场景提供丰富的环境目标信息[1]。由激光雷达采

收稿日期: 2023−05−16.　**网络出版日期:** 2024−06−04.
基金项目: 河北省重点研发计划项目 (20310103D); 河北省在读研究生创新能力培养资助项目 (CXZZBS2023153).
通信作者: 鲁斌. E-mail: lubin@ncepu.edu.cn.

集并生成的点云是目前描述空间环境三维信息的主要数据形式，其包含环境表面的三维坐标和反射率等信息，能够准确表达三维环境的关键特征，适用于三维目标检测任务。基于点云的三维目标检测旨在利用环境点云数据，提取目标空间特征，实现对环境的理解，最终输出准确包围场景中目标的三维边界框，包括类别、位置和朝向角度等信息[2]。

与二维图像中规则且密集排列的像素不同[3]，三维点云具有稀疏、无序和置换不变的特性，无法直接对其应用成熟的二维特征提取方法。同时，在点云场景中，被检测目标普遍存在被遮挡和被截断的情况，距离传感器较远的目标的点云会较为稀疏，且仅有朝向传感器的一侧包含点云[4]。因此，如何从稀疏点云中学习目标的有效特征并保持较高的推理速度，仍然是一个具有挑战的任务。

根据点云处理方式的不同，现有主流三维目标检测算法可以分为基于点的算法和基于体素的算法。基于点的算法通常使用 PointNet++[5] 作为处理点云的基础模块，在得到初始特征后输入下采样骨干网络进行多尺度特征提取和聚合，最后使用检测头预测候选框信息。PointRCNN[6] 引入最远点采样（furthest point sampling, FPS），使用 PointNet++对点云进行特征提取，并将点云分割为前景点和背景点后进行监督。文献 [7] 引入 Point-Pool 用于二阶段特征的细化，提高对点的利用效率。3DSSD[8] 提出空间特征采样方法直接处理点云，在下采样过程中充分保留目标内部点云，有效提升点特征聚合程度。Point-GNN[9] 直接使用原始点云构建图结构，并通过图采样进行下采样以实现特征聚合。IA-SSD[10] 提出了 2 个可学习的实例感知下采样策略以提升采样效率和有效性。文献 [11] 对集合抽象方法进行了改进，引入语义增强的集合抽象方法，在下采样过程中保留了更重要的前景点。基于点的方法可以充分利用原始点云信息，为预测框的生成和细化提供信息表达充分的特征。但是，其往往需要针对不同点云稀疏度的目标设计合适的点云采样方式，欠采样和过采样都会对检测效果造成影响[12]，是这类算法提升性能的瓶颈。

基于体素的算法则会将点云划分为规则堆叠的体素块，通过三维卷积进行特征提取[13]。首先，对点云空间进行过滤，并按照固定尺寸划分出的同等大小堆叠的三维立方体；然后，对体素内的点进行编码，通常取体素内点坐标的均值作为体素的特征；接着，使用三维卷积或二维卷积对得到的规则点云表达进行特征提取与特征聚合。基于体素的算法中，VoxelNet[14] 是先驱性的工作，其对点云进行体素化处理后，使用三维卷积得到体素的下采样特征，并在鸟瞰视角下基于下采样特征生成预测框。SECOND[15] 对 VoxelNet 进行了改进，提出了三维稀疏卷积以提高特征提取效率。两阶段的算法 Voxel R-CNN[16] 在第二阶段中引入多尺度体素特征池化，可以有效聚合感兴趣区域（region of interest, RoI）特征，提升细化过程对目标特征的学习能力。

基于体素的算法通常在 3 个维度上对点云空间进行划分，但也有一些算法将高度忽略，选择对点云空间进行柱体素化处理，生成规则的柱体素，以提高算法的推理效率。基于柱体素的算法可以方便地使用二维卷积进行特征提取，能够在计算效率上有较大幅度提升，易于实际部署。PointPillars[17] 是这类算法中的开创性工作，其首先将点云空间进行柱体素化处理并编码，再使用一个简单的自上而下结构提取特征，大幅提升了检测速率。PillarNet[18] 引入了三段式特征提取结构，应用二维稀疏卷积提升特征的聚合深度。文献 [19] 则将柱体素与多视图结合起来，使用无锚框的方法进行目标预测。相较于基于点的算法，体素化处理使得算法能够避免直接处理原始点云，大大降低对计算资源的需求。同时，体素化过程虽然会不可避免地导致原始点云空间信息的丢失，但能使算法获得较高的推理效率且更适用于实际应用场景。因此，挖掘体素中的潜在空间信息、增加体素特征对长程点间关系的表达，对于提升算法检测性能至关重要。

近些年，随着 Transformer[20] 在自然语言处理和计算机视觉等领域中的不断发展，一些研究者将其引入三维目标检测领域，为点云的特征提取和聚合方式提出新的设计范式。例如，Voxel Transformer[21] 引入局部注意力和可变形注意力，利用非空体素生成注意力加权特征，可以有效提升算法对关键特征的关注度。Voxel Set Transformer[1] 则引入由 2 个交叉注意力组成的集合注意力模块，有效增强算法对复杂体素特征的并行处理能力。CT3D[22] 在基于点的两阶段方法中引入通道自注意力模块，直接使用 RoI 内部的原始点云进行注意力编码，有效利用了目标内部的上下文信息，获得了较好的检测性能。Transformer 架构可以对具备长程上下文相关性的数据进行有效建模[23]，因此在全局特征信息的聚合上

有显著优势，且对数据的顺序性不敏感，适合处理稀疏且分布不均匀的点云数据[24]。然而，现有基于注意力机制的算法大都基于三维卷积提取局部特征[25]，且由于点数量较多，导致直接对点云应用注意力机制的效率较低，无法充分发挥注意力机制对长程上下文信息的聚合能力。

为了充分利用点云下采样特征中的潜在空间位置信息、基于注意力机制对点云场景中的长程上下文信息进行有效聚合，本文提出一个基于柱体素的单阶段三维目标检测算法。主要工作如下：

1) 提出通道交叉注意力（channel-wise cross attention, CCA）模块，基于注意力机制有效聚合不同层次的下采样特征，充分利用特征中潜在的上下文信息，提升关键特征间的关联程度；

2) 提出级联特征激励（cascade feature excitation, CFE）模块，级联结合原始下采样特征与注意力加权特征，增加特征信息交换范围，充分融合来自不同层次和感受野的特征，丰富特征的空间信息表达；

3) 在公共数据集 KITTI 上进行了广泛实验，并与主流检测算法进行对比，实验结果证明了本文算法及所提模块的有效性和先进性，在各个目标类别的检测结果上较基线算法有显著提升。

1　多通道交叉注意力融合算法

本文算法流程如图 1 所示。首先，对输入的原始点云进行预处理以及柱体素特征编码，生成柱体素特征后送入下采样骨干网络进行特征提取；然后，对最后两层下采样特征进行通道级分层，并对部分分层特征进行上采样以形成统一尺寸；然后，使用 CCA 模块对其进行处理，生成注意力加权特征；之后，结合原始下采样特征，在 CFE 模块中对注意力加权特征进行级联激励；最后，使用检测头生成检测结果，检测头由多层感知机和非极大值抑制模块组成。

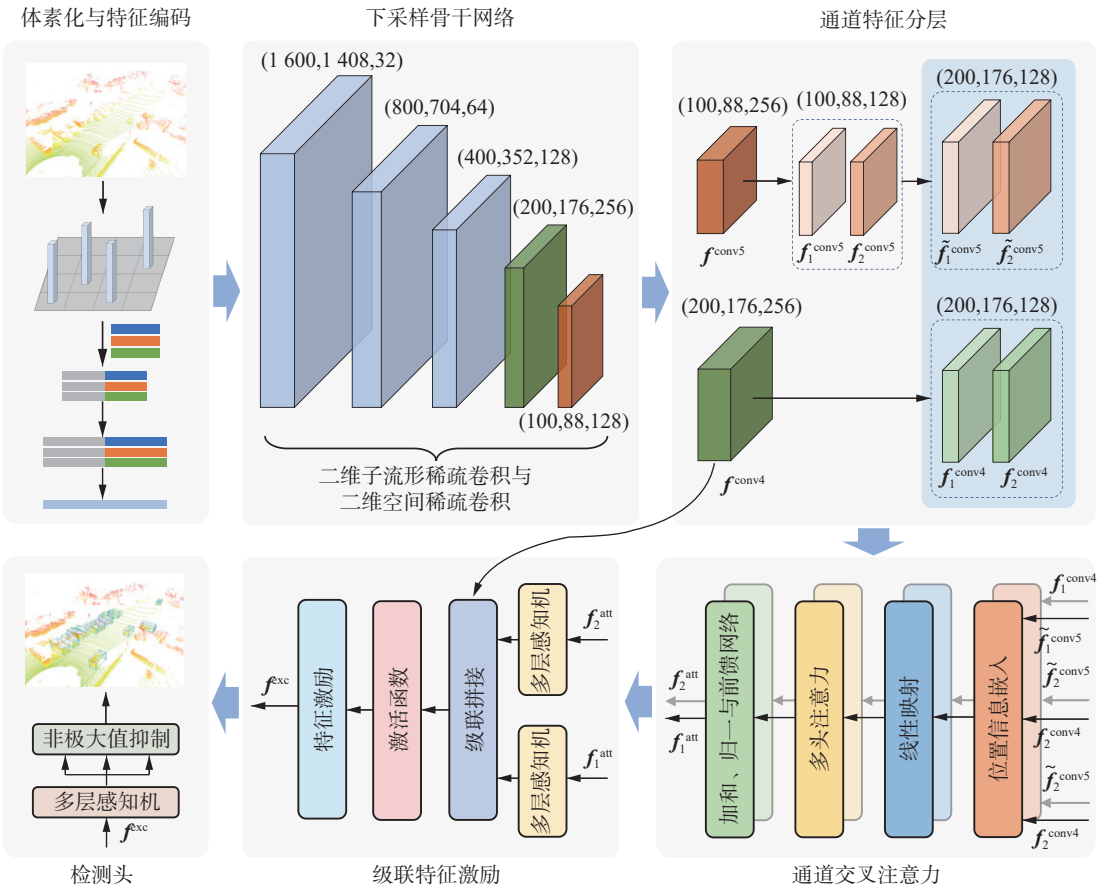


图 1　本文算法主要流程

Fig. 1　Main structure of the algorithm

1.1　点云预处理与柱体素特征编码

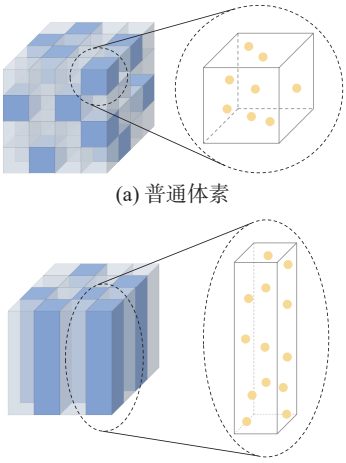
将三维点云定义为空间中点的集合，表示为

$$\boldsymbol{P}=\{\boldsymbol{p}_i|i=1,2,\cdots,n\}\in\mathbf{R}^{n\times 4}$$

式中：$\boldsymbol{p}_i=[x_i\ \ y_i\ \ z_i\ \ r_i]^{\mathrm{T}}$为点云场景中的点，$x_i$、$y_i$、$z_i$为点的三维坐标，$r_i$为点的反射率，$n$为点云场景中点的数量。首先，对输入的点云进行过滤处理

以形成规则空间。在x、y、z共 3 个方向分别设置上下限为$(x_{\min},x_{\max})$、$(y_{\min},y_{\max})$和$(z_{\min},z_{\max})$，超出范围的点会被丢弃。然后，在x和y方向上对点云空间进行均等划分，完成柱体素化，使用V_x和V_y表示柱体素在 2 个方向上的尺寸。每个点所属的柱体素在x和y方向上的坐标使用P_x和P_y表示。

由于对点云空间进行柱体素化处理，减少了一个特征维度，因此在后续处理中可以应用二维卷积代替三维卷积。同时，可以在充分利用原始点的空间位置信息的同时，降低后续流程的计算负载。而基于普通体素的算法通常会在点云预处理阶段根据设定的阈值对体素内的点进行丢弃，不可避免地丢失部分有用的空间信息，并对初始点云分布形成扰动。柱体素与普通体素的对比如图 2 所示。



(a) 普通体素

(b) 柱体素

图 2　普通体素与柱体素

Fig. 2　Ordinary voxels and pillars

对原始点云空间进行柱体素化处理之后，基于柱体素中所有的点对柱体素进行特征编码，形成可用于下采样的特征形式。首先，根据柱体素内所有点的坐标值、柱体素坐标以及对应方向的偏移量，得到偏移量特征，使用$\boldsymbol{c}_i=[d_i\ \ w_i\ \ h_i]^{\mathrm{T}}$表示。对应方向的偏移量计算公式为

$$d_i=x_i-(P_x\cdot V_x+D_{\text{offset}})$$

$$w_i=y_i-(P_y\cdot V_y+W_{\text{offset}})$$

$$h_i=z_i-H_{\text{offset}}$$

式中：d_i、w_i、h_i分别为在x、y、z共 3 个方向上每个点相对于柱体素质心的偏移量特征；D_{offset}、W_{offset}、H_{offset}为对应方向偏移量，由实际点云空间大小确定。

偏移量特征c_i包含了柱体素中所有点相对柱体素中心的空间位置关系，是重要的点云空间信息，对于算法准确建模点间上下文关系十分关键。将偏移量特征与原始点特征进行拼接后得到初始编码特征，使用$\boldsymbol{c}_i^{\text{ori}}=[x_i\ \ y_i\ \ z_i\ \ r_i\ \ d_i\ \ w_i\ \ h_i]^{\mathrm{T}}$表示。初始编码特征$\boldsymbol{c}_i^{\text{ori}}$是对原始点云空间的显式编码，为点云补充了空间位置信息。

为了增强特征表达，使用共享多层感知机(shared muti-layer perception, shared MLP)对初始编码特征进行升维，并对升维后的特征进行最大池化操作，得到最终的柱体素编码特征，表示为

$$\boldsymbol{f}_{\text{pillar}}=\left\{\tilde{\boldsymbol{c}}_t=[a_{t,1}\ \ a_{t,2}\ \ \cdots\ \ a_{t,m}]^{\mathrm{T}}\in\mathbf{R}^m\right\}_{t=1,2\cdots,T}$$

式中：m为特征维度，本文算法设置为 32；T为初始编码特征数量。shared MLP 由一个线性映射函数、一个批量归一化函数和一个 ReLU 激活函数组成，其权重值对所有初始编码特征共享，升维过程如图 3 所示。

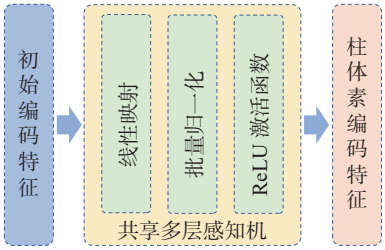


图 3　初始编码特征升维

Fig. 3　Process of changing initial encoding features

1.2　下采样与特征分层

不同于在二维空间中密集排列的像素，点在三维空间中的分布十分稀疏。直接使用常规的普通卷积对柱体素编码特征进行下采样会产生大量无意义的计算，严重降低算法的计算效率。基于柱体素化的点云处理方式，本文算法使用二维稀疏卷积作为下采样阶段的特征提取方法，由子流形稀疏卷积[26]和空间稀疏卷积[15]组合构成，能够有效减少算法的计算负载。子流形稀疏卷积可以在保持点云特征输入稀疏性的同时，对特征在维度和通道上进行变换。空间稀疏卷积则基于对非空体素位置的记录，实现仅对非空体素特征进行卷积，避免无效计算。

在下采样阶段，共使用 4 层二维稀疏卷积和 1 层普通二维卷积进行特征提取，主要参数如表 1 所示。卷积类型中，SubM表示子流形稀疏卷积，Spconv表示空间稀疏卷积，Conv2D表示普通卷积，“×2”表示使用 2 次。所有卷积操作的卷积核大小均为“3×3”。表 1 的输入与输出尺寸中，前 2 个数字表示特征图的长和宽，第 3 个数字表示通道数。下采样骨干网络中，第 1 层由 2 个子流形稀疏卷积构成，保持特征图大小和输出通道数不变；第 2~4 层则均由 1 个空间稀疏卷积和 2 个子流形稀疏卷积构成，特征图的大小依次成倍减

小，通道数依次成倍增加；第 5 层由 3 个普通卷积构成，特征图的大小再次减半，通道数则保持不变。

表 1　下采样骨干网络参数

Table 1　Down-sampling backbone network parameters

序号	卷积参数		输入尺寸	输出尺寸
	卷积类型	步长		
1	SubM	1	1 600、1 408、32	1 600、1 408、32
	SubM	1		
2	Spconv	2	1 600、1 408、32	800、704、64
	SubM×2	1		
3	Spconv	2	800、704、64	400、352、128
	SubM×2	1		
4	Spconv	2	400、352、128	200、176、256
	SubM×2	1		
5	Conv2D	2	200、176、256	100、88、256
	Conv2D×2	1		

下采样过程对柱体素特征进行了初步的提取和聚合，得到的下采样特征中蕴含了大量的空间位置信息。同时，为了减少注意力权重生成过程对计算资源的消耗，仅选择最后 2 层下采样特征作为 CCA 模块的输入，使用 $\boldsymbol{f}^{\text{conv4}}$、$\boldsymbol{f}^{\text{conv5}}$ 表示。$\boldsymbol{f}^{\text{conv4}}$ 与 $\boldsymbol{f}^{\text{conv5}}$ 的特征图的长、宽以及通道数分别为 (200,176,256) 和 (100,88,256)。首先，将 2 个特征分别在通道上进行均等划分，共形成 4 层特征，使用 $\boldsymbol{f}_1^{\text{conv4}}$、$\boldsymbol{f}_2^{\text{conv4}}$、$\boldsymbol{f}_1^{\text{conv5}}$、$\boldsymbol{f}_2^{\text{conv5}}$ 表示。其中 $\boldsymbol{f}_1^{\text{conv4}}$，和 $\boldsymbol{f}_2^{\text{conv4}}$ 尺寸同为 (200,176,128)，$\boldsymbol{f}_1^{\text{conv5}}$ 和 $\boldsymbol{f}_2^{\text{conv5}}$ 尺寸同为 (100,88,128)。接着，对 $\boldsymbol{f}_1^{\text{conv5}}$ 和 $\boldsymbol{f}_2^{\text{conv5}}$ 进行上采样，得到尺寸同为 (200,176,128) 的张量，使用 $\tilde{\boldsymbol{f}}_1^{\text{conv5}}$ 和 $\tilde{\boldsymbol{f}}_2^{\text{conv5}}$ 表示。上采样中使用的卷积核大小为 2，移动步长为 2。最后，将其输入所提出的 CCA 模块进行处理。需要说明的是，本节中所有卷积操作之后均会应用批量归一化函数与ReLU激活函数。

1.3　通道交叉注意力

柱体素较普通体素可保留更多点，其包含的空间信息也更为丰富。而相较于基于卷积的方式，注意力机制可基于更灵活的感受野对特征间的长程上下文依赖关系进行有效学习，对空间信息的利用也更加充分。

另一方面，点云场景中的点通常数量较多，直接对其应用注意力机制存在内存空间占用大和处理效率低下等问题。例如，CT3D[22] 提出的通道级 Transformer（channel-wise Transformer）模块仅作用于局部可能包含目标点云的区域以降低内存消耗，TANet[27] 提出的三重注意力（triple attention）模块则将通道级注意力（channel-wise attention）作为特征聚合时的补充，与其他类型的注意力堆叠使用。对于下采样特征的处理，一些算法提出使用交叉或分层操作以增加信息量。例如，CrossViT[28] 使用交叉注意力（cross attention）模块同时处理不同分支的输入，SMOKE[29] 则在生成检测框之前将特征按通道数分层。

受到以上算法启发，本文提出通道交叉注意力（CCA）模块，使用最后两层下采样特征作为注意力输入，基于注意力机制对点云特征中潜在的长程上下文信息进行充分提取。接着，对最后两层下采样特征在通道层面进行分层后交叉组合，各自使用对方的分层特征作为自己的查询变量，进一步发掘下采样特征中潜在的空间位置信息，增加不同聚合程度和不同尺度下点云特征的关联，扩大注意力机制的作用范围。CCA 模块主要由位置信息嵌入、线性映射、多头注意力、加和与归一化以及前馈网络构成，包含部分跳跃连接，如图 4 所示。

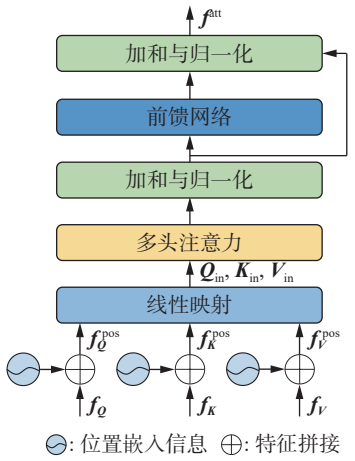


图 4　通道交叉注意力

Fig. 4　Channel-wise cross attention

将 $\tilde{\boldsymbol{f}}_1^{\text{conv5}}$ 和 $\boldsymbol{f}_2^{\text{conv4}}$ 作为第 1 组输入，其中上采样后的第 5 层分层特征 $\tilde{\boldsymbol{f}}_1^{\text{conv5}}$ 为查询输入，第 4 层分层特征 $\boldsymbol{f}_2^{\text{conv4}}$ 为键输入和值输入。将 $\boldsymbol{f}_1^{\text{conv4}}$ 和 $\tilde{\boldsymbol{f}}_2^{\text{conv5}}$ 作为第 2 组输入，对应位置的输入角色同第 1 组。如图 5 所示。由于后续处理方式相同，下文使用 $\boldsymbol{f}_Q$、$\boldsymbol{f}_K$、$\boldsymbol{f}_V$ 代表查询输入、键输入和值输入，作为统一注意力输入进行说明。

首先，对分层特征进行位置信息嵌入，得到对应特征 $\boldsymbol{f}_Q^{\text{pos}}$、$\boldsymbol{f}_K^{\text{pos}}$、$\boldsymbol{f}_V^{\text{pos}}$。再对特征进行线性映射，得到多头注意力输入，即 $\boldsymbol{Q}_{\text{in}} = \boldsymbol{W}_Q \otimes \boldsymbol{f}_Q^{\text{pos}}$，$\boldsymbol{K}_{\text{in}} = \boldsymbol{W}_K \otimes \boldsymbol{f}_K^{\text{pos}}$ 和 $\boldsymbol{V}_{\text{in}} = \boldsymbol{W}_V \otimes \boldsymbol{f}_V^{\text{pos}}$，其中，$\boldsymbol{W}_Q$、$\boldsymbol{W}_K$、$\boldsymbol{W}_V$ 为线性映射

矩阵；$\boldsymbol{Q}_{\text{in}}$、$\boldsymbol{K}_{\text{in}}$、$\boldsymbol{V}_{\text{in}}$为映射后的注意力输入特征，在维度上同$\boldsymbol{f}_{\boldsymbol{Q}}$、$\boldsymbol{f}_{\boldsymbol{K}}$、$\boldsymbol{f}_{\boldsymbol{V}}$一致，“⊗”表示矩阵乘法。然后，将$\boldsymbol{Q}_{\text{in}}$、$\boldsymbol{K}_{\text{in}}$、$\boldsymbol{V}_{\text{in}}$送入多头注意力，公式为

$$\mathcal{T}(\boldsymbol{Q}_{\text{in}},\boldsymbol{K}_{\text{in}},\boldsymbol{V}_{\text{in}})=\sigma\left(\frac{\boldsymbol{Q}_{\text{in}}(\boldsymbol{K}_{\text{in}})^{\text{T}}}{\sqrt{C}}\right)\otimes\boldsymbol{V}_{\text{in}}$$

式中：$\sigma(\cdot)$为Softmax激活函数；C为输入特征均分到每个注意力头中的特征维度，本文算法使用4个注意力头；$\mathcal{T}(\cdot)$表示注意力加权特征生成函数。接着，对多头注意力输出进行整合，公式为

$$\boldsymbol{f}^{\text{att}}=\sigma(\mathcal{A}(\mathcal{F}(\mathcal{A}(\mathcal{T}(\boldsymbol{Q}_{\text{in}},\boldsymbol{K}_{\text{in}},\boldsymbol{V}_{\text{in}})))))$$

式中：$\boldsymbol{f}^{\text{att}}$为最终的注意力加权特征；$\mathcal{A}(\cdot)$为加和与归一化操作；$\mathcal{F}(\cdot)$为前馈神经网络 (feed forward network, FFN)，由2个线性层和1个激活函数组成。

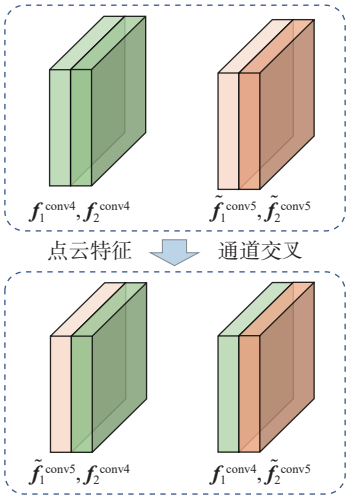


图 5　点云特征交叉组合

Fig. 5　Cross combination of point cloud features

经过CCA模块，生成2个注意力加权输出，使用$\boldsymbol{f}_1^{\text{att}}$和$\boldsymbol{f}_2^{\text{att}}$表示，分别和2组输入$\tilde{\boldsymbol{f}}_1^{\text{conv5}}$、$\boldsymbol{f}_2^{\text{conv4}}$与$\boldsymbol{f}_1^{\text{conv4}}$、$\tilde{\boldsymbol{f}}_2^{\text{conv5}}$对应。

1.4　级联特征激励

下采样特征经过注意力机制的作用后，聚合了全局信息和局部信息，增加了特征的表达能力。但由于数据类别分布不平衡，这一过程会造成少量原始分辨率下关键特征信息的丢失或模糊，例如点云稀疏、被遮挡目标的信息。为了保持和补充特征对目标空间信息的表达，提升算法对全局长程上下文信息的关注度，使用CFE模块融合原始下采样特征，对注意力加权特征进行级联激励。

CFE模块包含特征变换、级联拼接、激活函数和特征激励等内容，其中特征变换使用多层感知机完成，激活函数为ReLU函数，特征激励使用计算输入平方幂操作。CFE模块以下采样阶段的输出特征和两层注意力加权特征为作用对象，采取并行级联结构，如图6所示。

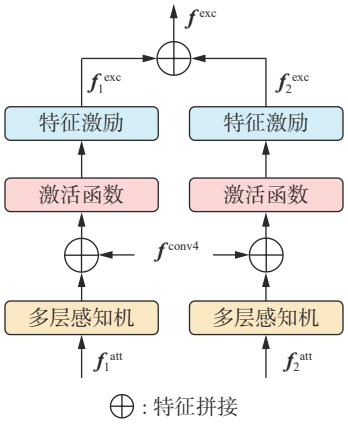


图 6　级联特征激励

Fig. 6　Cascade feature excitation

级联激励过程不改变特征维度大小，仅在拼接特征时增加通道数。级联激励特征公式为

$$\boldsymbol{f}_i^{\text{cascade}}=C\left(\text{MLP}(\boldsymbol{f}_i^{\text{att}}),\boldsymbol{f}^{\text{conv4}}\right)$$

$$\boldsymbol{f}_i^{\text{exc}}=\mathcal{P}\left(\mathcal{J}\left(\boldsymbol{f}_i^{\text{cascade}}\right)\right)$$

$$\boldsymbol{f}^{\text{exc}}=C(\boldsymbol{f}_1^{\text{exc}},\boldsymbol{f}_2^{\text{exc}})$$

式中：$\boldsymbol{f}_i^{\text{cascade}}$为级联输入，$\boldsymbol{f}_i^{\text{exc}}$为2个输入对应的中间激励特征，$i=1,2$，MLP(·)为多层感知机，$C(\cdot)$为特征拼接，$\mathcal{J}(\cdot)$为激活函数，$\mathcal{P}(\cdot)$为计算输入平方幂的操作，$\boldsymbol{f}^{\text{exc}}$为最终的激励特征。

为保证输入特征的信息量，CFE模块在激励注意力加权特征时使用$\boldsymbol{f}^{\text{conv4}}$作为级联拼接的对象之一。结合图1和图6可看出，$\boldsymbol{f}^{\text{conv4}}$经过分层交叉及注意力加权后与自身融合，实现了远程级联。同时，与CasA[30]、Cascade R-CNN[31]等算法的级联方式不同的是，CFE模块的核心是更为高效的相同模块的短程级联。因此，相较于普通的直接拼接，本文算法基于CFE模块同时实现了宏观与微观的级联拼接，并在级联过程中完成特征激励，对特征信息的保持与融合更加充分。此外，由于简洁的结构设计，CFE模块不会消耗太多计算资源，有着良好的迁移性，可以作为一个基础特征融合处理模块被使用。

1.5　损失函数

在检测头中，点云特征经过一个全连接层后作为输入分别送进预测框方向预测分支、类别置信度预测分支以及框位置预测分支中，用以计算方向损失、类别损失和位置损失。

使用$(x_B,y_B,z_B,l_B,w_B,h_B,\theta_B)$表示预测框的三维坐标、长宽高以及方向值，使用$(x_G,y_G,z_G,l_G,w_G,h_G,\theta_G)$表示真实标注框的三维坐标、长宽高以及方向值。计算预测框与真实标注框在各个维度上的差值用于生成损失，各维度差值公式为

$$\Delta x=\frac{x_B-x_G}{d_G},\Delta y=\frac{y_B-y_G}{d_G},\Delta z=\frac{z_B-z_G}{h_G},$$
$$\Delta l=\log\left(\frac{l_B}{l_G}\right),\Delta w=\log\left(\frac{w_B}{w_G}\right),\Delta h=\log\left(\frac{h_B}{h_G}\right),$$
$$\Delta\theta=\sin(\theta_B-\theta_G)$$

式中 $d_G=\sqrt{(w_G)^2+(l_G)^2}$。将各个维度的差值组合为 $\Delta g=(\Delta x,\Delta y,\Delta z,\Delta l,\Delta w,\Delta h,\Delta\theta)$，表示预测框与真实标注框之间的差距[15,17]。

为了平衡前后景样本分布不平衡的问题，使用 Focal Loss[32] 作为损失函数。计算公式为

$$L_{\text{cls}}=-\alpha(1-p_t)^{\gamma}\log(p_t)$$

式中：L_{cls}为类别损失；α和γ为权重参数，在算法训练中分别设置为 0.25 和 2；p_t为每个预测框的类别预测分数。

为了使用预测框与真实标注框的差值优化预测框位置，使用 SmoothL1 函数，计算公式为

$$L_{\text{loc}}=\text{SmoothL1}(\Delta g)$$

式中L_{loc}为预测框位置损失。

预测框与真实标注框在角度上的差值达到 180°时会导致预测框位置损失较大，不利于算法的正常训练。因此本文算法选择增加一个方向损失L_{dir}用于保证预测框方向的正确性，使用交叉熵函数实现，计算公式为

$$L_{\text{dir}}=-\sum_{j=1}^{M}\rho_j\log\left(\frac{\exp R_j}{\sum_{j=1}^{M}\exp R_j}\right)$$

式中：R_j为预测框方向的预测值；ρ_j为真实标注框方向的真实值；M为方向类别，本文设置为 2。

本文算法总损失的计算公式为

$$L_{\text{total}}=\beta_1L_{\text{loc}}+\beta_2L_{\text{cls}}+\beta_3L_{\text{dir}}$$

式中：β_1、β_2、β_3为 3 个损失的权重，与文献 [15] 保持一致，分别设置为 1.0、2.0 和 0.2。

2　实验结果与可视化分析

本节对算法的实现细节以及实验结果进行说明，包括实验环境、参数设置、实验结果、消融实验结果以及可视化结果等。本文算法在广泛使用的公共自动驾驶数据集 KITTI 进行训练与验证。该数据集主要包含激光雷达点云和视觉图像 2 种类型的数据。数据划分上，训练样本共 7 481 帧，测试样本共 7 518 帧。与文献 [15] 保持一致，将训练样本分为 3 712 帧的训练集和 3 769 帧的验证集。目标类别上，有车辆、行人以及自行车 3 种；难度级别上，有简单、中等和困难 3 种。

2.1　实验环境

本文算法的主要实验环境如表 2 所示。在训练过程中使用 2 块 RTX3090 GPU，在推理验证过程中使用 1 块 RTX3090 GPU，其他配置以及参数保持不变。在消融实验中，本文算法只更改被消融部分对应的参数，实验环境以及其他算法参数设置均同推理验证过程一致。

表 2　实验环境

Table 2　Experimental environment

环境	参数
CPU	Intel Xeon Gold 5218R
GPU	NVIDIA RTX 3 090
操作系统	Ubuntu 20.04
显存/GB	24
内存/GB	128
PyTorch	1.8.1
CUDA	11.3

2.2　参数设置

在点云预处理阶段，对x、y、z共 3 个方向的点云场景范围分别设置上下限为[0,70.4] m、[−40.0, 40.0] m 和[−3.0,1.0] m。在预测框 3 个方向的尺寸设置上，车辆类别为 3.90、1.60 和 1.56 m，行人类别为 0.80、0.60 和 1.73 m，自行车类别为 1.76、0.60 和 1.73 m。训练时使用 adam_onecycle 优化器，初始学习率设置为 0.003，批次大小为 8，训练轮数为 80。

使用预测框与真实标注框之间的交并比值（intersection over union, IoU）划分正负样本。对于车辆类别，大于 0.60 划为正样本，小于 0.45 划为负样本，丢弃其他预测框。对于其他 2 类，大于 0.50 划分正样本，小于 0.35 划分为负样本，丢弃其他预测框。柱体素在水平方向上的投影设置为正方形，边长为 0.075 m。使用常规的数据增强方法：1) 建立一个字典存放所有场景中的真实标注目标，随机取出并放置在点云场景中，增加场景中的真实标注目标数量，放置完成后进行物理碰撞监测，删除不符合物理规律的被放置目标；2) 对真实标注目标进行随机旋转和缩放，缩放倍数范围限制在 [0.95,1.05]，旋转角度限制在 [−π/4,+π/4]；3) 对真实标注目标进行随机转向，范围限制在[−π/2,+π/2]。

2.3　实验结果与分析

2.3.1　车辆类别检测结果

表 3 和表 4 分别给出了本文算法与主流单阶段和二阶段算法在 KITTI 数据集车辆类别上的检测结果对比，使用 KITTI 官方评价指标，其中

IoU 阈值为 0.7，采用 40 个召回位置计算准确率。在车辆类别的简单、中等与困难级别中，本文算法的检测准确率分别为 91.34%、79.85% 以及 75.98%。推理耗时对比结果如表 5 所示，本文算法的推理耗时为 47 ms，结果均在相同实验环境中生成，取一个推理周期内的平均值。推理时间是指算法在批量大小为 1 时，处理一帧点云场景所消耗的时间。

表 3　KITTI 数据集上与主流单阶段算法车辆检测结果对比

Table 3　Comparison of car object detection results of leading single stage algorithms on KITTI dataset　%

算法	车辆目标		
	简单	中等	困难
SA-SSD[33]	88.75	79.79	74.16
SIEV-Net[34]	85.21	76.18	70.60
SECOND[15]	83.34	72.55	65.82
MV3D[35]	74.97	63.63	54.00
3D-CenterNet[36]	86.83	80.17	75.96
VoxelNet[14]	77.47	65.11	57.73
CIA-SSD[37]	89.59	**80.28**	72.87
PointPillars[17]	82.58	74.31	68.99
PillarNet[18]	86.51	76.59	72.66
TANet[27]	85.94	75.76	68.32
HVNet[38]	87.21	77.58	71.79
本文算法	**91.34**	79.85	**75.98**

注：加黑为最高结果，下同。

表 4　KITTI 数据集上与主流两阶段算法车辆检测结果对比

Table 4　Comparison of car object detection results of leading two stage algorithms on KITTI dataset　%

算法	车辆目标		
	简单	中等	困难
PointRCNN[6]	86.96	75.64	70.70
BADet[39]	89.28	81.61	76.58
PV-RCNN[40]	90.25	81.43	76.82
Graph R-CNN[41]	91.29	**82.77**	77.20
Part-A2[42]	87.81	78.49	73.51
Voxel R-CNN[16]	90.90	81.62	77.06
STD[7]	87.95	79.71	75.09
Focals Conv[43]	90.20	82.12	**77.50**
CT3D[22]	87.83	81.77	77.16
FV2P[44]	88.53	81.58	77.37
本文算法	**91.34**	79.85	75.98

表 5　推理时间对比

Table 5　Comparison of inference time　ms

算法	推理时间
SECOND[15]	52
PointPillars[17]	28
PillarNet[18]	38
PV-RCNN[40]	131
Voxel R-CNN[16]	75
PointRCNN[6]	107
本文算法	47

结合表 3 和表 5 可知，与经典单阶段算法 VoxelNet 和 SECOND 相比，本文算法在车辆类别 3 个难度级别上的检测准确率均有较大幅度提升，其中困难级别的提升幅度最大，分别为 10.16% 和 18.25%，且在推理时间上较 SECOND 少用 5 ms。与同样基于柱体素的 PointPillars 相比，本文算法虽然在推理时间上有所增加，但仍在合理范围内，并且在车辆类别 3 个难度级别的检测准确率上均有显著提升，分别为 8.76%、5.54% 和 6.99%。这不仅说明了子流形稀疏卷积与空间稀疏卷积的有效性，也说明了对下采样特征进行充分挖掘的重要性，而本文提出的 CCA 模块的核心作用就是深入探索下采样特征中潜在的空间信息。

与基线算法 PillarNet 相比，本文算法在推理时间仅增加 9 ms的基础上，3 个难度级别的准确率分别提升了 4.83%、3.26% 以及 3.32%。这充分证明了本文提出的 2 个模块的有效性，说明了基于注意力机制对下采样特征进行长程上下文信息聚合的重要性，也说明了对注意力加权特征进行激励的必要性。而 PillarNet 仅对最后两层下采样特征基于二维卷积进行了简单地处理以及拼接，对特征中丰富的潜在空间位置信息的利用明显存在较大提升空间。

与 CIA-SSD 和 SA-SSD 相比，本文算法对车辆类别的检测准确率在简单与困难级别上分别提升 1.75% 和 2.59% 以及 3.11% 和 1.82%。这说明所提模块对简单目标的丰富特征和困难目标的模糊特征在提取和激励上是同等有效的。在中等级别目标的检测中，本文算法与 CIA-SSD 有 0.43% 的差距，仅比 SA-SSD 高出 0.06%。这说明虽然中等级别场景中的目标较困难目标有更多点且被遮挡情况也有所减少，但本文算法对中等目标特征的敏感度不够高，对其语义信息的学习能力仍有

提高空间。而 CIA-SSD 使用的空间语义特征融合模块对中等目标的关注度显然更高, 可以基于语义分组策略有效提取语义信息, 因此对中等目标特征的聚合更为有效, 在对比结果中表现最优。

由表 4 可看出本文算法仅在车辆类别的简单级别上较基于体素的 Voxel R-CNN 高出 0.44%, 但在推理时间上少用 28 ms, 而与基于点的 Point-RCNN 相比则在 3 个级别上分别高出 4.38%、4.21% 和 5.28%, 并在推理时间上少用 60 ms。这说明对点云进行柱体素化对于提升准确率和效率是十分有效的。本文算法较 Graph RCNN 仅在简单级别高出 0.05%, 中等和困难级别则有 2.92% 和 1.22% 的差距, 较 Focals Conv 则在简单级别高出 1.14%, 中等和困难级别则有 2.27% 和 1.52% 的差距。Graph R-CNN 在二阶段中构建了复杂的图结构用于提取池化特征, 获得的特征表达能力更强。而 Focals Conv 则同时使用点云和图像数据, 可为检测头提供信息更多的输入。本文算法由于缺少二阶段的特征细化, 除了推理速度的优势外, 相比整体性能较强的两阶段算法在简单级别上有一定的竞争力。

综合来看, 本文算法在车辆类别的检测中与主流单阶段算法相比仅增加少量推理时间, 而检测准确率提升较为明显。另一方面, 本文算法虽然较主流两阶段算法在检测性能上优势不够显著, 但在推理实时性上明显领先。最后, 综合表 3~5 可看出, 本文算法在检测准确率和效率之间实现了较好的平衡, 在中等和困难级别目标的检测上仍有提升空间。

2.3.2　行人与自行车类别检测结果分析

表 6 和表 7 分别给出了本文算法与主流单阶段和二阶段算法在 KITTI 数据集行人与自行车类别上的检测结果对比, 使用 KITTI 官方评价指标, 其中 IoU 阈值为 0.5, 采用 40 个召回位置计算准确率。在行人类别的检测中, 本文算法在 3 个难度级别上的检测准确率分别为 52.51%、47.72% 和 43.47%, 相较基线算法分别提升了 1.24%、3.17% 和 0.69%。本文算法的平均检测精度 (mean average precision, mAP) 为 47.90%, 较基线算法提升了 1.70%。这不仅说明本文算法对于中等级别的行人目标较为敏感, 因此检测准确率提升幅度较大, 还证明了本文算法对于不同难度级别的行人目标在检测上有较好的整体性。与其他主流算法相比, 本文算法在中等和困难级别的检测准确率较高, 简单级别的检测准确率也保持了一定的竞争力。

在自行车类别的检测中, 本文算法在 3 个难度级别上的检测准确率分别为 78.97%、58.69% 和 56.98%, 相较基线算法分别提升了 2.82%、0.83% 和 1.22%。本文算法的 mAP 为 64.88%, 较基线算法提升了 1.62%。这说明本文算法对基线算法在自行车类别上的提升更侧重于简单级别的目标, 证明了所提模块对特征中潜在空间信息的挖掘是有效的。在主流单阶段算法中, 本文算法对简单级别和困难级别的自行车目标检测效果最优, 证明了本文所提模块对提升算法学习长程上下文信息能力的有效性和必要性。而与主流二阶段算法相比, 本文算法在 3 个级别上的检测准确率也比较有竞争力。

表 6　KITTI 数据集上与主流算法行人检测结果对比

Table 6　Comparison of pedestrian object detection results of leading algorithms on KITTI dataset　%

类型	算法	行人目标		
		简单	中等	困难
单阶段	SECOND[15]	51.07	42.56	37.29
	PointPillars[17]	51.45	41.92	38.89
	PillarNet[18]	51.27	44.55	42.78
	STD[7]	**53.29**	42.47	38.35
二阶段	PointRCNN[6]	47.98	39.37	36.01
	Point-GNN[9]	51.92	43.77	40.14
	PV-RCNN[40]	52.17	43.29	40.29
	Part-A2[42]	53.10	43.35	40.06
	本文算法	52.51	**47.72**	**43.47**

表 7　KITTI 数据集上与主流算法自行车检测结果对比

Table 7　Comparison of cyclist object detection results of leading algorithms on KITTI dataset　%

类型	算法	自行车目标		
		简单	中等	困难
单阶段	SECOND[15]	70.51	53.85	46.90
	PointPillars[17]	77.10	58.65	51.92
	PillarNet[18]	76.15	57.86	55.76
	STD[7]	78.69	61.59	55.30
二阶段	PointRCNN[6]	74.96	58.82	52.53
	Point-GNN[9]	78.60	63.48	57.08
	PV-RCNN[40]	78.61	**63.71**	**57.65**
	Part-A2[42]	**79.17**	63.52	56.93
	本文算法	78.97	58.69	56.98

2.4　消融实验

2.4.1　CCA 模块输入特征消融实验

为了验证 CCA 模块中交叉使用分层下采样特征的有效性，对 CCA 模块的输入进行消融实验，结果如表 8 所示。检测类别为车辆，该实验中所有特征的尺寸和通道均变换为与f_1^{conv4}相同。

表 8　CCA 模块输入特征消融实验结果

Table 8　Results of the CCA module input feature ablation experiment

层数	输入特征	推理时间/ms	mAP/%
1	$\left(f_1^{\text{conv4}},f_2^{\text{conv4}}\right)$	42	80.98
1	$\left(\tilde{f}_1^{\text{conv5}},\tilde{f}_2^{\text{conv5}}\right)$	41	81.83
2	$\left(f_1^{\text{conv4}},\tilde{f}_2^{\text{conv5}}\right)\left(\tilde{f}_1^{\text{conv5}},f_2^{\text{conv4}}\right)$	47	82.39
2	$\left(f_1^{\text{conv4}},f_2^{\text{conv4}}\right)\left(\tilde{f}_1^{\text{conv5}},\tilde{f}_2^{\text{conv5}}\right)$	47	79.81
3	$\left(\tilde{f}_1^{\text{conv5}},f_2^{\text{conv3}}\right)\left(f_1^{\text{conv3}},f_2^{\text{conv4}}\right)\left(f_1^{\text{conv4}},\tilde{f}_2^{\text{conv5}}\right)$	55	71.37

由表 8 的前 2 行可以看出，在只使用单层下采样特征作为 CCA 模块的输入时，2 种输入下检测性能均有提升，且使用$\left(\tilde{f}_1^{\text{conv5}},\tilde{f}_2^{\text{conv5}}\right)$的检测效果较使用$(f_1^{\text{conv4}},f_2^{\text{conv4}})$提升了 0.85%。这说明注意力机制对下采样特征聚合的有效性，也说明注意力机制对聚合程度更高的特征有更好的接受能力。

由表 8 的第 4 行可以看出，同时使用这 2 层特征但不交叉时，算法检测性能的提升相对不够明显，且较单独使用时分别下降 1.17% 和 2.02%。这说明尽管输入更加丰富了，但不同聚合程度的下采样特征之间没有建立合适的关联，注意力机制无法对两层特征中的上下文信息进行有效建模，提升效果反而不如使用单层下采样特征。相反，从表 8 的第 3 行可以看出，对下采样特征在通道上进行交叉后重新组合作为输入，检测效果的提升是最为显著的。这说明对不同尺度的下采样特征在通道上进行交叉后进行融合可以充分发挥注意力机制对长程上下文信息的提取能力，从而为算法在检测目标时提供上下文信息更加全面的点云特征，进一步提升算法检测性能。

由表 8 的最后 1 行可以看出，尽管在增加输入特征层数后进行了交叉，但检测性能显著下降。这是由于新增加的第 3 层下采样特征聚合程度太低，而注意力机制需要更为抽象的底层特征进行作用。

2.4.2　主要模块消融实验

对本文算法的 2 个主要模块 CCA 与 CFE 进行消融实验，对比 2 个模块对算法在检测性能上的影响，使用对车辆类别下 3 个难度级别的 mAP 作为对比依据，结果如表 9 所示。表 9 的第 1 行结果为基线算法的 mAP。表 9 中，“C.C.A”与“C.F.E”分别表示通道交叉注意力模块和级联特征激励模块，“×”和“√”分别表示“使用对应模块”与“不使用对应模块”。

表 9　CCA 模块与 CFE 模块的消融实验结果

Table 9　Results of ablation experiments with the CCA module and the CFE module

C.C.A	C.F.E	mAP/%
×	×	78.59
√	×	81.24
×	√	79.85
√	√	82.39

可以看到，本文提出的 2 个模块对算法检测性能的提升均较为明显，在各自单独使用的情况下分别较基线算法提升了 2.65% 和 1.26%，证明了 2 个模块的有效性。同时，单独使用 CCA 模块较单独使用 CFE 模块提升了 1.39%，说明注意力机制在对算法检测性能的提升上较激励机制有更大的优势。同时使用 2 个模块后，相较基线算法提升了 3.80%，提升幅度相对其他结果较大，证明了组合使用 2 个模块的有效性。综合表 9 的结果可以看出，在对下采样特征基于注意力机制进行作用后，结合原始下采样特征进行激励能够明显提升算法对全局上下文信息的聚合能力，并增加对关键特征的关注度，进而提高检测性能。

2.5　可视化分析

将本文算法在点云场景中的检测结果可视化，分析算法的实际检测效果，如图 7 所示。图 7 中每张小图由上至下分 3 个部分，第 1 部分为点云场景对应的实际图像，第 2 部分为放置了预测框与真实标注框的点云场景，第 3 部分为加入了真实标注框投影的实际图像。其中，第 2 部分中的绿色框为本文算法给出的预测框，蓝色框为真实标注框，第 3 部分中的红色框为真实标注框的二维投影。需要说明的是，第 1 部分的实际图像受限于视角，对场景展示不够全面，第 2 部分的点云则较为全面，包含了更多的车辆目标以及对应场景。

图 7(a) 是街区道路场景，可以看到本文算法不仅对距离传感器较近的车辆目标实现了准确检测，对于远处点云稀疏的车辆目标也完成了准确检测，还检测到了遮挡较为严重且没有真实标注框的车辆目标，证明了本文算法的有效性。图 7(b)

是公路场景，车辆较多且遮挡严重，远处车辆目标点云更为稀疏，可以看到本文算法同样实现了对视野范围内车辆目标的准确检测。图 7(c) 城区道路场景与图 7(a) 同为城市内的道路场景，但图 7(c) 中车辆目标的方向与图 7(a) 中车辆目标相垂直，且排列更为密集，遮挡情况更为严重。可以看到本文算法对密集排列的车辆目标实现了准确检测，还检测到了更远处点云稀疏的目标。

(a) 街区道路场景　(b) 公路场景　(c) 城区道路场景

图 7　算法检测结果可视化

Fig. 7　Visualisation of algorithm detection results

3　结束语

针对现有单阶段算法对下采样特征处理方式单一、特征对长程上下文信息表达不够充分的问题，本文提出了一个基于多通道注意力特征融合思想的三维目标检测算法。基于注意力机制提出通道交叉注意力模块，将最后两层下采样特征进行交叉重组后作为注意力输入，生成对全局上下文信息聚合程度更高的点云特征，提升算法对被遮挡目标和点云稀疏目标的识别程度。基于激励机制提出级联特征激励模块，将原始下采样特征与注意力特征进行结合后进行激励，增加点云特征对关键位置信息的表达，提升算法的检测性能。

本文算法在 KITTI 数据集上进行了广泛实验并与主流算法对比，实验结果表明检测性能较基线算法有明显提升，证明了本文算法及 2 个模块的有效性。另一方面，综合消融实验结果来看，本文算法对下采样特征中潜在上下文信息的提取效率仍可以继续提升，对柱体素中空间信息的保留和表达也有一定的优化空间。因此，之后的研究重点是对本文提出的算法进行进一步改进，在保持算法鲁棒性的同时提升算法对下采样特征的利用效率，优化柱体素特征编码过程，进而提高算法的检测性能。

参考文献：

[1] HE Chenhang, LI Ruihuang, LI Shuai, et al. Voxel set transformer: a set-to-set approach to 3D object detection from point clouds[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 8417−8427.

[2] 张新钰, 邹镇洪, 李志伟, 等. 面向自动驾驶目标检测的深度多模态融合技术 [J]. 智能系统学报, 2020, 15(4): 758–771.
ZHANG Xinyu, ZOU Zhenhong, LI Zhiwei, et al. Deep multi-modal fusion in object detection for autonomous driving[J]. CAAI transactions on intelligent systems, 2020, 15(4): 758–771.

[3] 王凤随, 陈金刚, 王启胜, 等. 自适应上下文特征的多尺度目标检测算法 [J]. 智能系统学报, 2022, 17(2): 276–285.
WANG Fengsui, CHEN Jingang, WANG Qisheng, et al. Multi-scale target detection algorithm based on adaptive context features[J]. CAAI transactions on intelligent systems, 2022, 17(2): 276–285.

[4] 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.

[5] QI Charles R, SU Hao, MO Kaichun, et al. PointNet: deep learning on point sets for 3d classification and segmentation[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 652−660.

[6] SHI Shaoshuai, WANG Xiaogang, LI Hongsheng. Pointrcnn: 3D object proposal generation and detection from point cloud[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019: 770−779.

[7] YANG Zetong, SUN Yanan, LIU Shu, et al. STD: sparse-to-dense 3D object detector for point cloud[C]//2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019: 1951−1960.

[8] YANG Zetong, SUN Yanan, LIU Shu, et al. 3DSSD: point-based 3D single stage object detector[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 11040−11048.

[9] SHI Weijing, RAJKUMAR R. Point-GNN: graph neural network for 3D object detection in a point cloud[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 1711−1719.

[10] ZHANG Yifan, HU Qingyong, XU Guoquan, et al. Not all points are equal: learning highly efficient point-based detectors for 3D lidar point clouds[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 18953−18962.

[11] CHEN Chen, CHEN Zhe, ZHANG Jing, et al. SASA: semantics-augmented set abstraction for point-based 3D object detection[C]//2022 AAAI Conference on Artificial Intelligence. Vancouver: IEEE, 2022, 36(1): 221−229.

[12] GUO Yulan, WANG Hanyun, HU Qingyong, et al. Deep learning for 3D point clouds: a survey[J]. IEEE transactions on pattern analysis and machine intelligence, 2020, 43(12): 4338–4364.

[13] XIAO Aoran, HUANG Jiaxing, GUAN Dayan, et al. Unsupervised point cloud representation learning with deep neural networks: a survey[J]. IEEE transactions on pattern analysis and machine intelligence, 2023, 45(9): 11321–11339.

[14] ZHOU Yin, TUZEL O. VoxelNet: end-to-end learning for point cloud based 3D object detection[C]//2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 4490−4499.

[15] YAN Yan, MAO Yuxing, LI Bo. Second: sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10): 3337.

[16] DENG Jiajun, SHI Shaoshuai, LI Peiwei, et al. Voxel R-CNN: towards high performance voxel-based 3D object detection[C]//2021 AAAI Conference on Artificial Intelligence. [S.l.]: IEEE, 2021, 35(2): 1201−1209.

[17] 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. Long Beach: IEEE, 2019: 12697−12705.

[18] SHI Guangsheng, LI Ruifeng, MA Chao. Pillarnet: real-time and high-performance pillar-based 3D object detection[C]//2022 European Conference on Computer Vision. Tel Aviv: Springer, 2022: 35−52.

[19] WANG Yue, FATHI A, KUNDU A, et al. Pillar-based object detection for autonomous driving[C]//2020 European Conference on Computer Vision. Glasgow: Springer, 2020: 18−34.

[20] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30: 6000–6010.

[21] MAO Jiageng, XUE Yujing, NIU Minzhe, et al. Voxel transformer for 3D object detection[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 3164−3173.

[22] SHENG Hualian, CAI Sijia, LIU Yuan, et al. Improving 3D object detection with channel-wise transformer[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 2743−2752.

[23] 吴军, 崔玥, 赵雪梅, 等. SSA-PointNet++: 空间自注意力机制下的 3D 点云语义分割网络 [J]. 计算机辅助设计与图形学学报, 2022, 34(3): 437–448.
WU Jun, CUI Yue, ZHAO Xuemei, et al. SSA-PointNet++: a space self-attention CNN for the semantic segmentation of 3D point cloud[J]. Journal of computer-aided design & computer graphics, 2022, 34(3): 437–448.

[24] GUO Menghao, XU Tianxing, LIU Jiangjiang, et al. Attention mechanisms in computer vision: a survey[J]. Computational visual media, 2022, 8(3): 331–368.

[25] LU Dening, XIE Qian, WEI Mingqiang, et al. Transformers in 3D point clouds: a survey[EB/OL]. (2022−09−21) [2023−05−16]. https://www.arxiv.org/abs/2205.07417v2.

[26] GRAHAM B, ENGELCKE M, MAATEN L. 3D semantic segmentation with submanifold sparse convolutional networks[C]//2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 9224−9232.

[27] LIU Zhe, ZHAO Xin, HUANG Tengteng, et al. Tanet: robust 3D object detection from point clouds with triple attention[C]//2020 AAAI Conference on Artificial Intelligence. New York: IEEE, 2020, 34(7): 11677-11684.

[28] CHEN Chunfu, FAN Quanfu, PANDA R. Crossvit: cross-attention multi-scale vision transformer for image classification[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 357−366.

[29] LIU Zechen, WU Zizhang, TÓTH R. Smoke: single-stage monocular 3D object detection via keypoint estimation[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. Seattle: IEEE, 2020: 996−997.

[30] WU Hai, DENG Jinhao, WEN Chenglu, et al. CasA: a

cascade attention network for 3-D object detection from LiDAR point clouds[J]. IEEE transactions on geoscience and remote sensing, 2022, 60: 1–11.

[31] CAI Zhaowei, VASCONCELOS N. Cascade R-CNN: high quality object detection and instance segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 2019, 43(5): 1483–1498.

[32] LIN T, GOYAL P, GIRSHICK R, et al. Focal loss for dense object detection[C]//2017 IEEE International Conference on Computer Vision. Venice: IEEE, 2017: 2980−2988.

[33] HE Chenhang, ZENG Hui, HUANG Jianqiang, et al. Structure aware single-stage 3D object detection from pointcloud[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 11873−11882.

[34] YU Chuanbo, LEI Jianjun, PENG Bo, et al. SIEV-Net: a structure-information enhanced voxel network for 3D object detection from LiDAR point clouds[J]. IEEE transactions on geoscience and remote sensing, 2022, 60: 1–11.

[35] CHEN Xiaozhi, MA Huimin, WAN Ji, et al. Multi-view 3D object detection network for autonomous driving[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 1907−1915.

[36] WANG Qi, CHEN Jian, DENG Jianqiang, et al. 3D-CenterNet: 3D object detection network for point clouds with center estimation priority[J]. Pattern recognition, 2021, 115: 107884.

[37] ZHENG Wu, TANG Weiliang, CHEN Sijin, et al. CIA-SSD: confident IoU-aware single-stage object detector from point cloud[C]//2021 AAAI Conference on Artificial Intelligence. [S.l.]: IEEE, 2021, 35(4): 3555−3562.

[38] YE Maosheng, XU Shuangjie, CAO Tongyi. HvNet: hybrid voxel network for lidar based 3D object detection[C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 1631−1640.

[39] QIAN Rui, LAI Xin, LI Xirong. BADet: boundary-aware 3D object detection from point clouds[J]. Pattern recognition, 2022, 125: 108524.

[40] SHI Shaoshuai, GUO Chaoxu, JIANG Li, et al. PV-RCNN: point-voxel feature set abstraction for 3D object detection[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Seattle: IEEE, 2020: 10529−10538.

[41] YANG Honghui, LIU Zili, WU Xiaopei, et al. Graph R-CNN: towards accurate 3D object detection with semantic-decorated local graph[C]//2022 European Conference on Computer Vision. Tel Aviv: Springer, 2022: 662−679.

[42] SHI Shaoshuai, WANG Zhe, SHI Jianping, 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, 2020, 43(8): 2647–2664.

[43] CHEN Yukang, LI Yanwei, ZHANG Xiangyu, et al. Focal sparse convolutional networks for 3D object detection[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 5428−5437.

[44] LI Jiale, DAI Hang, SHAO Ling, et al. From voxel to point: iou-guided 3D object detection for point cloud with voxel-to-point decoder[C]//2021 ACM International Conference on Multimedia. New York: ACM, 2021: 4622−4631.

作者简介：

鲁斌，教授，博士，博士生导师，CCF 高级会员，主要研究方向为智能计算与计算机视觉、综合能源系统与大数据分析。E-mail：lubin@ncepu.edu.cn。

杨振宇，博士研究生，主要研究方向为机器学习、计算机视觉。E-mail：yangzhenyu536@163.com。

孙洋，博士研究生，主要研究方向为机器学习、计算机视觉。E-mail：bless2016@163.com。