

DOI: 10.11830/ISSN.1000-5013.202204030



动态卷积的3D点云目标检测算法

蔡黎明^{1,2,3}, 杜吉祥^{1,2,3}, 刘怀进^{1,2,3}, 张洪博^{1,2,3}, 黄敬东^{1,2,3}

1. 华侨大学 计算机科学与技术学院, 福建 厦门 361021;
2. 华侨大学 福建省大数据智能与安全重点实验室, 福建 厦门 361021;
3. 华侨大学 厦门市计算机视觉与模式识别重点实验室, 福建 厦门 361021

摘要: 针对不规则且稀疏的点的提取特征问题, 提出一种以动态卷积作为特征提取的3D点云目标检测算法。首先, 以一种新型的动态卷积的方式自适应学习点的位置特征, 分类出前景点与背景点, 同时对提取出的前景点逐一做回归框; 然后, 用非极大值抑制选出分数值最好的回归框。其次, 进行粒度的细化, 得到修正规范的3D回归框, 完成3D物体的目标检测。最后, 在KITTI数据集上验证算法的有效性。结果表明: 文中所提算法在汽车类、行人、自行车类数据集上的3D点云目标检测精度更高。

关键词: 点云; 3D目标检测; 动态卷积; 分类回归

中图分类号: TP 391.41 **文献标志码:** A **文章编号:** 1000-5013(2023)01-0111-08

3D Point Cloud Target Detection Algorithm Based on Dynamic Convolution

CAI Liming^{1,2,3}, DU Jixiang^{1,2,3}, LIU Huaijin^{1,2,3},
ZHANG Hongbo^{1,2,3}, HUANG Jingdong^{1,2,3}

1. College of Computer Science and Technology, Huaqiao University, Xiamen 361021, China;
2. Fujian Key Laboratory of Big Data Intelligence and Security, Huaqiao University, Xiamen 361021, China;
3. Xiamen Key Laboratory of Computer Vision and Pattern Recognition, Huaqiao University, Xiamen 361021, China

Abstract: Aiming at the problem of extracting features from irregular and sparse points, a 3D point cloud target detection algorithm using dynamic convolution as feature extraction is proposed. Firstly, a new dynamic convolution method is used to adaptively learn the position features of points, and classify the foreground points and background points. At the same time, the extracted foreground points are used as regression boxes one by one. Then, non maximum suppression is used to select the regression box with the best score, secondly, the granularity is refined to obtain a 3D regression box of the revised standard, and the target detection of 3D objects is completed. Finally, the validity of the algorithm is verified on KITTI data set. The results show that the proposed algorithm has higher detection precision of 3D point cloud target on the car, pedestrian and bicycle data sets.

Keywords: point cloud; 3D target detection; dynamic convolution; classification regression

在汽车、飞机等交通工具中, 以计算机智能为核心的无人操作, 称为自动驾驶。为了提高驾驶安全、

收稿日期: 2022-04-26

通信作者: 杜吉祥(1977-), 男, 教授, 博士, 博士生导师, 主要从事模式识别及图像处理的研究。E-mail: jxdu@hqu.edu.cn.

基金项目: 国家自然科学基金资助项目(61673186); 国家重点研发计划专项项目子课题(2019YFC1604705); 福建省厦门市科技计划项目(3502220193037)

缓解交通压力、适应不同人群等问题,自动驾驶已经成为人工智能领域的热门话题.在自动检测场景中出现的目标是自动驾驶研究的关键.在过去研究中,目标检测研究主要侧重于二维图像中的目标检测.然而,当目标检测图片出现严重的遮挡与噪声时,常导致检测算法不精确.在自动驾驶中,常用的传感器是激光雷达传感器,与单目相机相比,激光雷达传感器可以提供更准确的深度信息,且能生成三维(3D)点云来捕捉场景的3D结构.准确的空间位置信息使自动驾驶汽车能够有效地预测和规划路径,避免发生不必要的交通事故.因此,基于3D点云的目标检测研究在自动驾驶领域备受关注.

为了处理非结构化的三维点云,体素的检测器通常将不规则的点转换成规则的体素网格,采用3D卷积神经网络进行特征学习.Zhou等^[1]提出一种新的体素特征编码层,提高了体素网格的有效并行处理能力.Yan等^[2]利用3D子流形稀疏卷积^[3],提出一种高效架构,使计算和内存效率显著提高.Lang等^[4]提出将体素简化为柱子.Shi等^[5]提出一种两阶段检测器,由部件感知模块和聚合模块组成.Deng等^[6]在两阶段的算法中引入全卷积网络来扩展一种新的高性能3D目标检测(PV-RCNN)算法,在体素特征中提取感兴趣区域特征,使粗体素粒度的检测精度得到提高.Ma等^[7]将点云分配到规则的三维网格中,并使用三维稀疏卷积提取体素特征.Xu等^[8]将原始的3D点云数据进行体素化,同时设计一个轻量级的位置注意模块,采用加权特征融合的方法,高效地获得多层次特征.

与体素的方法不同,基于点的算法是直接从非结构化的点中学习特征生成感兴趣区域.Qi等^[9-10]开创性地采用直接从点中学习特征的算法,保留了较多的特征信息,提升了模型的分类分割精度.Shi等^[11]提出一种从原始点云中检测3D物体的两阶段3D目标检测(PointRCNN)算法,该算法可以直接从原始点云中分割前景点,为预测3D回归框减少搜索空间.Yang等^[12]在下采样过程中提出一种融合采样策略,对点进行检测,设计了包含候选生成层、无锚回归头和3D中心性分配策略的预测网络.Shi等^[13]将图神经网络引入三维目标检测的框架中.

上述的3D目标检测算法在特征提取时采用的卷积均为静态.针对此问题,Xu等^[14]提出一种用于点云处理的位置自适应卷积算子.在点云三维立体空间中,点与点之间的关系复杂,可以使卷积核的选取适应周围的变化.点云三维目标检测的难点主要是点云的不规则性,若样本遮挡严重且较为稀疏,则难以提取关键特征,经常要转为有序的表达方式.基于此,本文提出一种新的3D目标检测算法,尝试解决不规则且稀疏点云数据中的目标检测问题.

1 实验方法

1.1 算法的整体框架

提出一种结合动态卷积学习特征的PointRCNN(PAC-PointRCNN)算法,其框架如图1所示.整体架构模型分为2个阶段:第1阶段主要通过构建动态卷积进行特征提取,并构建3D检测框;第2阶段对第1阶段的输出进行精细化处理.在第1阶段首先对点进行预处理,将得到的点数据送入骨干网络提取特征,每个点产生特征向量,同时,对周围的上下文信息进行编码,增大其感受野,通过KITTI数据集标

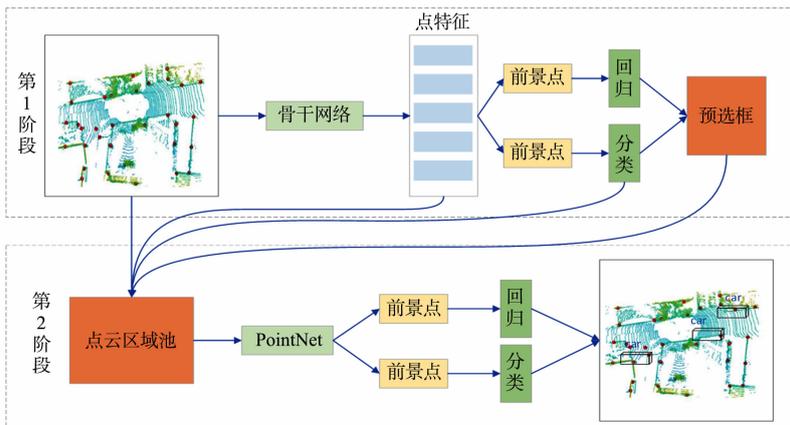


图1 PAC-PointRCNN 算法框架图

Fig. 1 PAC-PointRCNN algorithm framework diagram

注的 3D 边界框可以判断出前景点信息;然后,根据特征向量进行前景点分割,对得到的前景点进行分类回归,在对前景点预测所属对象的包围框时,用 bin 机制^[11]把回归问题先转化为分类问题,再进行回归,以提高效率及精度;得到大量预选框后,将预选框进行非极大值抑制处理,送入第 2 阶段进行特征融合,提取局部特征,得到的点位置坐标通过规范化处理,再分类回归得到精细框。

1.2 骨干网络

骨干网络,如图 2 所示。骨干网络由 4 个下采样编码的集合抽象层和 4 个上采样编码的特征传播层组成。

集合抽象层,如图 3 所示。图 3 中: N, N_1 为点集中点的数量; d 为坐标空间维度; C, C_1 为特征空间维度; k 为固定值。在集合抽象层中,采用最远点采样的方法对输入点进行降采样,将输入点集从 N 降到更小的 N_1 ,最远点采样是使采样的各个点之间尽可能远,这种降采样的结果比较均匀且具有更好的覆盖效果。在骨干网络的第 1 个集合抽象层中,对输入的原始点进行最远点采样,得到的降采样结果作为第 2 个集合抽象层的输入点,以此类推,直到第 4 个集合抽象层。将降采样得到点的数量 N_1 进行分组,选取这 N_1 个中心点,每个中心点采用球查询的方式寻找固定的 k 个邻域点,先划定某一半径,找在该半径球内的点作为邻域点,如果点数大于固定值 k ,取前 k 个点为邻域点,如果点数小于固定值 k ,对点进行重采样以达到固定值 k 。这样形成一个局部邻域,共组成 N_1 个局部邻域,这个过程中,坐标空间维度 d 和特征空间维度 C 没有发生变化。球查询在局部邻域组内固定了区域尺度,使局部区域特征在空间上更具有泛化性。

动态网络,如图 4 所示。在动态网络中,对每一组局部邻域进行动态网络编码,将采样和分组后的点作为输入,送进具有动态内核组装的位置自适应卷积即 PACConv 动态卷积里,使每个点的特征维度升高,以获取更多的特征信息,再通过最大池化选取最重要的特征作为新特征输出。在此过程中,坐标空间维度 d 没有发生变化,特征空间维度从 C 变成更高的维度 C_1 ,即特征的通道数升高了。在特征传播层中,采用基于距离的插值和跨越跳级链路的分层传播策略,与集合抽象层的反过程类似,将集合抽象层每一层得到的结果通过插值点特征传播到原始点集。PointNet 为直接从点云中学习特征的网络结构;PointNet++ 在 PointNet 的基础上进行改进,具有多层次特征提取结构。在 PointNet++ 的集合抽象层基础上,骨干网络将 PointNet 层嵌入 PACConv 动态卷积,生成动态网络,没有改变其他架构配置。

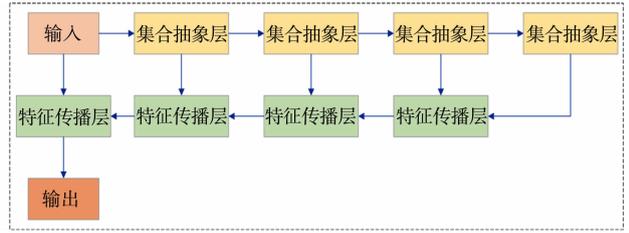


图 2 骨干网络

Fig. 2 Backbone network

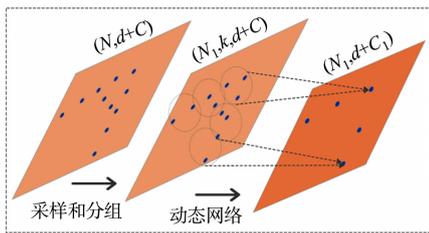


图 3 集合抽象层

Fig. 3 Set abstraction layer

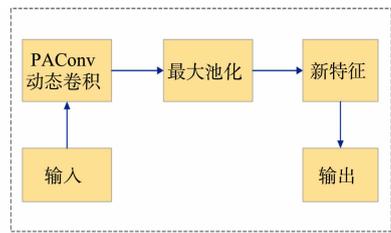


图 4 动态网络

Fig. 4 Dynamic network

1.3 动态卷积

在三维空间中,点与点之间的关系与二维平面中点之间的关系有很大的不同。例如,在二维平面的网格中,中心点周围的邻域点个数是固定的,但在点云的三维空间里,取一个中心点(图 5),它周围的邻近点有无限可能,多一个维度会使点关系变的错综复杂。在二维平面空间中,使用卷积神经网络学习得到的特征可以很好地反映点之间的相关性;而在三维空间里,由于点的无序不规则性,使用二维空间里学习特征的卷积核子会使相关性变差,从而导致检测不精确。因此,重新设计卷积核函数,使其能动态地学习点特征。PACConv 动态卷积,如图 6 所示。首先,定义由几个权重矩阵组成的权重库;接着,设计一个 LearnNet 学习系数向量,根据点的位置组合权重矩阵;最后,结合权值矩阵及其相关的位置自适应系数生成动态核。

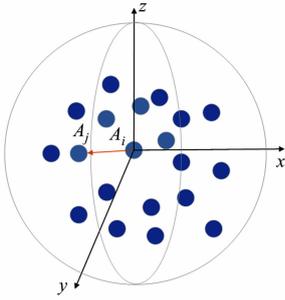


图5 3D点

Fig. 5 3D point

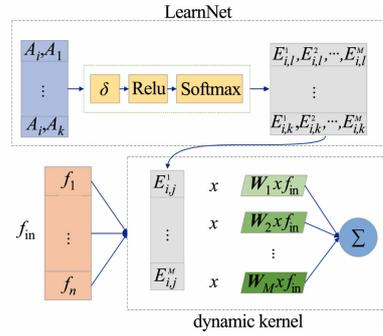


图6 PAConv 动态卷积

Fig. 6 PAConv dynamic convolution

定义权重库 $\mathbf{W} = \{\mathbf{W}_m \mid m=1, \dots, M\}$, 其中, $\mathbf{W}_m \in \mathbf{R}^{C_{in} \times C_{out}}$ 是一个权重矩阵, C_{in} 为输入通道, C_{out} 为输出通道; m 为控制存储在权重库 \mathbf{W} 中权重矩阵的个数, m 值越大, 内核组装的权重矩阵更加多样, 但过多的权重矩阵可能会带来冗余并导致沉重的内存和计算开销, 有实验表明^[14], 当 m 为 8 时可以取得最优的性能. 建立权重矩阵的离散核到连续三维空间的映射, 结合权值矩阵, 生成拟合点输入的动态核, LearnNet 将相对位置与权重库 \mathbf{W} 中的不同权重矩阵联系起来. 对于给定中心点 $A_i(x_i, y_i, z_i)$ 与相邻点 $A_j(x_j, y_j, z_j)$ 之间的特定位置关系, LearnNet 预测每个权值矩阵 \mathbf{W}_m 的位置自适应系数 $E_{i,j}^m$, 其输入基于位置关系, 点位置关系 P 的表达式为

$$P = (A_j - A_i, A_i), \quad (1)$$

$$(A_j - A_i, A_i) = (x_j - x_i, y_j - y_i, z_j - z_i, x_i, y_i, z_i). \quad (2)$$

LearnNet 的输出向量 $\mathbf{E}_{i,j}$ 表达式为

$$\mathbf{E}_{i,j} = \text{softmax}(\text{relu}(\delta(P))). \quad (3)$$

式(3)中: δ 为使用多层感知机实现的非线性函数; relu 为激活函数; softmax 为归一化函数; $\mathbf{E}_{i,j} = \{E_{i,j}^m \mid m=1, \dots, M\}$, softmax 输出的分数在 $(0, 1)$ 范围内, 归一化保证每个权重矩阵都有概率被选择, 分数越高, 意味着位置输入和权重矩阵之间的关系越强, 从而 PAConv 的动态核 K 是将 \mathbf{W} 中的权重矩阵与输入特征 f_{in} 结合, 再与 LearnNet 预测的相应系数 $\mathbf{E}_{i,j}$ 相乘, 得到动态核 K , 即

$$K(P) = \sum_{m=1}^M E_{i,j}^m \mathbf{W}_m f_{in}. \quad (4)$$

得到动态核后, 生成的自适应动态卷积可以更加灵活地学习特征. 因此, 权重库的规模越大, 权重矩阵的灵活性和可用性也越大, 但由于权重矩阵是随机初始化的, 在网络的训练过程中, 卷积核学习得到的合理权值会趋于相似, 不能保证权重矩阵的多样性. 为了避免这种情况, 采用权值正则化来惩罚不同权值矩阵之间的相关性, 让权重矩阵的分布更加多样化, 进一步保证生成核的多样性. 因此, 将 PAConv 动态卷积嵌入骨干网络, 从而可以更加灵活精确地学习三维空间的点特征.

1.4 损失函数

为了便于比较, 采用 PointRCNN 的损失函数, 由于大型户外场景前景点的数量比背景点的数量少, 因此, 在第 1 阶段的子网络中, 使用焦点损失^[15] 解决前景-背景分类不均衡的问题, 得到的损失为 L_{focal} , 即

$$L_{focal}(p_t) = -\varphi_t(1-p_t)^\gamma \ln p_t. \quad (5)$$

式(5)中: t 为前景点即正确分类点; p_t 为正确分类的概率; φ_t, γ 为超参数调节因子, $\varphi_t = 0.25, \gamma = 2$.

使用回归损失函数^[11] 生成 3D 预选框, 即

$$L_{reg} = \frac{1}{|G|} \sum_{g \in G} (L_{bin}^{(g)} + L_{res}^{(g)}). \quad (6)$$

式(6)中: L_{reg} 为生成 3D 预选框的损失; G 为前景点的集合; $|G|$ 为集合中的点个数; $L_{bin}^{(g)}$ 为运用 bin 机制的损失; $L_{res}^{(g)}$ 为训练中具有不同损失项的整体三维边界框的回归损失.

在第 2 阶段的子网络中, 将预选框坐标转到规范的坐标系下, 总的损失 L_{refine} 为

$$L_{refine} = \frac{1}{|T|} \sum_{r \in T} (F_{cls}(\text{pro}(b_r), \text{label}_r) + \frac{1}{|D|} \sum_{d \in D} (\tilde{L}_{bin}^{(d)} + \tilde{L}_{res}^{(d)})). \quad (7)$$

式(7)中: T 为预选框的集合; $\text{pro}(b_r)$ 为预选框 b_r 的置信度; label_r 为对应的标签; F_{cls} 为用于监督预测置信度的交叉熵损失; $\tilde{L}_{\text{bin}}^{(d)}$ 为第 2 阶段规范坐标下, 运用 bin 机制的损失; $\tilde{L}_{\text{res}}^{(d)}$ 为第 2 阶段规范坐标下, 更细化的位置回归损失; D 为交并比大于阈值的预选框集合.

2 实验结果与分析

2.1 数据集

KITTI 的三维检测数据集包含 7 481 个训练样本和 7 518 个测试样本, 与 PointRCNN 算法一样, 将训练样本分为训练集(3 712 个样本)和验证集(3 769 个样本). 标注文件里的字段含义分别为类别、截断率、遮挡程度、观测角度、2D 边界框左右下角坐标、3D 边界框的长宽高和在相机的坐标、相对 Y 轴旋转角度. 激光雷达点云数据以浮点二进制文件格式存储.

2.2 实验设置

采用两阶段的训练方式, 对于训练集中的每个 3D 点云场景, 从每个场景中抽取 16 384 个点作为输入, 不足 16 384 个点的场景, 随机重复抽取以达到 16 384 个点. 在骨干网络中, 使用 4 个多尺度分组的集合抽象层, 将输入的点上采样为 4 096, 1 024, 256, 64 的组. 然后, 使用 4 个特征传播层获取点特征向量, 用于分割和生成预选框细化子网络. 从每个感兴趣的集合区域随机抽取 512 个点作为细化子网络的输入, 为了使对象置信度分类和提案位置细化, 采用 3 个单尺度分组的集合抽象层(组大小为 128, 32, 1)生成单个特征向量. 在 KITTI 数据集中, 汽车类的样本数量最多, 对其训练细节的描述也适用于其他类别(如行人和自行车). 将 3D 真值框内的所有点视为前景点, 其余点视为背景点, 在训练过程中, 忽略物体边界附近的背景点. 由于 3D 真值框可能会有小的变化, 所以, 通过在物体的每一侧将 3D 真值框放大 0.2 m 进行鲁棒分割. 对于 bin 机制, 参数设置的搜索范围 $S=3$ m, 2 个垂直方向共划分 12 个区间, 每个区间为 0.5 m.

评估时, 难度等级 Easy, Moderate, Hard 表示的最小边界框高度分别为 40, 25, 25 px, 最大遮挡程度分别为完全可见、部分可见、很难看到, 最大的截断率分别为 15%, 30%, 50%. 在 PointRCNN 开源代码的基础上进行修改, 为了便于对比分析, 实验的参数设置与原模型一致, 只在第 1 阶段的骨干网络部分将 PConv 动态卷积嵌入其中进行改进. 为了增加额外的非重叠真值框, 在场景中使用随机翻转、缩放、旋转和数据增强等方法. 两个阶段的子网络分开训练, 在训练过程中, 第 1 阶段采用单个图形处理器(GPU)方式进行训练; 第 2 阶段采用两种训练方式, 即在线的数据增强和离线的数据增强, 在线数据增强易于训练, 离线数据增强训练效果更好.

2.3 实验对比结果

为了验证文中算法的有效性, 根据 PointRCNN 提供的代码指令, 采用在线和离线两种数据增强方式, 在汽车类数据集中进行复现, 在交并比阈值(IOU)为 0.7 的情况下, 记录平均检测精度(P_A). 汽车类数据集上文中算法与现有算法、PointRCNN 算法的平均检测精度对比, 如表 1 所示. 表 1 中: 黑体数据表示最好的实验结果.

表 1 不同算法在汽车类 KITTI 验证集上的平均检测精度对比

Tab. 1 Comparison of average detection precision of different algorithms on car class of KITTI val split

算法	$P_A/\%$			算法	$P_A/\%$		
	Easy	Moderate	Hard		Easy	Moderate	Hard
文献[16]算法	71.29	62.68	56.56	PointRCNN(在线)	85.64	75.44	72.54
文献[1]算法	81.98	65.46	62.85	PointRCNN(离线)	86.37	76.60	73.80
文献[2]算法	87.43	76.48	69.10	文中算法(在线)	84.95	75.57	72.88
文献[17]算法	84.41	74.44	68.65	文中算法(离线)	85.87	77.02	74.46
文献[18]算法	83.76	70.92	63.65				

由表 1 可知: 文中算法在 PointRCNN 的基础上融入 PConv 动态卷积模块, 在汽车类数据集的 Moderate, Hard 等级难度下, 文中算法的平均检测精度均优于现有其他算法.

在行人和自行车类数据集上进行实验, 文中算法与 PointRCNN 算法的平均检测精度对比, 如表 2

所示.由表2可知:由于行人和自行车类数据集的体积小且样本稀疏,文中算法在行人数据集 Hard 等级难度下的检测精度较好,在自行车类数据集 Easy, Moderate 等级难度下的检测精度更好.由此可见,文中算法在遮挡严重且样本稀疏的目标检测上具有优越性.

表2 不同算法在行人和自行车类 KITTI 验证集上的平均检测精度对比

Tab. 2 Comparison of average detection precision of

different algorithms on pedestrian and bicycle class of KITTI val split

算法	$P_A/\%$ (行人类)			$P_A/\%$ (自行车类)		
	Easy	Moderate	Hard	Easy	Moderate	Hard
PointRCNN(离线)	67.03	59.93	54.89	82.39	64.37	61.58
文中算法(离线)	66.90	59.54	55.42	82.91	65.25	61.04

为了验证文中算法的可行性,采用文中算法和 PointRCNN 算法对两阶段进行训练,其训练时间的对比,如表3所示.

表3中: t_1, t_2 分别为第1阶段、第2阶段的训练时间.由表3可知:文中算法比 PointRCNN 算法所用的训练时间更少,再次验证了文中算法的优越性.

表3 训练时间的对比

Tab. 3 Comparison of training time

算法	t_1/ms	t_2/ms
PointRCNN	18	129
文中算法	20	123

2.4 消融实验

为了探究不同的输入方式对 PAConv 动态卷积效果的影响,采用4种不同的输入方式进行消融分析实验.以中心点 A_i 和邻近点 A_j 两者之间的关系设计4

种不同的任务, $O_{i,j}$ 表示 A_i 和 A_j 的欧式距离.不同输入方式下的召回率(η_R),如表4所示.由表4可知:将邻近点与中心点的距离差和中心点一起输入,召回率最高,加入两点的欧式距离及邻近点后,召回率变差,表明在大型的户外场景中运用动态卷积自适应时,输入的越多,越容易产生误差,从而影响检测效果.因此,选择 $(A_j - A_i, A_i)$ 作为输入,该输入方式的召回率高,得到的最终检测精度最好.

表4 不同输入方式下的召回率

Tab. 4 Recall rate under different input modes

输入方式	η_R	
	第1阶段	第2阶段
$(A_j - A_i)$	0.69	0.82
$(A_j - A_i, A_i)$	0.73	0.85
$(A_j - A_i, A_i, O_{i,j})$	0.71	0.83
$(A_j - A_i, A_j, A_i, O_{i,j})$	0.72	0.84

2.5 效果可视化

在 KITTI 训练集上对汽车、行人和自行车三类进行标记,采用分开训练的方式,将得到的模型可视化展示.汽车、行人和自行车类的检测对比图,分别如图7~9所示.图7~9中:图(a)为2D标记框;图(b)为3D标记框;图(c)为预测框;图(d)为预测框与3D标记真值框的对比;图(e)为图(d)部分结果的放大效果;绿色为汽车类标记框;蓝色为行人标记框;黄色为自行车类标记框;红色为预测框;红色箭头表示方向.

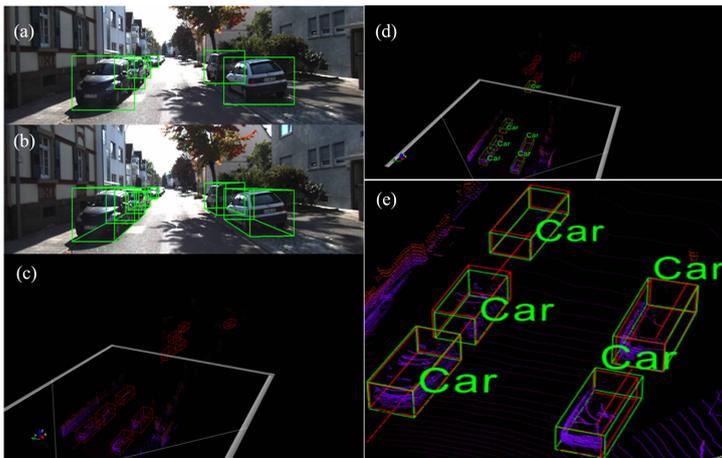


图7 汽车类的检测对比图

Fig. 7 Car detection comparison chart

由图7~9可知:2D标记框的遮挡问题在3D点云场景中得到很好地解决,因为3D检测框是立体

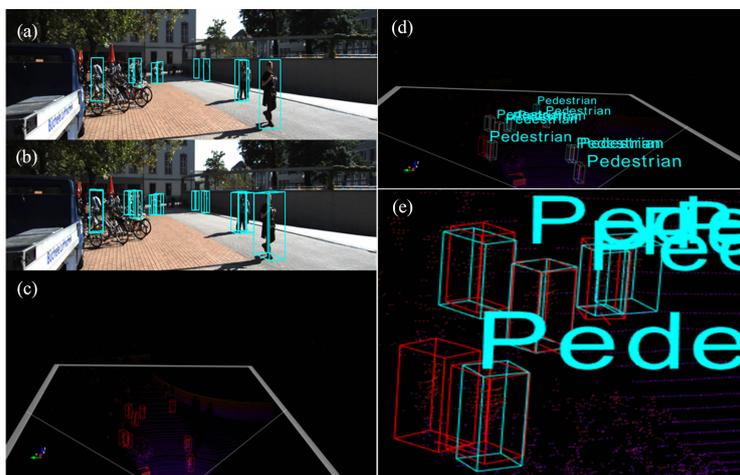


图 8 行人类的检测对比图

Fig. 8 Pedestrian detection comparison chart

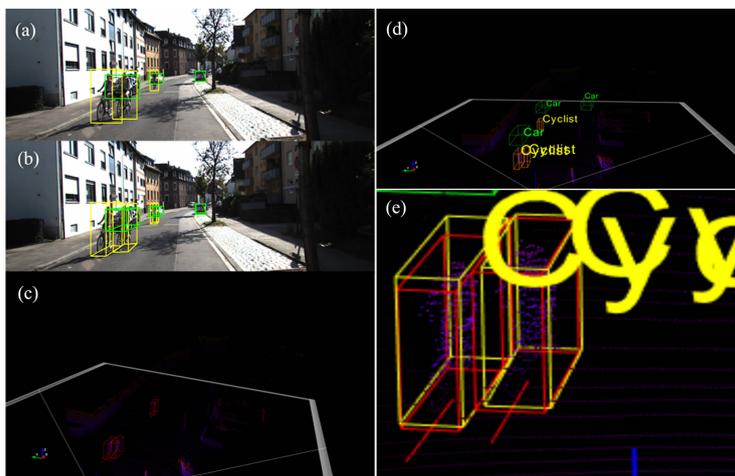


图 9 自行车类的检测对比图

Fig. 9 Bicycle detection comparison chart

的,使检测框得到很好的分离,显示出 3D 目标检测的优越性;在汽车、行人、自行车类的检测效果图中,预测框与标记框的误差小,证明了文中算法的有效性。

3 结束语

提出一种将动态卷积作为特征提取的 3D 点云目标检测算法. 针对点云的不规则无序性问题,相较于现有算法,文中算法的优点是运用动态卷积的方法,可以更好地提取关键特征. 文中算法采用 PAConv 动态卷积的方式,通过相对点位置自适应地学习生成分配系数,从而动态调整权重,有效地避免了因遮挡和样本稀疏造成的模型不精确问题. 从 KITTI 数据集的训练结果可知,在汽车类的 Moderate, Hard 等级难度下,文中算法的平均检测精度优于现有算法;由于行人和自行车类数据集的体积小且样本稀疏,在行人类数据集的 Hard 等级难度下,文中算法的平均检测精度较好,在自行车类数据集的 Easy, Moderate 等级难度下,文中算法的平均检测精度较好. 实验结果表明:针对不规则无序、遮挡、稀疏的点数据,文中算法具有更好的检测精度。

参考文献:

- [1] ZHOU Yin, TUZEL O. Voxelnet: End-to-end learning for point cloud based 3D object detection[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE Press, 2018: 4490-4499.
- [2] YAN Yan, MAO Yuxing, LI Bo. Second: Sparsely embedded convolutional detection[J]. Sensors, 2018, 18(10):

3337. DOI:10.3390/s18103337.
- [3] GRAHAM B, ENGELCKE M, VAN DER MAATEN L. 3D semantic segmentation with submanifold sparse convolutional networks[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City; IEEE Press, 2018; 9224-9232. DOI:10.48550/arXiv.1711.10275.
- [4] 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, Long Beach; IEEE Press, 2019; 12689-12697. DOI:10.1109/CVPR.2019.01298.
- [5] SHI Shaoshuai, GUO Chaoxu, JIANG Li, *et al.* PV-RCNN: Point-Voxel feature set abstraction for 3D object detection[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach; IEEE Press, 2021; 10529-10538. DOI:10.48550/arXiv.1912.13192.
- [6] DENG Jiajun, SHI Shaoshuai, LI Peiwei, *et al.* Voxel R-CNN: Towards high performance Voxel-based 3D object detection[C]// AAAI Conference on Artificial Intelligence. [S. l.]: AAAI Press, 2021; 1201-1209.
- [7] MA Ruiqi, CHEN Chi, YANG Bisheng, *et al.* CG-SSD: Corner guided single stage 3D object detection from LiDAR point cloud[EB/OL]. (2022-02-24)[2022-03-28]. <https://doi.org/10.48550/arXiv.2202.11868>.
- [8] XU Wencai, HU Jie, CHEN Ruinan, *et al.* Keypoint-aware single-stage 3D object detector for autonomous driving[J]. *Sensors*, 2022, 22; 1451. DOI:10.3390/s22041451.
- [9] QI C R, SU Hao, MO Kaichun, *et al.* PointNet: Deep learning on point sets for 3D classification and segmentation[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu; IEEE Press, 2017; 77-85. DOI:10.1109/CVPR.2017.16.
- [10] QI C R, YI Li, SU Hao, *et al.* PointNet++: Deep hierarchical feature learning on point sets in a metric space[C]// Annual Conference on Neural Information Processing Systems, Long Beach; NIPS Press, 2017; 5099-5108.
- [11] SHI Shaoshuai, WANG Xiaogang, LI Hongsheng. PointRCNN: 3D object proposal generation and detection from point cloud[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Los Angeles; IEEE Press, 2019; 770-779. DOI:10.48550/arXiv.1812.04244.
- [12] YANG Zetong, SUN Yanan, LIU Shu, *et al.* 3DSSD: Point-based 3D single stage object detector[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle; IEEE Press, 2020; 11040-11048. DOI:10.1109/CVPR42600.2020.01105.
- [13] SHI Weijing, RAJKUMA R. Point-GNN: Graph neural network for 3D object detection in a point cloud[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle; IEEE Press, 2020; 1708-1716. DOI:10.1109/CVPR42600.2020.00178.
- [14] XU Mutian, DING Runyu, ZHAO Hengshuang, *et al.* PAConv: Position adaptive convolution with dynamic kernel assembling on point clouds[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition. [S. l.]: IEEE Press, 2021; 3172-3181. DOI:10.48550/arXiv.2103.14635.
- [15] LIN T Y, GOYAL P, GIRSHICK R, *et al.* Focal loss for dense object detection[C]// IEEE International Conference on Computer Vision, Venice; IEEE Press, 2017; 2999-3007. DOI:10.1109/TPAMI.2018.2858826.
- [16] CHEN Xiaozhi, MA Huimin, WAN Ji, *et al.* Multi-view 3D object detection network for autonomous driving[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Honolulu; IEEE Press, 2017; 6526-6534. DOI:10.1109/CVPR.2017.691.
- [17] KU J, MOZIFIAN M, LEE J, *et al.* Joint 3D proposal generation and object detection from view aggregation[C]// IEEE/RSJ International Conference on Intelligent Robots and Systems, Madrid; IEEE Press, 2018; 5750-5757. DOI:10.48550/arXiv.1712.02294.
- [18] QI C R, LIU Wei, WU Chenxia, *et al.* Frustum PointNets for 3D object detection from RGB-D data[C]// IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City; IEEE Press, 2018; 918-927. DOI:10.48550/arXiv.1711.08488.

(责任编辑:黄晓楠 英文审校:吴逢铁)