基于点云稀疏空间特征聚合激励的单阶段 3D 目标检测模型

鲁斌[1,2)], 孙洋[1,2)], 杨振宇[1,2)]

[1)] (华北电力大学控制与计算机工程学院 保定 071000)
[2)] (复杂能源系统智能计算教育部工程研究中心 保定 071000)
(lubin@ncepu.edu.cn)

摘 要: 针对目前基于点云的 3D 目标检测中单阶段体素法存在感受野固定、特征尺度单一, 导致模型对点云特征学习不够充分、模型检测效果存在瓶颈等问题, 提出了一种可端对端训练的基于体素的单阶段 3D 目标检测模型. 首先, 利用多尺度稀疏空间特征聚合模块, 聚合点云在不同稀疏空间尺度上的特征, 使特征充分保留点云的空间信息; 然后, 对特征进行分层激励, 通过多尺度感受野对特征进行分层学习, 强化特征的表达能力, 降低噪声信息对检测结果的影响; 最后, 将特征输入检测头进行候选框的分类和回归. 在公开的自动驾驶数据集 KITTI 上与主流单阶段 3D 目标检测模型进行了对比实验, 包含对 3 类目标共 9 个的难度等级目标的检测. 所提模型在其中 5 个等级中的平均准确率有明显提升, 尤其对点云稀疏的目标, 表现出较好的检测效果. 实验结果表明, 所提模型能够充分提取点云空间信息并有效地学习点云多尺度特征.

关键词: 3D 目标检测; 激光雷达点云; 多尺度稀疏空间特征聚合; 分层激励
中图法分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2024.19850

Aggregate and Excitate Sparse Spatial Feature: Single-Stage 3D Object Detector from Point Clouds

Lu Bin[1,2)], Sun Yang[1,2)], and Yang Zhenyu[1,2)]

[1)] (*School of Control and Compute Engineering, North China Electric Power University, Baoding* 071000)
[2)] (*Engineering Research Center of Intelligent Computing for Complex Energy Systems, Ministry of Education, Baoding* 071000)

Abstract: In order to solve the problems of fixed receptive field and single feature scale in the single-stage voxel-based 3D object detectors, which cause insufficient learning of point clouds features and bottleneck in detection performance, a voxel-based single-stage 3D object detection model is proposed which can be trained end-to-end. First, the multi-scale sparse spatial feature aggregation module is used to aggregate the features of point clouds at different sparse spatial scales, so that the features can fully preserve the spatial information of point clouds. Then, the detector performs hierarchical learning of features through multi-scale receptive fields by feature hierarchical excitation module, which can strengthen the representation of the features, and reduce the influence of noise information to make the detector more robust. Finally, the features are fed into the detection head for classification and regression of candidate boxes. Extensive experiments on the public autonomous driving dataset KITTI compared with other mainstream single-stage 3D object detectors, including 3 categories of 9 difficulty levels, demonstrate that, the average accuracy of the proposed model is significantly improved in 5 levels, especially for objects with sparse point clouds. The experimental results show that the proposed model can fully extract the spatial information and effectively

收稿日期: 2022-07-25; 修回日期: 2022-11-09. **基金项目**: 河北省重点研发计划(20310103D). **鲁斌**(1975—), 男, 博士, 教授, 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; **孙洋**(1991—), 男, 博士研究生, 主要研究方向为机器学习、计算机视觉; **杨振宇**(1998—), 男, 硕士研究生, 主要研究方向为机器学习、计算机视觉.

learn the multi-scale features of the point clouds.

Key words: 3D object detection; light detection and ranging point clouds; multi-scale feature aggregation; hierarchical excitation

1 相关工作

目前, 3D 目标检测技术正处于快速发展时期, 在自动驾驶、机器视觉等场景中有重要应用, 是计算机视觉和模式识别领域的核心问题之一. 如何高效利用点云数据中的空间信息, 并从中提取可靠的特征来进行 3D 目标检测, 是当下国内外研究的重要课题.

在早期研究中, 人们使用包括几何、密度和尺度不变特征变换[1]等手工设计的特征进行目标检测, 并取得了一定的效果. 但由于其无法表示更为抽象的特征, 导致模型的泛化能力较差. 后来, 有研究者选择不直接处理原始点云, 而是将其转换为其他表示形式进行处理. 例如, 文献[2]将 3D 结构的点云数据投影到鸟瞰视角或其他视角, 进而应用相对成熟的 2D 卷积进行特征提取; 还有的研究者将点云转换为密集体素表示, 从而使用 3D 卷积进行特征提取. Qi 等[3-4]提出的 PointNet[3]和 PointNet++[4]是深度学习在点云研究中的开山之作, 此后, 相关研究工作不断涌现.

基于点的模型直接处理点云中的点. PointNet 则选择直接从点中学习数据的高维特征, 用于点云的分类和分割. 文献[5]基于深度霍夫投票思想, 通过多层感知机(multilayer perceptron, MLP)生成靠近目标中心的候选点后, 对其进行分组聚合并生成候选框; 文献[6]将点云分割为前景点和背景点, 为每一个前景点预测一个 3D 锚框, 计算点云的全局语义特征并将其拼接到前景点的特征值上进行预测, 以此获得更精确的锚框预测位置; 文献[7]使用球形锚框替代长方体锚框, 通过使用 3D 交并比(intersection over union, IoU)来划分正负样本, 提高了候选框的质量; 文献[8]改进了点采样方式, 即通过特征距离采样来获得更多的前景点, 并聚合周围的采样点对中心点进行估计, 以实现对预测框的分类与回归; 文献[9]通过多尺度采样, 在每一层采样点上应用扩张感受野来提取更多空间局部信息, 使锚框位置预测更加准确.

基于体素的模型将非结构化的点云转换为规则的体素网格进行处理. 文献[10]通过将点云转换为体素, 使用 3D 卷积对体素进行下采样, 然后将其映射到鸟瞰视角来实现 3D 候选框的生成; SECOND[11]通过改进 3D 卷积过程, 使用 3D 稀疏卷积以减少对空体素的无效计算, 使计算效率得到提高; PointPillars[12]将点云转换为柱状体素, 直接将特征压缩到鸟瞰视角后生成候选框, 省去 3D 卷积过程, 从而提高对点云的处理效率; 文献[13]选择从体素特征中提取感兴趣区域(region of interest, RoI)特征, 用于对锚框位置进行精细调整; 文献[14]通过堆叠多层注意力模块强化特征表达能力, 提高单阶段模型检测效果.

基于融合体素和点的模型结合了体素和点模型各自的优势. PV-RCNN[15]先通过提取体素特征来生成候选框, 然后逐点提取特征用于候选框的二阶段调整, 提高了检测准确率; 文献[16]改进 PV-RCNN 的点采样过程和点云局部特征编码方法, 即通过在采样点附近构造体素格对采样点进行池化, 接着对体素格按顺序拼接, 得到点云的局部相对位置特征; 文献[17]引入高效的内存管理模块来提高点的特征提取效率, 从而在准确性和效率之间实现更好的平衡.

基于多视角的模型通过将点云映射到不同的 2D 视角后, 使用 2D 目标检测范式进行目标检测. 文献[18]融合点云的俯视图、前视图和 RGB 色彩前视图像 3 个角度的信息进行目标的定位和识别; 文献[19]则从点云的俯视图中提取 3D 锚框, 使用特征金字塔(feature pyramid networks, FPN)[20]融合多视角特征并对锚框进行回归和分类, 通过改进锚框编码方式, 从而获得更高的编码效率; 文献[21]选择直接在鸟瞰视角下进行检测, 并将高度信息保留到颜色通道中, 以简化 3D 目标检测问题.

基于融合图像和点云的模型使用图像的 RGB 像素值和点云中的点坐标值共同进行检测. 文献[22]使用 2D 目标检测模型对前视图像进行分割, 并在初步找到目标位置后将位置信息映射到点云中, 以得到包含目标物体的视锥空间来进行后续的检测; 文献[23]通过把点云映射到被分割的 2D 图像上为点云增加分割信息, 然后进行目标检测.

以往的模型尽管取得了一定效果, 但仍然具

有改进空间. 基于点的模型需要对点云进行采样，分层聚合点的局部特征，效率较低，实时检测困难；基于体素的模型的问题是将点云体素化后会带来空间信息的损失，且感受野固定，对点云特征学习不够充分；基于融合体素和点的模型难以在速度和效率之间实现很好的平衡；基于多视角的模型会损失空间信息，给提升模型检测准确率带来困难；基于融合图像和点云的模型推理速度较慢，且检测结果易受 2D 图像处理结果的影响.

为解决上述问题，本文提出一个基于原始点云的高效单阶段 3D 目标检测模型，可以充分利用点云的空间特征进行聚合和分层激励，取得比其他使用单一尺度特征进行检测的模型更好的检测效果. 本文通过聚合多尺度稀疏空间特征，强化特征的表达能力来提升模型检测的准确率，提出多尺度稀疏空间特征聚合(multi-scale sparse spatial feature aggregation, MSS-FA)模块，使用 3D 稀疏卷积对点云进行特征提取，聚合多尺度稀疏空间特征，在提高特征聚合效率的同时，增强特征对点云空间信息的表达能力；提出分层激励(feature hierarchical excitation, FHE)模块，设计多通道特征提取路径，可以有效地学习目标不同感受野上的特征，解决单一感受野对目标描述不充分的问题，提升特征表达能力，使模型更加鲁棒. 通过上述模块，提升单阶段体素法 3D 目标检测的准确率，尤其使单阶段 3D 目标检测模型对 KITTI 数据集[24]困难样本的检测准确率得到显著的提高.

2　本文模型

本文的模型为可端对端训练的基于体素的单阶段 3D 目标检测模型，框架如图 1 所示. 模型以原始点云为输入，首先将点云进行预处理和体素化，并对体素进行编码；然后使用 MSS-FA 模块对 3D 体素空间进行多尺度特征提取和聚合，引入 3D 稀疏卷积对 3D 空间进行下采样，聚合多尺度稀疏空间特征，以获得丰富的空间语义信息；接着通过 FHE 模块使用具有不同感受野的 3 个卷积通道对特征进行强化；最后用得到的特征经检测头来完成预测框的回归和分类.

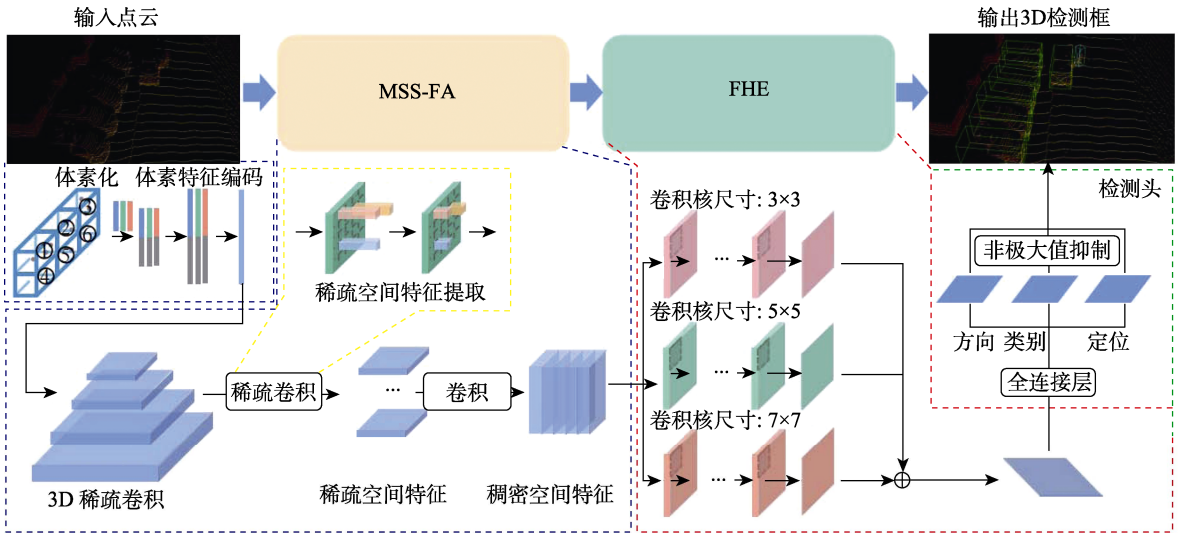


图 1　本文模型框架示意图

2.1　点云预处理阶段

在点云预处理阶段，首先对输入的点云进行过滤. 为限制点云采样空间的大小，只保留点云沿 x，y，z 方向上指定范围 $(x_{\min},x_{\max})$，$(y_{\min},y_{\max})$，$(z_{\min},z_{\max})$ 内的点，超出范围的点则被舍弃. 然后过滤点数量小于阈值 t 的前景物体. 反射点太少的物体被认为所包含的信息不足，无法表达物体的空间几何信息，以至于难以对其进行识别，因此这些物体不会被标记为真实值.

2.2　点云体素化阶段

点云具有无序、不规则、置换不变的特点，无法使用传统卷积操作对其进行特征提取. 体素法首先将不规则的点云空间划分为均匀堆叠的体素，然后模型以体素为最小单位处理点云，通过 3D 卷积对点云进行特征提取，相比于直接对点云中的点进行处理，这种处理方式具有更高的效率. 具体来说，定义每个体素在 x，y，z 方向上的尺寸分别为 V_x，V_y，V_z，则在 x，y，z 方向上的体素格数量

分别为

$$D=\lfloor(x_{\max}-x_{\min})/V_x\rfloor, W=\lfloor(y_{\max}-y_{\min})/V_y\rfloor,$$
$$H=\lfloor(z_{\max}-z_{\min})/V_z\rfloor.$$

其中，D,W 和 H 分别为深度、宽度和高度方向上体素格的数量. 在训练阶段, 体素格数量被限制最多为 16 000 个.

给定原始点云 $p_n=(x_n,y_n,z_n,r_n)$，其中，x_n，y_n，z_n 为点的空间坐标，r_n 为点的反射率信息. 将过滤后的点放到对应体素格内, 则点 p_n 对应的体素格坐标为 $D_n=\lfloor x_n-x_{\min}/V_x\rfloor$，$W_n=\lfloor(y_n-y_{\min})/V_y\rfloor$，$H_n=\lfloor(z_n-z_{\min})/V_z\rfloor$.

虽然模型已经在预处理阶段按照范围对点云进行过滤, 但空间中的点仍然很多, 且点在空间中的分布不是均匀的, 在有目标的地方点会表现得密集, 没有目标的地方则会稀疏. 点云体素化后, 每个体素格内包含的点数量是不同的. 为平衡体素之间点的数量差别并降低体素编码的计算量, 对每个体素格内包含的点的数量进行限制. 如果体素格内的点数量超过阈值 T，则随机采样 T 个点, 未被采样到的点会被丢弃. 阈值 T 是一个经验值, 如果 T 设置得过大, 则难以达到平衡体素格之间点数量的效果, 而且会降低体素的编码效率; 相反, 如果阈值 T 设置得太小, 则会丢失过多的点, 导致目标包含点的数量过少, 从而降低模型的检测准确率. 为平衡编码效率和检测准确率, 本文设置 $T=5$.

2.3 体素编码阶段

体素特征编码(voxel feature encoder, VFE)[10]是一种常见的体素编码方法. 首先, 定义非空体素

$$\boldsymbol{V}=\left\{\boldsymbol{p}_i=\left[x_i,y_i,z_i,r_i\right]^{\mathrm{T}}\in\mathbb{R}^4\right\},\ i=1,\cdots,n.$$

对于每个体素 $\boldsymbol{V}$，计算其所包含点的平均坐标值作为质心 (v_x,v_y,v_z)，用点到质心的距离来增强每个点的特征, 同时将反射率 r_i 作为补充信息, 得到

$$\boldsymbol{V}_{\mathrm{in}}=\left\{\hat{\boldsymbol{p}}_i=\left[x_i,y_i,z_i,r_i,x_i-v_x,y_i-v_y,z_i-v_z\right]^{\mathrm{T}}\in\mathbb{R}^7\right\},$$

其中，$\boldsymbol{V}_{\mathrm{in}}$ 为体素特征，$i=1,\cdots,n$；然后，将 $\boldsymbol{V}_{\mathrm{in}}$ 通过 PointNet 转化到 m 维特征空间, 记为 $f_i\in\mathbb{R}^m$. 接着使用最大池化函数对 f_i 进行聚合，得到体素 V 的特征 $\tilde{\boldsymbol{f}}\in\mathbb{R}^m$.

由于空间中体素数量较多, 因此体素编码方式的设计会影响模型的推理效率. 为了提高体素编码效率, 仅使用质心坐标 (v_x,v_y,v_z) 作为每个体素的特征，即 $\hat{\boldsymbol{V}}_{\mathrm{in}}=\left\{\hat{\boldsymbol{p}}_i=\left[v_x,v_y,v_z\right]^{\mathrm{T}}\in\mathbb{R}^3\right\}$，相比传统 VFE 效率更高. 将 $\hat{\boldsymbol{V}}_{\mathrm{in}}$ 输入 PointNet 进行升维处理, 得到体素的特征

$$f_i=g\left(\frac{1}{t}\sum_{k=1}^{t}x_{i,k},\frac{1}{t}\sum_{k=1}^{t}y_{i,k},\frac{1}{t}\sum_{k=1}^{t}z_{i,k}\right).$$

其中，$g(\cdot)$ 为特征变换函数. 为减少模型的训练参数, 在升维过程中使用权值共享机制对点进行处理, 即共享多层感知机, 所有点共享同一个 MLP 的权重. 如图 2 所示, 将 $\hat{\boldsymbol{V}}_{\mathrm{in}}$ 送入共享多层感知机中，输出 f_i 为体素的特征，其中，$(v_{x,n},v_{y,n},v_{z,n})$ 为体素质心坐标，$(a_{n,1},a_{n,2},\cdots,a_{n,m})$ 为每个体素升维后的特征, 本文设置 $m=16$.

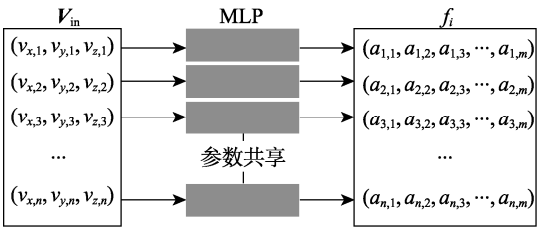


图 2 共享多层感知机示意图

2.4 MSS-FA 模块

在对 3D 体素格进行特征提取时需要考虑点云的稀疏问题. 对于点云空间, 其大部分位置不包含任何点, 在这些位置生成的体素为空体素, 如果直接使用 3D 卷积来对整个体素空间进行特征提取, 会使得大量计算浪费在空体素上, 极大地降低模型提取体素特征的计算效率, 并且会导致特征膨胀的现象出现. 在 KITTI 数据集上, 体素的稀疏度接近 0.005[10]. 因此, 本文使用 3D 稀疏卷积来替代传统 3D 卷积, 以提升模型计算效率.

本文使用子流形稀疏卷积[25]和稀疏卷积[11] 2 种 3D 稀疏卷积来提高卷积效率、降低显存消耗. 子流形卷积保持了特征的稀疏性, 其输出特征图与输入特征图的非空位置是相同的. 所以, 使用子流形卷积来对特征进行升维和通道变换, 可以保证特征图输入与输出的稀疏性是一致的. 稀疏卷积通过建立规则手册, 记录非空位置的特征, 使得卷积只在非空体素上进行计算, 提高了 3D 卷积效率. 基于这个特点, 本文使用稀疏卷积对 3D 体素进行下采样, 以获得 3D 体素在不同空间尺度上的语义特征. 基于 3D 稀疏卷积, 本文设计了 MSS-FA 模块, 通过聚合点云在不同尺度稀疏空间上的特征来获得更丰富的语义信息.

点云空间信息的损失是影响基于体素的 3D 目标检测模型提升准确率的一个重要因素，单一尺度的空间体素特征不足以表示点云场景中的所有信息. 因此，本文提出 MSS-FA 模块对点云在多个尺度空间中的特征进行编码和聚合，MSS-FA 模块结构如图 3 所示.

本文以自下而上的方式设计编码网络，每层特征的下采样倍数为 2，通过下采样逐渐降低体素特征分辨率. 每层空间体素特征表示点云在不同空间尺度上的语义特征. 为获得更好的特征表达，MSS-FA 模块将多个尺度的体素特征进行聚合. 具体来说，在 3D 稀疏卷积网络的每层增加一个横向连接，使用 3D 稀疏卷积对特征进行通道和尺度变换，最后将各个尺度的体素特征进行聚合，从而形成 MSS-FA 的层次结构.

图 3 中橘黄色为稀疏卷积层，每层参数含义为 $[C_{\text{in}}, C_{\text{out}}, k, s, p]$；其中，$C_{\text{in}}$ 为输入特征通道数量，C_{out} 为输出特征通道数量；k 为卷积核尺寸；s 为卷积的步长值；p 为特征图的宽度填充数量. 设每层特征为 $f^{\text{conv}i}$，其中，$i=1,2,3,4,5$. $f^{\text{conv1}} \sim f^{\text{conv4}}$ 每一层进行 2 倍下采样，得到点云在不同尺度稀疏空间上的特征，然后对 $f^{\text{conv1}} \sim f^{\text{conv5}}$ 进行特征聚合，得到聚合后的特征为

$$\tilde{\boldsymbol{f}} = g\left(\text{concat}\left(g\left(f^{\text{conv}i}\right)\right)\right).$$

其中，$g(\cdot)$ 表示特征变换函数；concat 表示特征拼接，用来对特征进行聚合，再由另一个特征变换函数对特征的通道数量进行调整. $\tilde{\boldsymbol{f}}$ 可以聚合不同层级的空间语义特征，能够更详细地描述点云空间结构，为候选框的生成提供更多空间信息.

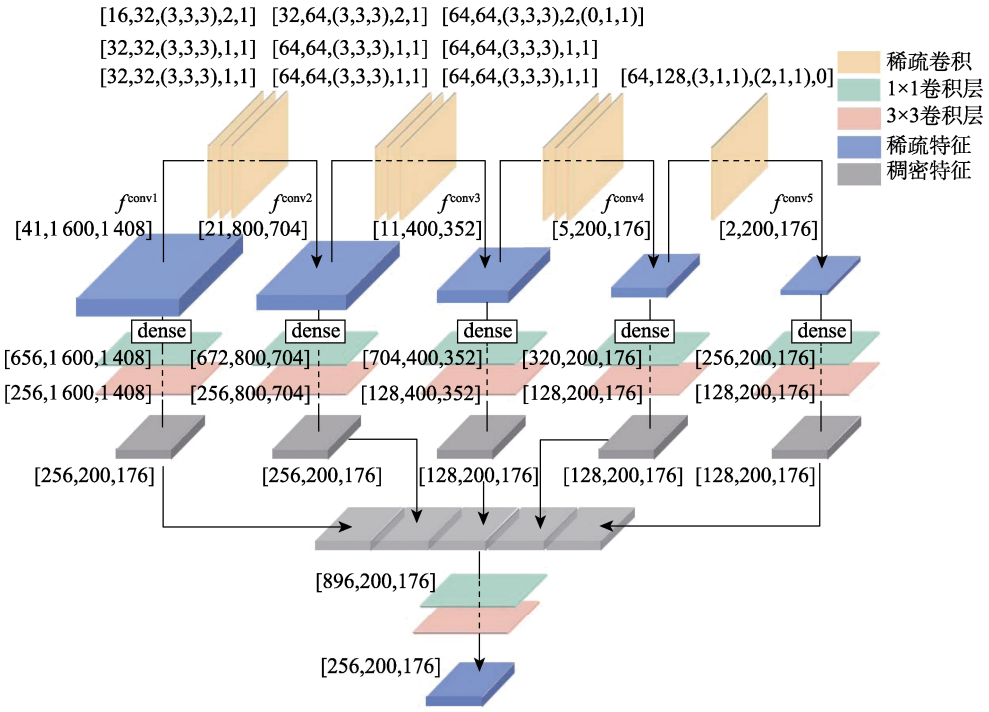


图 3　MSS-FA 模块结构示意图

为更有效地利用显存资源、降低模型计算量，并解决分辨率较高的底层特征图需要占用大量显存资源的问题，在进行特征变换时，MSS-FA 模块首先对各层体素特征进行通道变换，将 f^{conv1}，f^{conv2} 的通道数量调整为 256，$f^{\text{conv3}} \sim f^{\text{conv5}}$ 的通道数量调整为 128. 接着，将体素特征按照通道维度进行聚合，MSS-FA 模块堆叠多个卷积核大小为 3×3 的卷积层来统一特征图尺寸，每一层均包含卷积层、批量归一化层和激活函数，如图 4 所示. 将各尺度的特征分辨率调整到 200×176 像素，拼接后的特征图共包含 896 个通道. 然后使用卷积核尺寸为 1×1 的卷积层将通道数调整为 256. 最后，聚合了多尺度稀

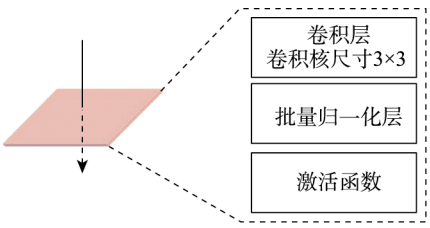


图 4　卷积层细节

疏空间特征图的尺寸为 256×200×176.

2.5 FHE 模块

为增强特征对目标的表达能力，对聚合后的特征通过 FHE 模块进行分层激励. 该模块可以有效地提高模型对困难样本的检测准确率.

点云对物体空间几何结构的描述与图像不同. 物体在图像中会表现出近大远小的特点，而在点云中的大小则不受距离的影响，在空间上具有尺度不变的特点. 所以，将基于图像的目标检测模型中广泛应用的下采样特征提取方法直接应用到点云中可能会进一步造成物体空间信息的损失. 但是，如果不对特征进行下采样，模型的计算量会变得非常大，而且在特征提取过程中由于卷积核的感受野较小，难以有效地提取点云局部特征. 为解决上述问题，本文提出适用于点云的 FHE 模块，将特征下采样到一定的尺寸后，通过扩大感受野来改善特征学习的效果，可以更充分地学习点云局部特征，其具体结构如图 5 所示.

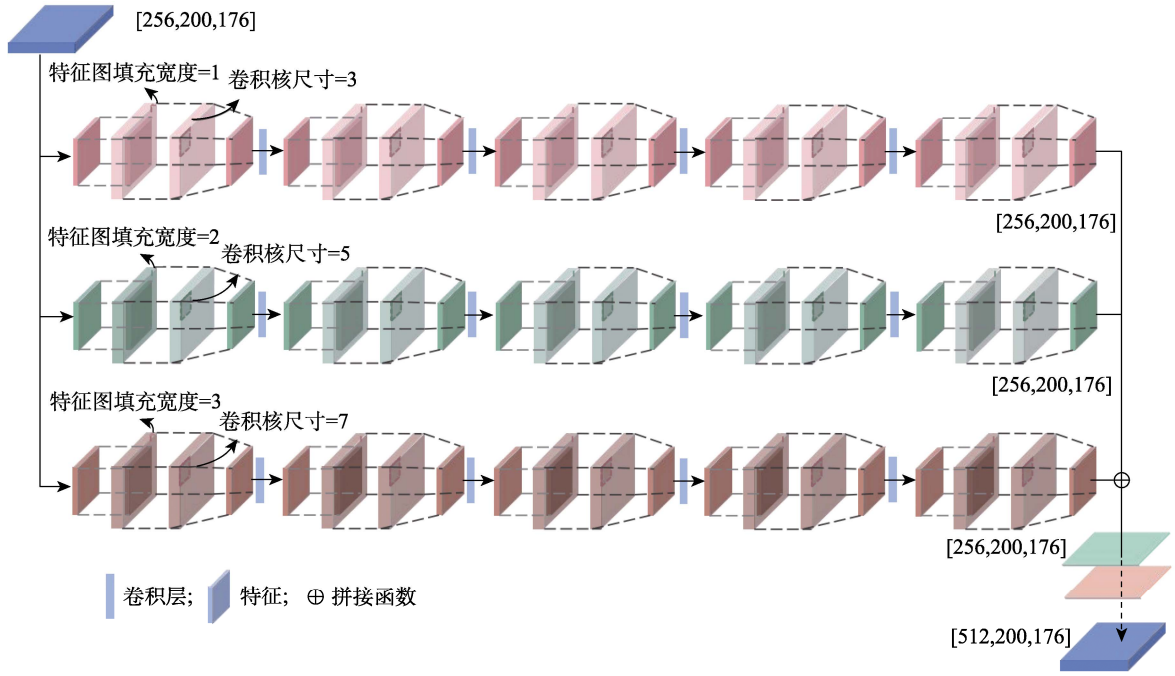


图 5 FHE 模块结构示意图

FHE 模块通过 3 层卷积通道对目标特征进行强化，每层通道堆叠多层卷积神经网络，卷积核尺寸分别为 3×3, 5×5 和 7×7. 原始点云空间在鸟瞰视角下尺寸为 80.0 m×70.4 m. 映射到分辨率为 200×176 像素的特征图后，3×3 大小的卷积核对应的原始点云鸟瞰视角尺寸为 1.2 m×1.2 m, 5×5 大小的卷积核对应 2.0 m×2.0 m, 7×7 大小的卷积核对应 2.8 m×2.8 m. 这些尺寸的卷积核基本能够覆盖汽车、行人和自行车这 3 种类别的目标. 为保证训练时特征图尺寸不变，本文选取的训练参数如表 1 所示.

通道内每一层特征图尺寸固定为 200×176，使用 5 层堆叠的卷积神经网络来进行学习，每层均包含卷积层、批量归一化层和激活函数. 最后将 3 层特征通过 MLP 进行整合，增加目标所包含的信息量，强化特征的表达能力，降低噪声信息对检测结果的影响，提高模型鲁棒性.

表 1 FHE 模块训练参数

卷积核尺寸	特征图填充宽度	卷积步长	特征图通道数量
3×3	1	1	256
5×5	2	1	256
7×7	3	1	256

2.6 损失函数

将点云特征分别输入置信度预测、锚框位置预测和朝向预测分支，用于计算锚框的分类损失、边框损失和朝向损失. 对于锚框位置，使用 (x,y,z,w,l,h,θ) 格式的向量进行编码. 其中，(x,y,z) 为框的中心点的坐标；w，l，h 分别为框的宽、长、高；θ 为锚框的朝向角度. 预测框和真实框之间的差值计算公式为

$$x_{\mathrm{t}}=\frac{x_{\mathrm{g}}-x_{\mathrm{a}}}{d_{\mathrm{a}}},\ y_{\mathrm{t}}=\frac{y_{\mathrm{g}}-y_{\mathrm{a}}}{d_{\mathrm{a}}},\ z_{\mathrm{t}}=\frac{z_{\mathrm{g}}-z_{\mathrm{a}}}{h_{\mathrm{a}}},$$

$$w_{\mathrm{t}}=\ln\left(\frac{w_{\mathrm{g}}}{w_{\mathrm{a}}}\right),\ l_{\mathrm{t}}=\ln\left(\frac{l_{\mathrm{g}}}{l_{\mathrm{a}}}\right),\ h_{\mathrm{t}}=\ln\left(\frac{h_{\mathrm{g}}}{h_{\mathrm{a}}}\right),$$

$$\theta = \theta_{\mathrm{g}} - \theta_{\mathrm{a}}.$$

其中，$d_{\mathrm{a}} = \sqrt{(w_{\mathrm{a}})^2 + (l_{\mathrm{a}})^2}$；$x_{\mathrm{t}}$，$y_{\mathrm{t}}$ 和 z_{t} 分别表示在 3 个方向上预测框和真实框中心点坐标的差值，w_{t}，l_{t} 和 h_{t} 分别代表预测框和真实框宽、长和高的差值，θ 代表预测框和真实框朝向角度的差值. 最后用 $\Delta \boldsymbol{g} = (x_t, y_t, z_t, w_t, l_t, h_t, \theta)$ 表示预测框和真实框之间的距离.

考虑正负样本不平衡问题，本文使用 Focal Loss[26]作为分类损失函数，为正负样本的损失分别设置不同的权重，以平衡每种损失对总分类损失的贡献度. Focal Loss 计算公式为

$$L_{\mathrm{cls}} = -\alpha_{\mathrm{t}} (1 - p_{\mathrm{t}})^{\gamma} \log(p_{\mathrm{t}}).$$

其中，α 和 γ 为 Focal Loss 的参数，在训练时通常使用 $\alpha = 0.25$，$\gamma = 2.0$；

$$p_{\mathrm{t}} = \begin{cases} \hat{p}, & \text{if } y = 1 \\ 1 - \hat{p}, & \text{otherwise} \end{cases}.$$

其中，$\hat{p}$ 由模型预测的分类分数经过 Sigmoid 函数计算得到.

本文模型在计算角度损失时，使用和 SECOND[11]一样的角度损失函数，计算公式为

$$L_{\theta} = \mathrm{Smooth}L_1\left(\sin\left(\theta_{\mathrm{g}} - \theta_{\mathrm{a}}\right)\right).$$

在角度损失计算时，由于当锚框的角度为 0 和 π 时会对应同一个锚框，但预测值和真实值相差 π 会导致损失过大，使用 sin 函数可以解决这一问题. 同时，为了表示锚框的朝向，定义朝向损失为 L_{dir}.

位置损失使用 SmoothL_1 损失函数，计算公式为

$$L_{\mathrm{loc}} = \mathrm{Smooth}L_1(\Delta p, \Delta g).$$

其中，Δp 为头部检测网络生成的锚框位置编码的预测值；Δg 为该预测框对应的真实框位置编码真实值.

本文模型的总损失计算公式为

$$L_{\mathrm{total}} = \beta_1 L_{\mathrm{cls}} + \beta_2 (L_{\theta} + L_{\mathrm{loc}}) + \beta_3 L_{\mathrm{dir}} \quad (1)$$

其中，β_1，β_2，β_3 分别为 3 种损失对应的权重.

3 实　验

本文使用 KITTI 数据集进行实验，该数据集由激光雷达点云和图像样本组成，包含 7 481 个训练样本和 7 518 个自动驾驶场景中的测试样本；其中，训练样本分为 3 712 个训练样本集和 3 769 个验证样本集，包含车辆、行人和自行车共 3 个类别. 本文给出了模型的实验结果，并对结果进行可视化分析，最后通过消融实验证明所提出模块的有效性.

3.1 实验环境设置

本文选取如下软硬件配置作为实验环境: CPU 为 Intel(R) Xeon(R) Gold 5218R(10 核); GPU 为 RTX 3090; 显存为 24 GB; 操作系统为 Ubuntu 20.04; Python 版本为 3.8; 深度学习框架为 PyTorch 1.8.1; CUDA 版本为 11.1; cuDNN 版本为 8.0; 运行环境为 Visual Studio Code 1.66.1.

3.2 实验细节

3.2.1 锚框尺寸

本文针对汽车、行人和自行车 3 种类别进行统一训练，其中，汽车锚框的尺寸在 x, y, z 方向分别设定为 3.90 m, 1.60 m, 1.56 m; 行人为 0.80 m, 0.60 m, 1.73 m; 自行车为 1.76 m, 0.60 m, 1.73 m. 对于汽车类别的正负样本划分，如果模型所预测的锚框与真实值间的 IoU>0.60, 则被划分为正样本; IoU<0.45 则被划分为负样本; IoU 在 0.45~0.60 的预测锚框会被丢弃. 对于行人和自行车，若模型所预测的锚框与真实值间的 IoU>0.50, 则被划分为正样本; IoU<0.35 则被划分为负样本; IoU 为 0.35~0.50 的预测锚框会被丢弃

3.2.2 体素化

为了数据对齐，点云在 x, y, z 方向的取值范围分别为 $[0.0, 70.4]$，$[-40.0, 40.0]$ 和 $[-3.0, 1.0]$；相应的体素块尺寸分别为 0.05 m, 0.05 m 和 0.10 m, 体素块数量分别为 1 408, 1 600 和 40, 设置每个体素内点的数量不超过 5 个.

3.2.3 数据增强

(1) 增加被检测目标. 将所有场景的真实值都存放到一个字典中，在训练过程中，随机从字典中选择真实值加入到当前点云中，增加了每帧点云的目标数量，模拟在不同环境中的物体. 在加入真实值时进行物理碰撞测试，移除与其他物体有碰撞的采样物体.

(2) 全局旋转和缩放. 对点云中所有的点在 $[0.95, 1.05]$ 范围内进行随机倍数的缩放，并在 $\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$ 范围内进行随机角度的旋转.

(3) 真值旋转和线性变换. 对所有真实框和真实框内的点进行服从高斯分布的随机线性变换，在 $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ 范围内进行随机角度的旋转，以模拟不同尺寸的真实值和转向.

(4) 翻转. 将点云沿 x 轴进行随机翻转.

3.2.4 特征提取网络

特征提取网络包括 MSS-FA 和 FHE 这 2 个模块.

MSS-FA 模块使用 3D 稀疏卷积来加速卷积运算. 模块的网络包含 5 层中间特征, 尺寸分别为 656×1 600×1 408, 672×800×704, 704×400×352, 320×200×176 和 256×200×176. 分别对 5 个中间层特征进行通道变换和下采样, 得到 5 个新的特征矩阵, 尺寸分别为 256×200×176, 256×200×176, 128×200×176, 128×200×176 和 128×200×176. 将上述 5 个特征矩阵拼接后得到 896×200×176 的特征矩阵. 最后再对其进行通道变换, 得到尺寸为 256×200×176 的矩阵.

FHE 模块进一步对空间特征进行强化. 按卷积核尺寸不同分为 3 个通道, 卷积核尺寸分别为 3×3, 5×5 和 7×7, 每个通道都包含 5 层卷积层. 为保证特征图尺寸不变, 3×3 通道的卷积层设定特征图填充宽度为 1, 卷积步长为 1; 5×5 通道的卷积层设定特征图填充宽度为 2, 卷积步长为 1; 7×7 通道的卷积层设定特征图填充宽度为 3, 卷积步长为 1. 3 个通道输出的特征矩阵尺寸均为 256×200×176, 将 3 个通道特征进行拼接, 得到尺寸为 768×200×176 的特征图, 然后对其进行通道变换, 映射为 512×200×176, 最后将其输入到检测头进行锚框的分类和回归.

3.2.5 多任务损失函数

本文模型的损失包括分类损失、位置损失和朝向损失, 3 种损失权重分别为 1.0, 2.0 和 0.2.

3.2.6 训练设置

本文所提模型使用 Adam_onecycle 优化器进行训练, 初始学习率设置为 0.01, 批次大小设置为 4, 训练次数设置为 80, 每次训练共有 928 个批次. 学习率随着训练次数的增加而改变, 分别在第 35 次训练和第 45 次训练时衰减 10%.

3.3 实验结果与分析

3.3.1 评价标准与实验设置

表 2 所示为本文模型与当前主流 3D 目标检测模型在 KITTI 数据集上对汽车、行人和自行车检测的准确率对比, 每个类别包含 3 种难度级别: 简单、中等和困难. 遵循 KITTI 标准的评价要求, 保持 IoU 阈值和平均准确率(mean average precision, mAP)计算方式和表中其他模型一致, 将汽车类别的 IoU 阈值设置为 0.7, 行人和自行车类别的 IoU 阈值设置为 0.5, mAP 使用第 40 个召回位置进行计算.

表 2 KITTI 数据集上 11 种模型对 3 类目标的检测准确率 %

模型			汽车			行人			自行车		
类型	模式	名称	简单	中等	困难	简单	中等	困难	简单	中等	困难
双阶段	RGB+LiDAR	F-PointNet[22]	82.19	69.79	60.59	50.53	42.15	38.08	72.27	56.12	49.01
		AVOD-FPN[19]	83.07	71.76	65.73	50.46	42.27	39.04	63.76	50.55	44.93
		F-ConvNet[27]	87.36	76.39	66.69	52.16	43.38	38.80	81.98	65.07	56.54
	LiDAR	Part-A2[28]	87.81	78.49	73.51	53.10	43.35	40.06	79.17	63.52	56.93
		PointRCNN[6]	86.96	75.64	70.70	47.98	39.37	36.01	74.96	58.82	52.53
		STD[7]	87.95	**79.71**	75.09	53.29	42.47	38.35	78.69	61.59	55.30
单阶段	LiDAR	SECOND[11]	83.13	73.66	66.20	51.07	42.56	37.29	70.51	53.85	46.90
		PointPillars[12]	82.58	74.31	68.99	51.45	41.92	38.89	77.10	58.65	51.92
		TANet[29]	84.39	75.94	68.82	53.72	44.34	40.49	75.70	59.44	52.53
		3DSSD[8]	**88.36**	79.57	74.55	**54.64**	44.27	40.23	**82.48**	64.10	56.90
		本文	88.17	77.91	**76.67**	53.27	**48.98**	**45.79**	76.02	**65.77**	**60.68**

注. 加粗表示当前类别最高检测准确率.

3.3.2 结果分析

表 2 按照单阶段和双阶段的分类, 列举了不同类型检测模型, 包括本文模型的实验结果. 本文模型采用同一个模型对 3 个类别进行训练, 未针对类别进行独立训练. 本文模型为单阶段模型, 因此重点对比与其他单阶段模型的检测性能. 可以看到, 本文模型在同类的单阶段模型中性能提升明显, 尤其是对困难难度目标的检测准确率明显优于其他模型. 针对简单、中等和困难 3 种不同难度级别的汽车目标, 本文模型检测准确率分别达到 88.17%, 77.91%和 76.67%; 在行人类别中, 分别达到 53.27%, 48.98%和 45.79%; 在自行车类别中, 分别达到 76.02%, 65.77%和 60.68%. 尤其在困难级别样本检测中, 本文模型相对主流单阶

段模型具有优势, 证明本文提出的 MSS-FA 和 FHE 能够有效获取更准确的点云特征, 提高 3D 目标检测准确率.

在鸟瞰视角下, 本文模型在检测准确率上同样具有优势. 本文比较了多个主流单阶段 3D 目标检测模型在 KITTI 数据集中对汽车、行人和自行车鸟瞰视角下的平均检测准确率, 检测结果如图 6 所示. 在与 7 种模型的比较中, 本文模型对汽车检测的平均检测准确率排名靠前, 对行人和自行车类别的平均检测准确率均排名第一.

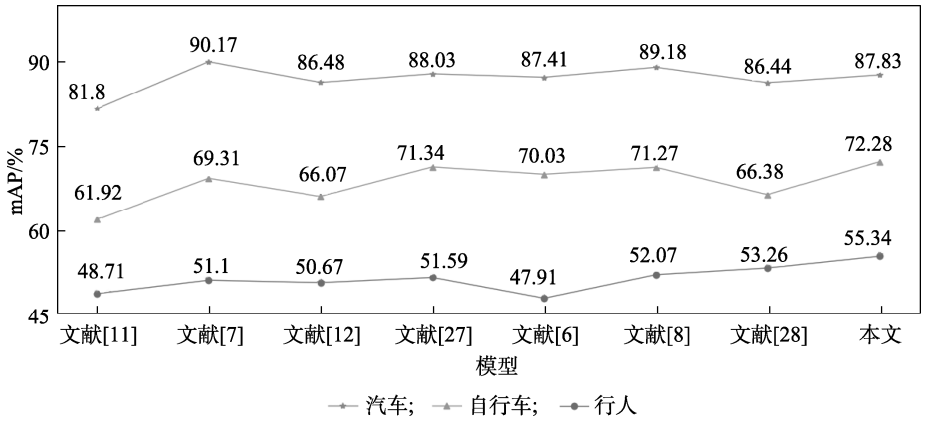


图 6　鸟瞰视角下各种模型平均检测准确率

模型训练过程的训练损失曲线如图 7 所示, 在训练起始阶段, 模型计算得到的损失较大, 然后迅速降低; 随着训练迭代次数增加, 损失平缓降低. 曲线表明本文模型在训练过程中损失会逐渐降低, 从整体趋势上说明模型训练有效, 未发生异常情况.

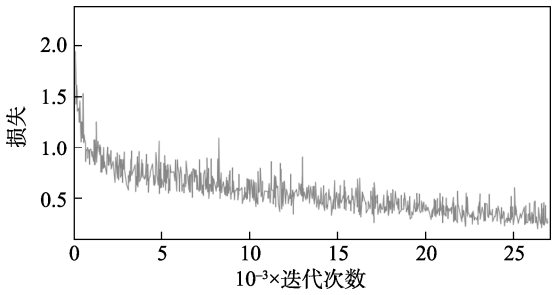


图 7　训练损失曲线

3.3.3　可视化分析

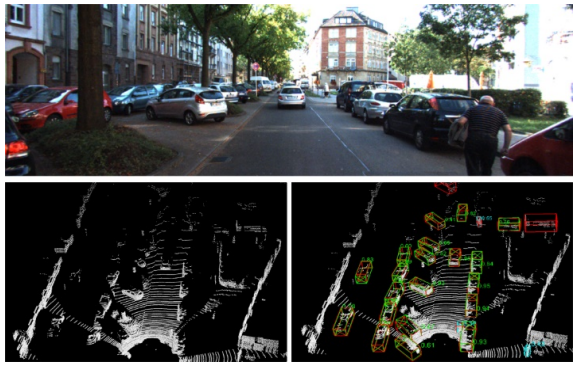
对模型的检测效果进行可视化分析, 如图 8 所示. 汽车、行人和自行车分别对应绿色、蓝色和黄色框, 红色为真实框. 根据可视化结果可以看出, 模型对不同点云场景检测均较为准确, 即使对远处点较为稀疏物体的检测同样有效, 对被遮挡、出现重叠的物体检测效果良好. 以图 8a 为例, 左侧的汽车虽然被遮挡, 点较为稀疏, 但仍然被模型准确检测到; 对自行车类别的检测同样有效, 以图 8d 为例, 模型准确检测出左侧出现的自行车. 本文对真实框进行了可视化, 以分析模型生成的预测框的质量. 从图 8 中可以看到, 模型生成的预测框与真实框接近, 重合度较高, 说明预测框质量较好. 虽然模型整体表现良好, 但仍然有一些漏检情况, 例如, 图 8a 右侧较远处只出现红色的真实框, 没有出现预测框, 说明该处的物体被漏检. 分析认为是由于这些远处的物体包含的点非常稀疏, 同时被其他物体遮挡, 减少了物体所包含的点, 使模型难以获得足够多的信息进行检测.

3.3.4　对比实验

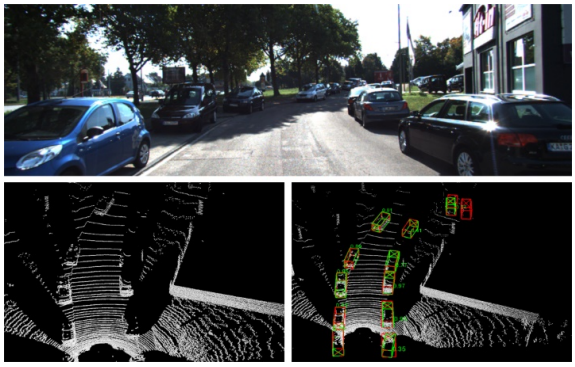
为进一步验证本文模型的实际检测效果, 与经典的单阶段模型 SECOND[11]和 PointPillars[12]在同一场景下进行可视化对比, 结果如图 9 所示, 其中, 红色框为真实框, 绿色框为预测框. 可以看出, 相比本文模型, SECOND[11]和 PointPillars[12]均出现不同程度的漏检和误检, 两者均未检测到近处路右边的白色汽车. SECOND[11]将右侧建筑的点云误检为行人; PointPillars[12]将右侧建筑的点云识别为汽车, 未检测到墙角处的自行车, 在没有物体的地方检测出汽车; 本文模型检测结果则较为准确, 较为鲁棒, 正确识别出近处的白色汽车和墙角的自行车, 且未对右侧建筑产生误检. 需要说明的是, 本文模型对距离激光雷达较远的被严重遮挡的目标存在一定误检, 由于模型同时针对 3 种类别目标进行训练, 而不同类别之间目标的点云在被严重遮挡或截断时存在一定相似性, 导致模型将汽车的部分点云误检为其他类别.

3.4　消融实验

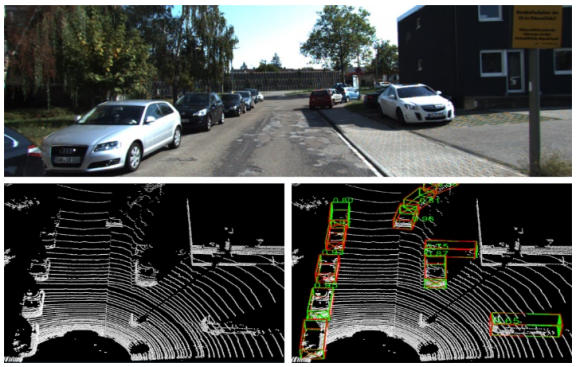
为进一步验证本文提出的 MSS-FA 和 FHE 模



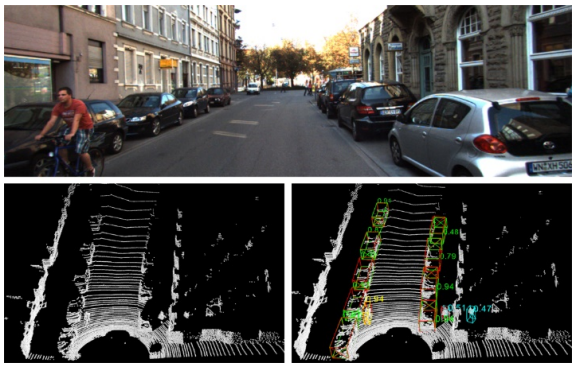
a. 场景 1



b. 场景 2



c. 场景 3

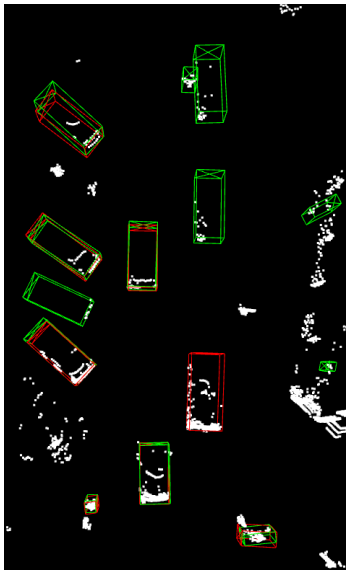


d. 场景 4

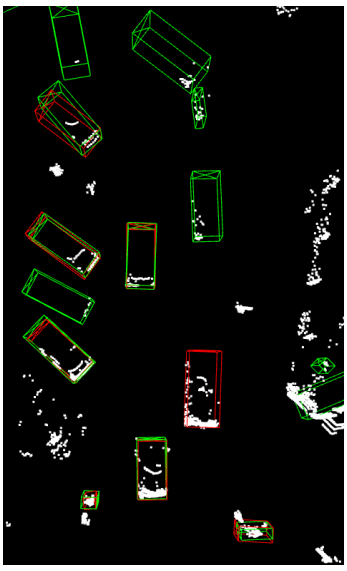
图 8　模型在 KITTI 数据集上检测结果的可视化



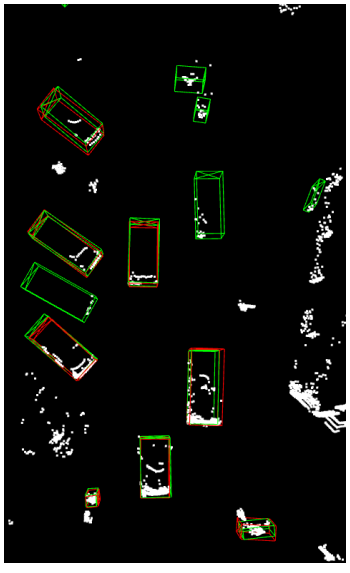
a. 场景正视图像视角



b. SECOND[11]



c. PointPillars[12]



d. 本文

图 9　3 种模型检测效果对比

块的效果，对 MSS-FA 和 FHE 模块进行消融实验，使用 SECOND[11]模型作为基线，将各模块分别加入 SECOND[11]模型中，验证所提出的模块的实际效果，结果如表 3 所示．可以看出，MSS-FA 和 FHE 模块对模型性能提升均是有效的．在 SECOND[11]模型中加入 MSS-FA 模块后，在汽车类别简单、中等和困难样本的检测中，准确率分别提升约 4.46 个百分点、3.26 个百分点和 8.73 个百分点；在行人检测中，简单样本检测准确率降低约 1.20 个百分点，中等和困难样本检测准确率分别提升约 1.11 个百分点和 3.20 个百分点；在自行车简单、中等和困难样本的检测中，准确率分别提升约 8.13 个百分点、11.59 个百分点和 14.64 个百分点．在 SECOND[11]模型中加入 FHE 模块后，在汽车类别简单、中等和困难样本的检测中，准确率分别提升约 4.96 个百分点、4.31 个百分点和 10.24 个百分点；在行人检测中，简单样本检测准确率降低约 0.46 个百分点，中等和困难样本检测准确率分别提升约 2.78 个百分点和 4.21 个百分点；在自行车简单、中等和困难样本的检测中，准确率分别提升约 10.35 个百分点、12.57 个百分点和 15.11 个百分点．值得一提的是，只加入 FHE 模块对自行车类别检测性能提升较多，而加上 MSS-FA 模块后，对自行车类别检测性能有所下降，但对行人类别检测性能得到提升．分析认为是由于汽车、行人和自行车的尺寸不同，一些模型在训练时会使用不同的超参数分别对 3 种类别进行处理，而本文模型则使用统一的参数训练，训练效果追求性能整体上的提高．由结果可知，MSS-FA 和 FHE 模块对困难目标检测性能的提升是有效的，将 MSS-FA 和 FHE 模块均加入到 SECOND[11]模型后，3 个类别不同难度等级的检测效果均得到提升，实验验证了各模块的有效性．

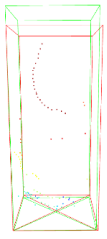
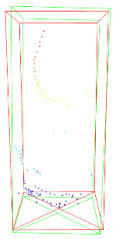
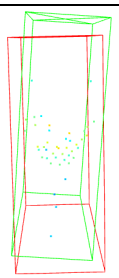
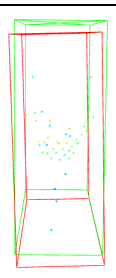
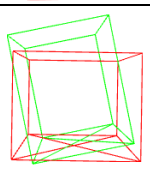
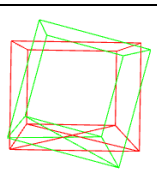
表 3　消融实验检测准确率　　%

模块	汽车			行人			自行车		
	简单	中等	困难	简单	中等	困难	简单	中等	困难
SECOND[11]	83.13	73.66	66.20	51.07	42.56	37.29	70.51	53.85	46.90
+ MSS-FA	87.59	76.92	74.93	49.87	43.67	40.49	78.64	65.44	61.53
+ FHE	88.09	**77.97**	76.44	50.61	45.34	41.50	**80.86**	**66.42**	**62.01**
+ MSS-FA + FHE	**88.17**	77.91	**76.67**	**53.27**	**48.98**	**45.79**	76.02	65.77	60.68

注．加粗表示当前类别最高检测准确率．

对 FHE 模块的有效性进行可视化分析，以 SECOND 模型为基线模型，红色框为真实框，绿色框为预测框，结果如表 4 所示．可以看出，为未使用 FHE 模块时，汽车、自行车与行人的点云预测框与真实框的位置具有明显偏差．使用 FHE 模块后对应的预测框与真实框的位置重合度较高，说明预测框定位更为地准确．实验表明，FHE 模块提取到的特征可以有效提高候选框的定位能力．

表 4　FHE 模块可视化实验

类别	点云预测框与真实框位置	
	未使用 FHE 模块	增加 FHE 模块
汽车		
自行车		
行人		

4　结　语

本文提出了一种可端对端训练的基于体素的单阶段 3D 目标检测模型，该模型聚合多尺度的稀疏空间特征来增强特征对空间的描述，并在此基础上通过分层激励对特征进行充分学习，提升特征对目标的表达能力．相比其他主流单阶段模型，本文模型在 KITTI 数据集上汽车、行人和自行车 3 个类别的检测准确率上具有优势，尤其对困难类别样本检测效果提升明显．实验结果表明，对点云空间信息的充分聚合和学习可以显著提高模型检

测准确率. 虽然多尺度感受野对提高模型性能十分关键, 但需要使用较大的卷积核尺寸并带来计算代价的增加, 影响模型的推理速度. 接下来, 将继续研究聚合多尺度信息的锚框精细化调整, 在扩大感受野范围的同时, 提高特征提取效率, 以进一步提升基于体素的单阶段 3D 目标检测模型性能.

参考文献(References):

[1] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91-110

[2] Su H, Maji S, Kalogerakis E, *et al*. Multi-view convolutional neural networks for 3D shape recognition. //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Boston: IEEE Computer Society Press, 2015: 945- 953

[3] Qi C R, Su H, Mo K C, *et al*. PointNet: deep learning on point sets for 3D classification and segmentation[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2017: 77-85

[4] Qi C R, Yi L, Su H, *et al*. PointNet++: deep hierarchical feature learning on point sets in a metric space[C] //Proceedings of the 31st International Conference on Neural Information Processing Systems. Cambridge: MIT Press, 2017: 5105-5114

[5] Qi C R, Litany O, He K M, *et al*. Deep Hough voting for 3D object detection in point clouds[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2019: 9276-9285

[6] Shi S S, Wang X G, Li H S. PointRCNN: 3D object proposal generation and detection from point cloud[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 770-779

[7] Yang Z T, Sun Y N, Liu S, *et al*. STD: sparse-to-dense 3D object detector for point cloud[C] //Proceedings of the IEEE/CVF International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2019: 1951-1960

[8] Yang Z T, Sun Y N, Liu S, *et al*. 3DSSD: point-based 3D single stage object detector[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2020: 11037-11045

[9] 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] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2022: 18931-18940

[10] Zhou Y, Tuzel O. VoxelNet: End-to-end learning for point cloud based 3D object detection[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 4490-4499

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

[12] Lang A H, Vora S, Caesar H, *et al*. PointPillars: fast encoders for object detection from point clouds[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2019: 12689-12697

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

[14] Li Qiao, Li Yaochen, Zhang Yulong, *et al*. A single-stage point cloud 3D object detection method using sparse 3D convolution[J]. Journal of Xi'an Jiaotong University, 2022, 56(9): 112-122(in Chinese)
(李悄, 李垚辰, 张玉龙, 等. 采用稀疏 3D 卷积的单阶段点云三维目标检测方法[J]. 西安交通大学学报, 2022, 56(9): 112-122)

[15] Shi S S, Guo C X, Jiang L, *et al*. PV-RCNN: point-voxel feature set abstraction for 3D object detection[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2020: 10526-10535

[16] Shi S S, Jiang L, Deng J J, *et al*. PV-RCNN++: point-voxel feature set abstraction with local vector representation for 3D object detection[OL]. [2022-07-25]. https://arxiv.org/abs/2102.00463

[17] Noh J, Lee S, Ham B. HVPR: hybrid voxel-point representation for single-stage 3D object detection[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2021: 14600-14609

[18] Chen X Z, Ma H M, Wan J, *et al*. Multi-view 3D object detection network for autonomous driving[C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2017: 6526-6534

[19] Ku J, Mozifian M, Lee J, *et al*. Joint 3D proposal generation and object detection from view aggregation[C] //Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. Los Alamitos: IEEE Computer Society Press, 2018: 1-8

[20] Lin T Y, Dollár P, Girshick R, *et al*. Feature pyramid networks for object detection[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2017: 936-944

[21] Yang B, Luo W J, Urtasun R. PIXOR: real-time 3D object detection from point clouds[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 7652-7660

[22] Qi C R, Liu W, Wu C X, *et al*. Frustum PointNets for 3D object detection from RGB-D data[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 918-927

[23] Vora S, Lang A H, Helou B, *et al*. PointPainting: sequential fusion for 3D object detection[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2020: 4603-4611

[24] Geiger A, Lenz P , Urtasun R. Are we ready for autonomous

driving? the KITTI vision benchmark suite[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Rhode Island: IEEE Computer Society Press, 2012: 3354-3361

[25] Graham B, Engelcke M, Van Der Maaten L. 3D semantic segmentation with submanifold sparse convolutional networks[C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Los Alamitos: IEEE Computer Society Press, 2018: 9224-9232

[26] Lin T Y, Goyal P, Girshick R, *et al*. Focal loss for dense object detection[C] //Proceedings of the IEEE International Conference on Computer Vision. Los Alamitos: IEEE Computer Society Press, 2017: 2999-3007

[27] Wang Z X, Jia K. Frustum ConvNet: sliding frustums to aggregate local point-wise features for Amodal 3D object detection[C] //Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems. Los Alamitos: IEEE Computer Society Press, 2019: 1742-1749

[28] Shi S S, Wang Z, Shi J P, *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, 2021, 43(8): 2647-2664

[29] Liu Z, Zhao X, Huang T T, *et al*. TANet: robust 3D object detection from point clouds with triple attention[C] //Proceedings of the AAAI Conference on Artificial Intelligence. Palo Alto: AAAI Press, 2020: 11677-11684