基于卦限卷积神经网络的 3D 点云分析

许 翔 [1] 帅 惠 [1] 刘青山 [1, 2]

摘 要 基于深度学习的三维点云数据分析技术得到了越来越广泛的关注, 然而点云数据的不规则性使得高效提取点云中的局部结构信息仍然是一大研究难点. 本文提出了一种能够作用于局部空间邻域的卦限卷积神经网络 (Octant convolutional neural network, Octant-CNN), 它由卦限卷积模块和下采样模块组成. 针对输入点云, 卦限卷积模块在每个点的近邻空间中定位 8 个卦限内的最近邻点, 接着通过多层卷积操作将 8 卦限中的几何特征抽象成语义特征, 并将低层几何特征与高层语义特征进行有效融合, 从而实现了利用卷积操作高效提取三维邻域内的局部结构信息; 下采样模块对原始点集进行分组及特征聚合, 从而提高特征的感受野范围, 并且降低网络的计算复杂度. Octant-CNN 通过对卦限卷积模块和下采样模块的分层组合, 实现了对三维点云进行由底层到抽象、从局部到全局的特征表示. 实验结果表明, Octant-CNN 在对象分类、部件分割、语义分割和目标检测四个场景中均取得了较好的性能.

关键词 深度学习, 点云, 卦限卷积神经网络, 局部几何特征

引用格式 许翔, 帅惠, 刘青山. 基于卦限卷积神经网络的 3D 点云分析. 自动化学报, 2021, **47**(12): 2791−2800

DOI 10.16383/j.aas.c200080

Octant Convolutional Neural Network for 3D Point Cloud Analysis

XU Xiang[1] SHUAI Hui[1] LIU Qing-Shan[1, 2]

Abstract The 3D point cloud data analysis based on deep learning has attracted increasing attention recently. However, it is still a great challenge to extract local structure information from point cloud efficiently due to its irregularity. In this paper, we propose a new network named octant convolutional neural network (Octant-CNN) which can handle local spatial neighborhoods. It consists of octant convolution module and sub-sampling module. For the input point cloud, the octant convolution module locates nearest points in eight octants of each point, and then transforms the geometric features into semantic features through a multi-layer convolution operation. The low-level geometric features are effectively fused with the high-level semantic features so that the local structure information can be efficiently extracted. The sub-sampling module groups the original point set and aggregates the features to expand the receptive field of features, and also reduce the computation overhead of the network. By stacking the octant convolution module and sub-sampling module, Octant-CNN obtains the feature representation of the 3D point cloud from low-level to abstract, and from local to global. Extensive experiments demonstrate that Octant-CNN achieves great performance in four 3D scene understanding tasks including object classification, part segmentation, semantic segmentation, and object detection.

Key words Deep learning, point cloud, octant convolutional neural network (Octant-CNN), local geometric feature

Citation Xu Xiang, Shuai Hui, Liu Qing-Shan. Octant convolutional neural network for 3D point cloud analysis. *Acta Automatica Sinica*, 2021, **47**(12): 2791−2800

随着自动驾驶和机器人应用技术的兴起, 3D 点云数据分析引起了广泛关注. 近年来, 由于基于深度学习的神经网络在图像分类[1−2]、目标检测[3−4]和图像分割[5−6] 等任务中取得了很大的成功, 基于深度学习的点云数据分析也成为了研究的热点[7]. 现有的基于深度学习的点云数据分析方法大体可以分为以下两类:

一类是基于无序点云规则化的深度学习方法, 这类方法先将 3D 点云转换为规则的体素结构[8−9]或多视图图像[10−11], 然后使用卷积神经网络 (Convolutional neural network, CNN) 方法来学习特征表示. 由于体素化过程存在量化误差, 多视图投影则压缩了数据维度, 这些都会不同程度上导致 3D 点云中几何信息的丢失. 另一类方法是直接基于点云的深度学习方法. 这类方法又可以分为基于多层

收稿日期 2020-02-25 录用日期 2020-07-21
Manuscript received February 25, 2020; accepted July 21, 2020
国家自然科学基金 (61825601, 61532009), 江苏省研究生科研创新计划 (KYCX21_0995) 资助
Supported by National Natural Science Foundation of China (61825601, 61532009) and Postgraduate Research and Practice Innovation Program of Jiangsu Province (KYCX21_0995)
本文责任编委 吴毅红
Recommended by Associate Editor WU Yi-Hong
1. 江苏省大数据分析技术重点实验室 南京 210044 2. 南京信息工程大学计算机学院、软件学院、网络空间学院 南京 210044
1. Jiangsu Key Laboratory of Big Data Analysis Technology, Nanjing 210044 2. School of Computer Science, Nanjing University of Information Science and Technology, Nanjing 210044

感知机 (Multi-layer perceptron, MLP) 的方法、基于卷积的方法和基于图的方法. 其中基于多层感知机的方法[12−14] 的核心思想是通过参数共享的 MLP 独立地提取每个点的特征, 然后通过一个对称函数聚合得到全局特征, 这类方法往往不能充分考虑到 3D 点之间的关系. 基于卷积的方法[15−17] 的核心思想是根据邻域点之间的空间位置关系去学习点之间的权重参数, 并根据学习到的权重参数自适应地聚合局部特征, 这类方法已经取得了极大的成功. 基于图的方法[18−20] 在近年来也受到了广泛的关注, 它们将每个点都作为图的顶点, 通过学习顶点之间边的权重来更新顶点的特征, 这类方法通常在构图的过程中会产生相当大的计算量.

在上述方法中, 基于 MLP 的方法是最直接简单的方法. PointNet[12] 是这类方法中的开创性工作, 其核心思想是通过参数共享的多层感知机独立地将每个点的坐标信息映射至高维特征空间, 再通过一个对称函数聚合最终的高维特征以获得全局表示, 从而解决了点云的无序性问题; 此外, PointNet 还使用 T-Net 网络[12] 学习变换矩阵对点云进行旋转标定, 从而保证点云的旋转不变性; 在分割任务中, PointNet 将全局特征与每个点的局部特征级联, 通过多层 MLP 提取每个点的语义特征, 实现对每个点的分类. 虽然该方法简单有效, 但是由于其是对每个点进行独立地处理, 因此该网络并没有有效提取点云的局部特征. 对此, PointNet++[13] 提出了一种层次化的网络结构, 通过在每一层级递归使用采样、分组和 PointNet 网络来抽象低层次的特征; 面对语义分割任务, PointNet++提出基于欧氏距离的插值法对点进行上采样, 并将通过插值计算所得语义特征与低层学习的语义特征进行融合以更准确地学习每个点的语义特征. 但是在每一个子区域中, PointNet++仍然独立地处理每个点的信息. PointSIFT[14] 引入卦限约束来有效探索各个点周围的局部模式, 其主要思想是以每个点为原点, 在周围 8 个卦限中找到特定范围内的最近点, 然后沿着 X, Y, Z 轴使用三阶段 2D 卷积来提取局部模式, 其三阶段的卷积操作会受到因点云旋转而造成的不同卦限顺序的影响, 从而使得提取的局部模式具有方向敏感性; 此外, 在下采样阶段, PointSIFT 沿用 PointNet++的网络结构, 采用可学习的方式聚合局部特征, 这为其引入额外的参数, 从而大大增加了其计算量.

为了克服上述问题, 本文提出了一种新的卦限卷积神经网络 (Octant-CNN) 来提取点云的局部几何结构. 该网络主要由卦限卷积模块和下采样模块两部分组成. 具体来说, 卦限卷积模块首先搜索每个点在 8 个卦限内的最近邻点, 由于点云的密度特性可以通过近邻点的距离来表征, 为了使 Octant-CNN 能更好地反映这一特性, 本文取消了对搜索半径的限制, 从而保证远离中心点的近邻点同样可以被度量. 卦限卷积模块使用单阶段卷积操作同时作用在 8 个卦限的近邻点, 从而克服了三阶段卷积操作对卦限顺序敏感这一问题, 并且配合 T-Net 的使用, 能够对点云旋转具有更好的鲁棒性. 最后通过级联各层的特征和残差连接方式实现了多层次特征的融合. 下采样模块根据空间分布对点云进行分组聚合, 扩大了中间特征的感受野, 构成了层次化的网络连结结构, 并且该模块并没有引入额外的可学习参数, 从而大大降低了 Octant-CNN 的计算复杂度. 通过对卦限卷积模块和下采样模块的多层堆叠, Octant-CNN 实现了从局部模式中不断抽象出全局特征.

1 Octant-CNN

Octant-CNN 的整体网络框架如图 1 所示. 以原始点云作为输入, 首先将点云送入 T-Net 中进行点云旋转, 将点云标定至规范空间, 接着通过卦限卷积模块 (Octant convolution module) 提取点云的局部几何结构, 其后采用下采样模块 (Sub-sampling module) 来减少点的数量, 以设计一种分层式的层次化网络结构, 从而增加中间层特征的感受野. 通过这两个模块的多层堆叠, Octant-CNN实现了对高层语义特征的抽象, 为点云处理提供了一种高效的特征编码方式.

1.1 卦限卷积模块

假设具有 n 个点的点云为 $S = (X; F) \subseteq \mathbf{R}^{3+C}$, 其中 $X = \{x_1, x_2, \cdots, x_n\} \subseteq \mathbf{R}^3$ 表示坐标信息, $F = \{f_1, f_2, \cdots, f_n\} \subseteq \mathbf{R}^C$ 表示点云的特征. 对于每个点 s_i, 以该点为原点建立一个三维局部坐标系, 可以将空间划分为 8 个卦限, 然后在 8 个卦限中分别找到 s_i 的最近邻点, 即 $N(s_i) = \{s_{i1}, \cdots, s_{i8}\}$. 在卦限卷积模块中, Octant-CNN 取消了搜索半径上限的限制, 这样可以确保远离中心点的近邻点同样可以被捕获到, 从而可以更好地反映点云的局部密度特性.

对于 8 个最近邻点, PointSIFT[14] 使用了具有三阶段操作的 2D 卷积, 如图 2(a) 所示. 该卷积操作沿 X, Y 和 Z 轴分别使用卷积核大小为 1×2 的 2D 卷积. 这种三阶段的卷积操作存在着先后顺序, 对于三维空间中的不同维度具有各向异性, 而且 PointSIFT 中没有采用 T-Net 对输入点云进行旋

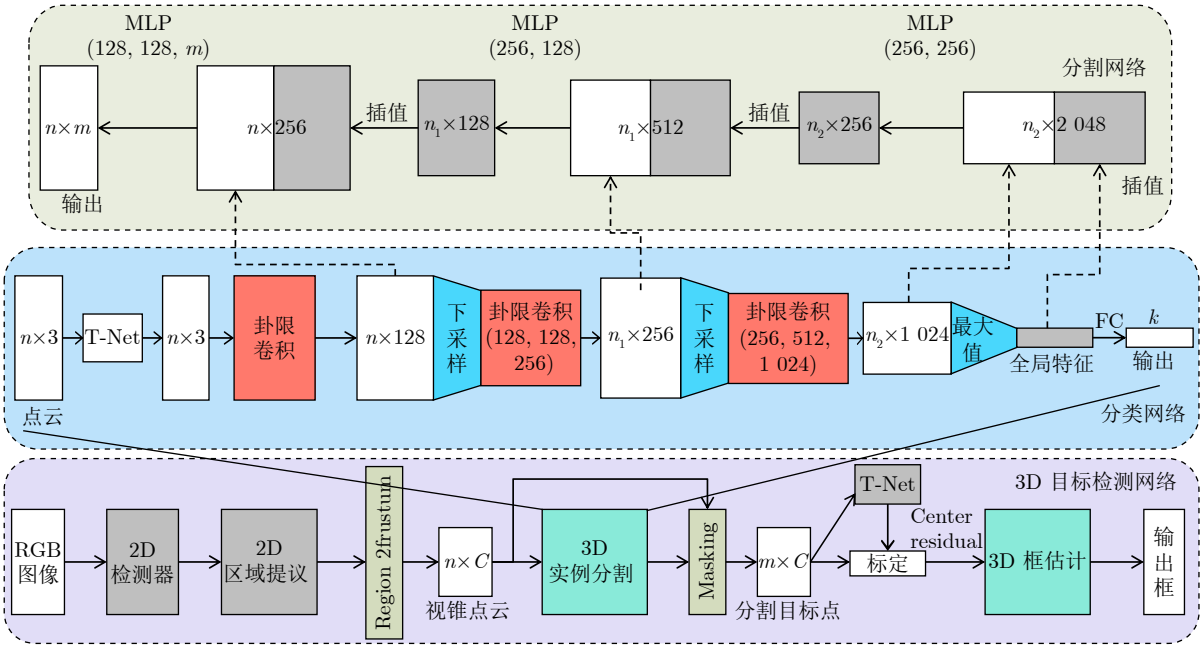
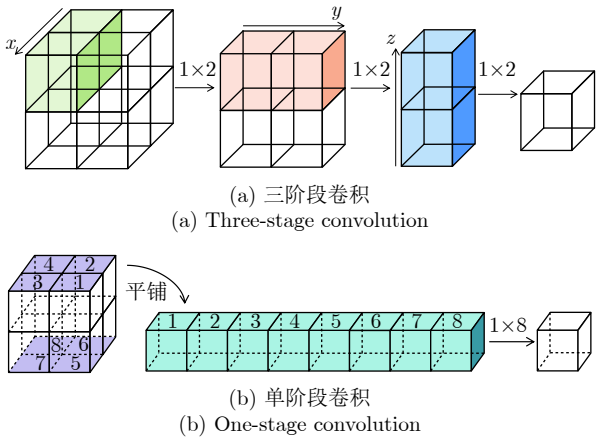


图 1　网络框架图

Fig. 1　Illustration of network architecture



(a) 三阶段卷积
(a) Three-stage convolution

(b) 单阶段卷积
(b) One-stage convolution

图 2　三阶段与单阶段 2D 卷积的对比

Fig. 2　Comparison of 2D CNN with three-stage and one-stage

转, 因此不同的卦限顺序会造成不同的卷积结果. 为了克服这个问题, 本文采用 T-Net 对点云进行旋转标定, 并通过单阶段的卷积操作直接在 8 个最近邻点上进行运算. 如图 2(b) 所示, Octant-CNN 先按照卦限的顺序对 8 个邻点进行排序, 接着利用一个卷积核大小为 1×8 的 2D 卷积直接作用在这 8 个邻点上. 给定中心点 s_i 及其最近邻点 $N(s_i)$, 卷积的输入通道包括中心点坐标 x_i, 中心点及其最近邻点之间的残差坐标 $x_i - x_{ij}$, 以及最近邻点的特征 f_{ij}, 该操作过程如下:

$$f^{(l)}(s_i) = \sum_{j=1}^{8} w_{ij}\left(concat(x_i, x_i - x_{ij}, f_{ij})\right) \tag{1}$$

其中, $f^{(l)}(s_i)$ 表示点 s_i 在第 l 层学到的特征, *concat* 表示级联操作, w_{ij} 表示 s_i 和 s_{ij} 之间的可学习权重.

由于 Octant-CNN 先通过 T-Net 对点云方向进行预先标定, 其后采用一个二维卷积同等处理各卦限内的点及其特征, 使得单阶段卷积对输入点云具有各向同性, 因此对于不同角度的同一点云输入, Octant-CNN 总能得到相似的特征表示, 具有旋转不变性.

为了使每个点能够提取更丰富的特征, Octant-CNN 在卦限卷积模块中堆叠了多层卷积操作, 并将各层的输出特征通过 MLP 进行融合, 以充分利用各层次特征信息, 并且 MLP 的输出尺寸与最后一个卷积层相同, 从而可以在最后一层添加残差块以缓解梯度消失问题. 整个卦限卷积模块可以表示为

$$f_O = F_{\text{res}} + F^{(l)} = MLP\left(F^{(1)}, F^{(2)}, \cdots, F^{(l)}\right) + F^{(l)} \tag{2}$$

卦限卷积模块的架构如图 3 所示.

1.2　下采样模块

下采样模块的目的是为了扩大每个点特征学习的局部感受野. 主要思路为: 从输入点集中选择一系列种子点作为聚类中心点; 然后, 将这些中心点周围的点的特征用对称函数聚合在一起. 由于 PointSIFT[14] 在下采样的过程中沿用了 PointNet++[13] 的结构设计, 这为 PointSIFT 引入了可学习的参

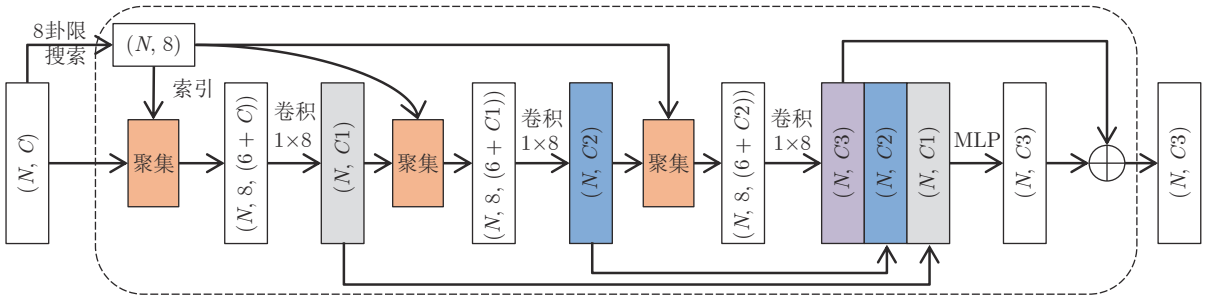


图 3 卦限卷积模块

Fig. 3 Octant convolution module

数, 从而增加了其计算量; 不同于此, Octant-CNN 在下采样模块中的核心操作是种子点的选择和特征聚合, 而在特征聚合时仅仅通过对称函数即可实现, 这并没有为 Octant-CNN 带来额外的可学习参数, 因此降低了 Octant-CNN 的计算复杂度.

给定输入点 $\{x_1, x_2, \cdots, x_n\}$, 下采样模块迭代使用最远点采样 (Furthest point sampling, FPS) 来选择该点集的一个子集 $\{x_1, x_2, \cdots, x_m\}$, $m < n$, 并将该子集作为聚类中心的种子点. 与随机采样相比, FPS 通过最大化采样点之间的距离来更好地覆盖整个点集[13].

对于每一个采样点, Octant-CNN 都可以在一定的半径内寻找它的邻点. 为了保持一致性, 本文在实际操作中设置了一个上限 K. 该操作可以表示为: 给定一个大小为 $N \times C$ 的原始点集, 以及通过 FPS 采样得到的大小为 $M \times C$ 的子集, 其中 M 和 N 表示集合中点的数量 ($M < N$), C 表示特征维度. 对于每个采样点, 都可以在一定半径内从原始点集中选取 K 个邻点, 并输出大小为 $M \times K \times C$ 的数据. 这些邻点的特征都可以通过一个对称函数聚合并输出大小为 $M \times C$. 在实际操作中, 本文使用最大值来聚合局部特征.

2 实验结果与分析

为了详细评估 Octant-CNN 的性能, 本文在对象分类、部件分割、场景语义分割、3D 目标检测四组任务中, 对其进行了实验测试, 并和相关方法进行了比较. 此外, 本文还通过一系列消融实验评估了卦限卷积和下采样模块的不同设置对网络性能的影响.

2.1 对象分类

首先在 ModelNet40[9] 分类基准上评估 Octant-CNN. 该数据集包含 40 个人工设计的对象类别, 共有 12 311 个 CAD 模型, 其中 9 843 个用于训练集, 2 468 个用于测试集. 参照 PointNet[12], 本文均匀采样 1 024 个点并将其标准化到单位球体中, 并仅将采样点的坐标作为模型的输入. 在训练过程中, 本文与 PointNet++[13] 一样, 通过随机旋转和缩放对象并扰动对象点的位置来扩充数据.

如图 1 所示, 首先使用 PointNet[12] 设计的 T-Net 对点云进行旋转标定. T-Net 首先通过三层共享的 MLP 提取点的特征, 然后通过最大值池化以获取全局表示, 最后通过两个全连接层来计算一个转换矩阵. Octant-CNN 在卦限卷积模块中学习点的局部特征, 然后在下采样模块中对点进行分组聚合局部特征. 在实际操作中, 对于最后一个下采样模块, 本文仅对原点进行采样, 然后使用最大值获取全局特征, 最后通过两层全连接层来输出对象的类别概率. 在训练过程中, 本文在全连接层中使用了 dropout[21] 机制, 并将该比率设置为 50%. 在测试阶段, 本文和 PointNet++[13] 一样, 使用投票机制将点云均匀旋转 12 个不同的角度后分别送入模型中预测, 并对这 12 个预测结果取平均获取最终的分类结果.

表 1 列出了 Octant-CNN 与最新一些相关方法进行的比较结果, 包括 PointNet[12], PointNet++[13], PAT[22] 等. 本文采用了整体准确率 (Overall accuracy, oAcc) 和平均准确率 (Mean accuracy, mAcc) 两种指标来衡量分类结果, 它们的定义分别为

$$oAcc = \frac{\sum_{i=1}^{N} p_{ii}}{\sum_{i=1}^{N}\sum_{j=1}^{N} p_{ij}} \tag{3}$$

$$mAcc = \frac{1}{N}\sum_{i=1}^{N}\frac{p_{ii}}{\sum_{j=1}^{N} p_{ij}} \tag{4}$$

其中, p_{ij} 表示真实标签为 i, 预测结果为 j 的数量, N 表示类别数. 为了客观分析比较, 本文还实现了基于 PointSIFT[14] 的对象分类任务. 从表 1 可以看到, Octant-CNN 取得了不错的效果, 这也说明了 Octant-CNN 在一定程度上可以更好地学习到点云的局部几何特征.

表 1　ModelNet40 分类结果 (%)
Table 1　Classification results on ModelNet40 (%)

方法	oAcc	mAcc
PointNet[12]	89.2	86.2
PointNet++[13]	90.7	—
PointSIFT[14]	90.2	86.9
SFCNN[15]	91.4	—
ConvPoint[17]	91.8	88.5
ECC[18]	87.4	83.2
RGCNN[19]	90.5	87.3
PAT[22]	91.7	—
SCN[23]	90.0	87.6
SRN-PointNet++[24]	91.5	—
JUSTLOOKUP[25]	89.5	86.4
Kd-Net[26]	91.8	88.5
SO-Net[27]	90.9	87.2
Octant-CNN	**91.9**	**88.7**

2.2　部件分割

ShapeNet[28] 数据集主要用于测试部件分割任务. 该数据集包含 16 个对象类别的 16 881 个不同形状, 总共被标记为 50 个部件. 本文参照 PointNet[12] 的方法对数据集进行划分, 并随机采样 2 048 个点作为网络输入. Octant-CNN 仅使用坐标信息作为网络的输入, 而没有采用 PointNet++[13] 中的法线信息.

但是, 对于分割任务, 模型希望获得每个点的语义特征以实现每个点的分类. PointSIFT[14] 首先参照 PointNet++[13] 的方法, 先使用基于欧氏距离的插值法对点进行上采样, 并将内插值与上一个卦限卷积模块中学习的特征进行级联, 然后通过共享的多层感知机提取丰富的语义特征; 紧接着, PointSIFT 在此基础上使用三阶段卷积操作进一步做特征变换. 该三阶段的卷积操作在原来多层感知机的基础上又引入了额外的参数. 不同于此操作, 考虑到计算量的问题, Octant-CNN 在上采样的过程中仅通过多层感知机来抽象高层的语义特征.

在实际操作中, 我们还将对象的 one-hot 标签级联到最后一层特征传播层中, 以进行准确的预测. 为了更好地评测 Octant-CNN 在部件分割上的性能, 本文还和 PointNet[12], PointNet++[13] 等方法进行了实验比较, 表 2 中给出了实验比较结果. 本文采用平均交并比 (Mean intersection over union, mIoU) 作为衡量分割任务性能的指标, 其定义为

$$mIoU = \frac{1}{N}\sum_{i=1}^{N}\frac{p_{ii}}{\sum\limits_{j=1}^{N}p_{ij}+\sum\limits_{j=1}^{N}p_{ji}-p_{ii}} \tag{5}$$

其中, p_{ij} 表示真实标签为 i, 预测结果为 j 的数量, N 表示类别数. 可以看到, 本文仅将坐标信息用作输入, 就可以得到比使用法线信息的 PointNet++[13] 更好的性能. 同时, PointSIFT[14] 在部件分割任务中并不能取得很好的效果, 主要由于其上采样使用了三阶段的卷积操作, 这带来了大量的参数, 对于 ShapeNet[28] 这种相对较小的数据集, 很容易造成模型的过拟合.

2.3　室内场景语义分割

为了进一步证明 Octant-CNN 的有效性, 本文还在斯坦福大学大型 3D 室内空间数据集 (3d semantic parsing of large-scale indoor spaces, S3DIS)[30] 上评估了其性能. 该数据集包含来自 6 个室内区域的 272 个房间. 每个点都用来自 13 个类别 (天花板, 地板, 墙壁, 梁和其他) 的语义标签进行标注. 参照 PointNet[12], 本文将每个房间分成面积为 1 m × 1 m

表 2　ShapeNet 部件分割结果 (%)
Table 2　Part segmentation results on ShapeNet (%)

方法	mIoU	aero	bag	cap	car	chair	earphone	guitar	knife	lamp	laptop	motor	mug	pistol	rocket	skateboard	table
PointNet[12]	83.7	83.4	78.7	82.5	74.9	89.6	73.0	91.5	85.9	80.8	95.3	65.2	93.0	81.2	57.9	72.8	80.6
PointNet++[13]	85.1	82.4	79.0	87.7	77.3	90.8	71.8	91.0	85.9	**83.7**	95.3	71.6	94.1	81.3	58.7	**76.4**	**82.6**
PointSIFT[14]	79.0	75.1	78.4	81.8	74.5	85.2	64.3	89.6	81.9	77.5	95.1	64.0	93.5	77.1	54.2	70.6	74.3
RGCNN[19]	84.3	80.2	82.8	**92.6**	75.3	89.2	73.7	91.3	**88.4**	83.3	**96.0**	63.9	**95.7**	60.9	44.6	72.9	80.4
DGCNN[20]	85.1	**84.2**	83.7	84.4	77.1	90.9	**78.5**	91.5	87.3	82.9	**96.0**	67.8	93.3	82.6	59.7	75.5	82.0
SCN[23]	84.6	83.8	80.8	83.5	79.3	90.5	69.8	91.7	86.5	82.9	**96.0**	69.2	93.8	82.5	**62.9**	74.4	80.8
Kd-Net[26]	82.3	80.1	74.6	74.3	70.3	88.6	73.5	90.2	87.2	81.0	94.9	57.4	86.7	78.1	51.8	69.9	80.3
SO-Net[27]	84.6	81.9	83.5	84.8	78.1	90.8	72.2	90.1	83.6	82.3	95.2	69.3	94.2	80.0	51.6	72.1	**82.6**
RS-Net[29]	84.9	82.7	**86.4**	84.1	78.2	90.4	69.3	91.4	87.0	83.5	95.4	66.0	92.6	81.8	56.1	75.8	82.2
Octant-CNN	**85.3**	83.9	83.6	88.3	**79.2**	**91.1**	70.8	**91.8**	87.5	82.9	95.7	**72.2**	94.5	**83.6**	60.0	75.5	81.9

的块, 每个点都表示为 9 维向量 (XYZ, RGB 和归一化坐标). 在训练过程中, Octant-CNN 在每个块中随机选取 4 096 个点, 并将所有的点用于测试. 与 PointNet[12] 一样, 本文在 6 个区域上使用了 6 折交叉验证的方式.

本文将 Octant-CNN 与 PointNet[12], PointNet++[13], PointSIFT[14] 进行了比较. 由于我们无法达到 PointSIFT[14] 中报告的结果, 因此仅显示根据作者提供的代码而获得的结果. 结果总结在表 3 中, 本文提出的 Octant-CNN 优于其他方法. 图 4 显示了 Octant-CNN 的一些可视化结果, 可以发现, Octant-CNN 可以更平滑地分割场景, 这是由于 Octant-CNN 在卦限卷积模块中更好地学习局部几何特征.

2.4 3D 目标检测

最后, 本文将 Octant-CNN 和 PointSIFT[14] 扩展到了 KITTI[31] 数据集上进行 3D 目标检测. KITTI 3D 目标检测数据集由 7 481 个训练图像和 7 518 个测试图像以及相应的点云数据组成. 它具有三个目标类别: 汽车、行人和自行车. 对于 3D 目标检测, 本文遵循 Frustum PointNets[32] 的检测流程, 仅将 PointNet 特征提取模块替换成 Octant-CNN 以客观比较. 由于 Frustum PointNets[32] 仅公开了在训练集和验证集上的 2D 检测框, 因此本文评估的是 Octant-CNN 及相关方法在验证集上的检测结果.

3D 目标检测的实验结果如表 4 所示, 这些方

表 3 S3DIS 语义分割结果

Table 3 Semantic segmentation results on S3DIS

方法	mIoU	OA	ceiling	floor	wall	beam	column	windows	door	chair	table	bookcase	sofa	board	clutter
PointNet[12]	47.7	78.6	88.0	88.7	69.3	42.4	23.1	47.5	51.6	42.0	54.1	**38.2**	9.6	29.4	35.2
PointNet++[13]	57.3	83.8	91.5	92.8	74.6	41.3	28.1	54.5	59.6	64.6	58.9	27.1	**52.0**	52.3	48.0
PointSIFT[14]	55.5	83.5	91.1	91.3	75.5	42.0	24.0	51.4	56.6	60.2	55.8	17.0	50.2	**57.1**	49.9
RS-Net[29]	56.5	—	**92.5**	92.8	**78.6**	32.8	**34.4**	51.6	**68.1**	59.7	**60.1**	16.4	50.2	44.9	**52.0**
Octant-CNN	**58.3**	**84.6**	92.1	**94.5**	76.3	**48.9**	30.8	**56.9**	62.9	**65.8**	55.5	28.0	48.1	50.3	48.4

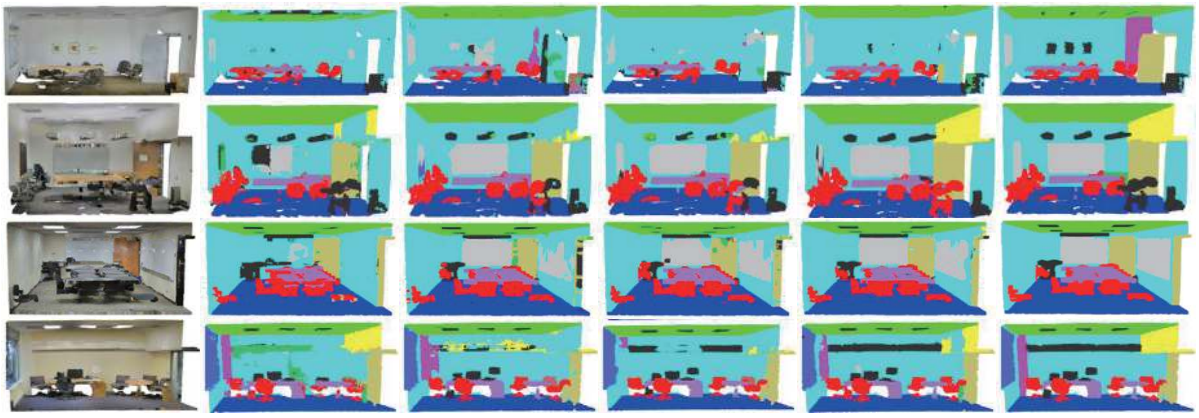


图 4 S3DIS 可视化结果

Fig. 4 Visualization of results on S3DIS

表 4 3D 目标检测对比结果 (%)

Table 4 Performance compression in 3D object detection (%)

方法	Cars			Pedestrians			Cyclists		
	Easy	Moderate	Hard	Easy	Moderate	Hard	Easy	Moderate	Hard
Frustum PointNet v1[32]	83.75	69.37	62.83	65.39	55.32	48.62	70.17	52.87	48.27
Frustum PointNet v2[32]	83.93	71.23	63.72	64.23	56.95	50.15	74.04	54.92	50.53
Frustum PointSIFT[14]	71.56	66.17	58.97	63.13	55.08	49.05	70.36	52.56	48.53
Frustum Geo-CNN[33]	85.09	71.02	63.38	**69.64**	**60.50**	**52.88**	75.64	56.25	52.54
Frustum Octant-CNN	**85.10**	**72.31**	**64.46**	67.90	59.73	52.44	**76.56**	**57.50**	**54.26**

法的检测流程都是基于 Frustum PointNets[32] 实现的, 主要不同之处在于点云的分割网络以及 3D 目标检测框的回归网络, 其中 Frustum PointNet v1 采用的是 PointNet[12] 的网络结构, Frustum PointNet v2 采用的是 PointNet++[13] 的网络结构, 可以发现, 本文提出的方法要优于这些方法. 尤其对于小目标的检测性能提升较为明显, 图 5 同时也展示了一些检测的可视化结果.

2.5　消融实验

本节在 ModelNet40[9] 数据集上进行了实验, 详细分析了网络结构中各个模块的作用, 并且分析了卦限卷积中不同特征融合方式、不同近邻点选择方法和不同特征输入的效果. 此外, 本节还对卦限卷积与其他方法的旋转鲁棒性和计算复杂度进行了比较.

1) 结构的设计. 为了分析卦限卷积模块中各个部件的重要性, 通过将各个部件分别加入卦限卷积模块中进行实验, 结果如表 5 所示. 在卦限卷积模块中, 首先通过堆叠多层 2D 卷积以获取点云丰富的局部特征, 此时该模型可以达到 90.7% 的准确率. 为了充分利用低层的几何特征, 接着将所有卷积层的输出特征级联起来, 并通过一层 MLP 实现多层特征的融合, 此时的准确率可以提升到 91.2%. 考虑到多层堆叠卷积可能带来的过拟合问题, 进一步以残差方式将融合特征与最后一层 2D 卷积层的输出特征相加, 准确率也进一步提升到 91.5%. 最后, 为了能够客观的与 PointNet++[13] 等方法对比, 采用了投票机制, 将输入点云均匀旋转 12 个不同角度并分别送入模型中预测, 并取平均值作为最终的结果, 最终取得 91.9% 的准确率.

2) 特征融合方式的选择. 为了比较 2D 卷积和 MLP 两种特征融合方法对最终结果性能的影响, 本组实验对这两种特征融合的方式进行了对比, 实验结果如表 6 所示. 可以观察到 2D 卷积效果更佳, 这是由于在使用 MLP 时, 其是对每个邻点单独处理, 然后通过最大值操作聚合局部特征, 该操作只保留了每个通道中最重要的信息, 从而导致细节信息的丢失; 而在使用 2D 卷积时, 其会考虑到所有邻点各个通道的信息, 充分利用了细节信息.

3) 近邻点的选择. K 近邻 (K-nearest neighbor, KNN) 是最常见的一种近邻选择方式, 本文提出了使用 8 卦限搜索的方式来选择近邻点. 对此, 本组实验对这两种近邻点的选择进行了对比, 实验结果如表 7 所示. 可以发现, 本文所使用的 8 卦限

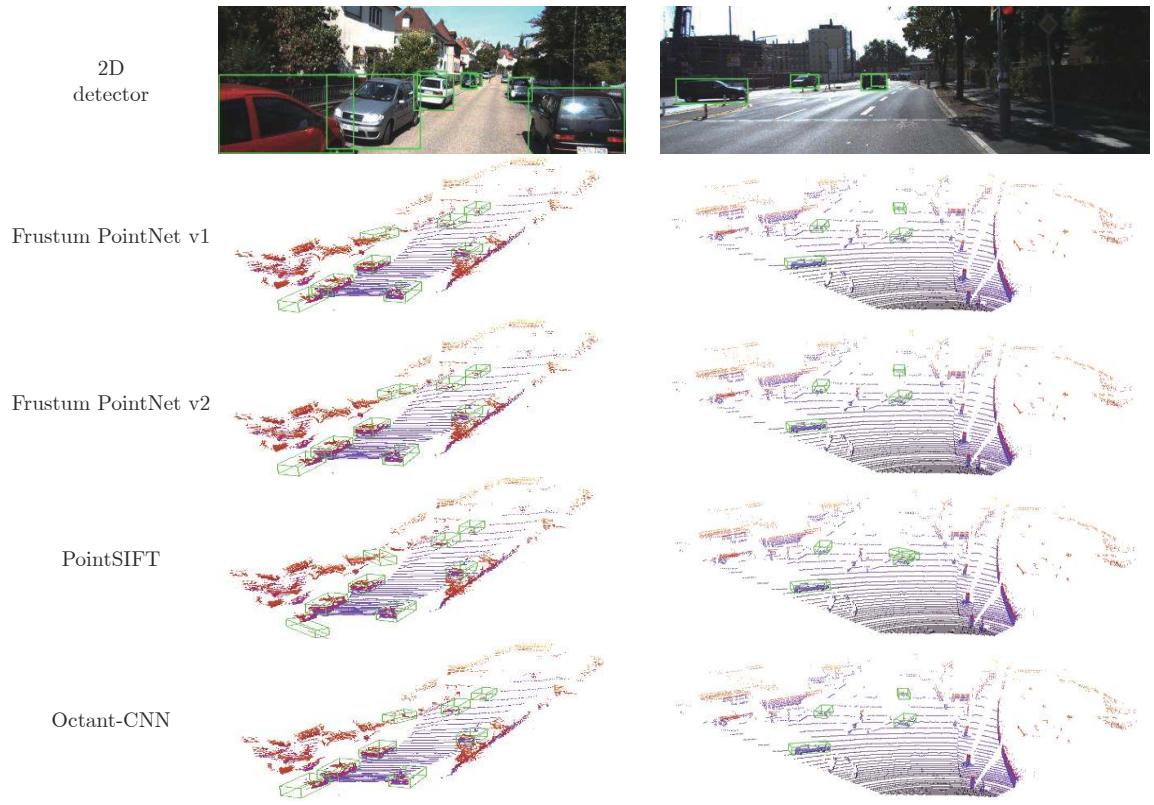


图 5　KITTI 目标检测可视化结果

Fig. 5　Visualization of detection results on KITTI

表 5 结构设计分析
Table 5 Analysis of the structure design

模型	多层融合	残差	投票	oAcc (%)
A				90.7
B	✓			91.2
C	✓	✓		91.5
D	✓	✓	✓	91.9

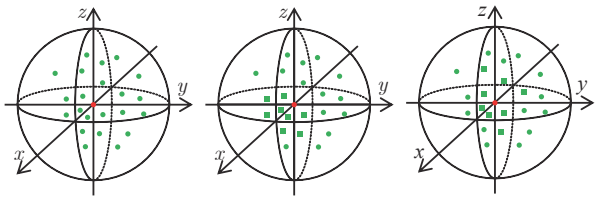
表 6 2D 卷积和 MLP 的对比
Table 6 Comparisons of 2D CNN and MLP

模型	运算	oAcc (%)
A	MLP	90.8
B	2D CNN	91.9

搜索的性能要优于 KNN. 这两种近邻点的区别如图 6 所示, 其中方框表示选择的近邻点. 当使用 KNN 时, 选取的近邻点会受到点云密度特性的影响而偏向某一特定方向; 而使用 8 卦限搜索时, 所选取的近邻点来自于不同的方向, 从而更好地覆盖在点云上.

表 7 不同邻点的比较
Table 7 The results of different neighbor points

模型	邻点	准确率 (%)
A	K近邻	90.2
B	8 卦限搜索	91.9



(a) 点云 (b) K 近邻搜索 (c) 8 卦限搜索
(a) Point cloud (b) KNN search (c) 8 octant search

图 6 K 近邻和 8 卦限搜索的比较
Fig. 6 Comparison of KNN and 8 octant search

4) 搜索半径的设置: 为了分析在使用 8 卦限搜索近邻点时, 搜索半径的限制对模型性能的影响, 通过设置几组不同的搜索半径进行实验. 由于在点云分类任务中, 点云首先被标准化到单位球体中, 因此最大搜索半径设置为 1. 结果如表 8 所示, 我们观察到, 当搜索半径越大, 分类准确率也会随之提升. 这是由于在设置搜索半径时, 部分偏离中心点较远的近邻点会被丢弃, 从而导致几何结构的不完整. 因此, 我们取消了搜索半径的限制.

5) 不同输入特征的比较: 本组实验对比了使用不同的输入特征对模型最终性能的影响, 实验结果如表 9 所示. 从实验结果可以看出来, 当仅使用邻点的特征作为卷积的输入时, 由于缺少点云的空间位置信息, 此时的效果不佳. 随着越来越多的坐标信息, 如中心点的坐标、中心点与邻点的残差坐标同时送入卷积中进行运算, 精度也会得到相应的提升.

表 8 不同搜索半径的比较
Table 8 Comparison of different search radius

模型	搜索半径	oAcc (%)
A	(0.25, 0.5, 1.0)	88.0
B	(0.4, 0.8, 1.0)	89.2
C	(0.5, 1.0, 1.0)	89.9
D	None	91.9

表 9 不同输入通道的结果比较
Table 9 The results of different input channels

模型	输入通道	oAcc (%)
A	(f_{ij})	90.1
B	$(x_i - x_{ij}, f_{ij})$	90.3
C	(x_i, f_{ij})	90.8
D	$(x_i, x_i - x_{ij}, f_{ij})$	91.9

6) 点云旋转的鲁棒性分析: 在本组实验中, 将输入点云分别旋转 0°, 30°, 60°, 90°, 180° 后送入 Octant-CNN 和 PointSIFT[14] 中进行预测, 通过计算由不同角度得到的准确率的均值和方差来比较这两种方法对点云旋转的鲁棒性. 由于 Octant-CNN 在一定程度上依赖于 T-Net, 为了更客观地比较单阶段卷积和三阶段卷积对点云旋转的鲁棒性的影响, 我们还将 T-Net 加入 PointSIFT 模型中, 实验结果如表 10 所示. 可以发现, T-Net 在一定程度上提高了 PointSIFT 的旋转鲁棒性, 但是本文提出的单阶段卷积对点云旋转依然更具鲁棒性, 这是由于三阶段卷积是存在先后顺序的, 对于三维空间不同维度具有各向异性, 而单阶段卷积同等处理各卦限的点, 对输入点云具有各向同性.

7) Octant-CNN 的复杂度: 最后, 我们对比了 Octant-CNN 和其他一些方法在语义分割任务中的参数量和每秒的浮点运算量 (Floating point operations per second, FLOPs), 结果如表 11所示. 可以观察到, 相比于 PointSIFT[14], Octant-CNN 的参数量和 FLOPs 都得到了明显的降低, 这主要来自两个方面: 1) 在下采样阶段, PointSIFT采用可学习的方式聚合局部特征, 这为 PointSIFT引入了额外的可学习参数, 而 Octant-CNN 直接采用最大值池化聚合局部特征, 这一操作不需要额外参数; 2) 由

表 10 点云旋转鲁棒性比较

Table 10 Comparison of robustness to point cloud rotation

方法	0° (%)	30° (%)	60° (%)	90° (%)	180° (%)	均值	方差
PointSIFT[14]	88.2	89.2	88.9	88.7	88.5	88.7	0.124
PointSIFT+T-Net	89.1	89.4	89.4	88.6	88.6	89.04	0.114
Octant-CNN	91.5	91.7	91.9	91.5	91.8	91.68	0.025

表 11 点云语义分割的复杂度

Table 11 Complexity in point cloud semantic segmentation

方法	参数量 (MB)	FLOPs (B)
PointNet[12]	1.17	7.22
PointNet++[13]	0.97	1.96
PointSIFT[14]	13.53	24.32
Octant-CNN	4.31	2.44

于语义分割任务需要上采样以恢复点的原始数量, PointSIFT 首先使用几层 MLP 抽象语义特征, 紧接着使用三阶段卷积进一步丰富语义信息, 这带来了大量的参数, Octant-CNN 则只使用了 MLP 来抽象高层语义特征. 同时可以发现, 对于部件分割和目标检测这两个数据集相对较小的任务, PointSIFT 由于参数量过大, 导致模型出现过拟合的情况, 因此在这两个任务上的效果不佳.

3 结论

为了有效捕获点云的局部几何信息, 本文提出了 Octant-CNN, 并在对象分类、部件分割、语义分割和目标检测上均取得显著提升. Octant-CNN 具有三个关键点: 首先, 在近邻空间中定位最近邻点时, Octant-CNN 取消了搜索范围的限制, 这使得远离中心点的近邻点可以被捕获, 从而更好地反映点云的密度特性. 其次, Octant-CNN 使用单阶段的卷积操作直接提取点的局部几何结构, 这克服了三阶段卷积操作带来的对卦限顺序敏感的问题, 从而对点云旋转更具鲁棒性. 最后, 通过下采样模块实现对原始点集的分组及特征聚合, 从而增大了中间特征的感受野, 并大大降低了卷积操作的计算量.

References

1 Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks. In: Proceedings of the 2012 Advances in Neural Information Processing Systems. Nevada, USA, 2012. 1097−1105

2 He K M, Zhang X Y, Ren S Q, Sun J. Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE, 2016. 770−778

3 Girshick R. Fast R-CNN. In: Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago, Chile: IEEE, 2015. 1440−1448

4 Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: Unified, real-time object detection. In: Proceedings of the 2016 I EEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, USA: IEEE, 2016. 779−788

5 Zhu Z, Xu M D, Bai S, Huang T T, Bai X. Asymmetric non-local neural networks for semantic segmentation. In: Proceedings of the 2019 IEEE International Conference on Computer Vision. Seoul, South Korea: IEEE, 2019. 593−602

6 Li Y, Qi H Z, Dai J F, Ji X Y, Wei Y C. Fully convolutional instance-aware semantic segmentation. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA: IEEE, 2017. 2359−2367

7 Peng Xiu-Ping, Tong Qi-Sheng, Lin Hong-Bin, Feng Chao, Zheng Wu. A deep residual-feature pyramid network for scattered point cloud semantic segmentation. *Acta Automatica Sinica*, 2019. DOI: 10.16383/j.aas.c190063
(彭秀平, 仝其胜, 林洪彬, 冯超, 郑武. 一种面向散乱点云语义分割的深度残差−特征金字塔网络框架. 自动化学报, 2019. DOI: 10.16383/j.ass.c190063)

8 Maturana D, Scherer S. Voxnet: A 3d convolutional neural network for real-time object recognition. In: Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems. Hamburg, Germany: IEEE, 2015. 922−928

9 Wu Z R, Song S R, Khosla A, Yu F, Zhang L G, Tang X O, Xiao J X. 3d shapenets: A deep representation for volumetric shapes. In: Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston, USA: IEEE, 2015. 1912−1920

10 Su H, Maji S, Kalogerakis E, Learned-Miller E. Multi-view convolutional neural networks for 3d shape recognition. In: Proceedings of the 2015 IEEE International Conference on Computer Vision. Santiago, Chile: IEEE, 2015. 945−953

11 Yang Z, Wang L W. Learning relationships for multi-view 3d object recognition. In: Proceedings of the 2019 IEEE International Conference on Computer Vision. Seoul, Korea (South): IEEE, 2019. 7505−7514

12 Qi C R, Su H, Mo K, Guibas L J. Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA: IEEE, 2017. 652−660

13 Qi C R, Yi L, Su H, Guibas L J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In: Proceedings of the 2017 Advances in Neural Information Processing Systems. Long Beach, USA, 2017. 5099−5108

14 Jiang M Y, Wu Y R, Zhao T Q, Zhao Z L, Lu C W. Pointsift: A sift-like network module for 3D point cloud semantic segmentation [Online], available: https://arxiv.org/abs/1807.00652, July 22, 2020

15 Rao Y M, Lu J W, Zhou J. Spherical fractal convolutional neural networks for point cloud recognition. In: Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA, 2019. 452−460

16 Liu Y C, Fan B, Xiang S M, Pan C H. Relation-shape convolutional neural network for point cloud analysis. In: Proceedings of

the 2019 IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA, 2019. 8895−8904

17 Boulch A. Convpoint: Continuous convolutions for point cloud processing. *Computers and Graphics*, 2020, **88**: 24−34

18 Simonovsky M, Komodakis N. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Hawaii, USA: IEEE, 2017. 3693−3702

19 Te G, Hu W, Zheng A, Guo Z M. RGCNN: Regularized graph cnn for point cloud segmentation. In: Proceedings of the 26th ACM International Conference on Multimedia. Seoul, Korea (South): ACM, 2018. 746−754

20 Wang Y, Sun Y B, Liu Z W, Sarma S E, Bronstein M M, Solomon J M. Dynamic graph CNN for learning on point clouds. *ACM Transactions on Graphics (TOG)*, 2019, **38**(5): 1−12

21 Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 2014, **15**(1): 1929−1958

22 Yang J C, Zhang Q, Ni B, B, Li L G, Liu J X, Zhou M D, Tian Q. Modeling point clouds with self-attention and gumbel subset sampling. In: Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA, 2019. 3323−3332

23 Xie S N, Liu S N, Chen Z Y, Tu Z W. Attentional shapecontextnet for point cloud recognition. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 4606−4615

24 Duan Y Q, Zheng Y, Lu J W, Zhou J, Tian Q. Structural relational reasoning of point clouds. In: Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA, USA: IEEE, 2019. 949−958

25 Lin H X, Xiao Z L, Tan Y, Chao H Y, Ding S Y. Justlookup: one millisecond deep feature extraction for point clouds by lookup tables. In: Proceedings of the 2019 IEEE International Conference on Multimedia and Expo. Shanghai, China: IEEE, 2019. 326−331

26 Klokov R, Lempitsky V. Escape from cells: Deep KD-networks for the recognition of 3d point cloud models. In: Proceedings of the 2017 IEEE International Conference on Computer Vision. Venice, Italy: IEEE, 2017. 863−872

27 Li J X, Chen B M, Hee L G. So-net: Self-organizing network for point cloud analysis. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 9397−9406

28 Yi L, Kim V G, Ceylan D, Shen I C, Yan M Y, Su H, et al. A scalable active framework for region annotation in 3D shape collections. *ACM Transactions on Graphics (ToG)*, 2016, **35**(6): 1−12

29 Huang Q G, Wang W Y, Neumann U. Recurrent slice networks for 3d segmentation of point clouds. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 2626−2635

30 Armeni I, Sener O, Zamir A R, Jiang H, Brilakis I, Fischer M, Savarese S. 3d semantic parsing of large-scale indoor spaces. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, USA: IEEE, 2016. 1534−1543

31 Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving. In: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Rhode Island, USA: IEEE, 2012. 3354−3361

32 Qi C R, Liu W, Wu C, X Su H, Guibas L J. Frustum pointnets for 3d object detection from RGB-D data. In: Proceedings of the 2018 IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, USA: IEEE, 2018. 918−927

33 Lan S Y, Yu R C, Yu G, Davis L S. Modeling local geometric structure of 3d point clouds using GEO-CNN. In: Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, USA: IEEE, 2019. 998−1008

许 翔 南京信息工程大学自动化学院硕士研究生. 2018 年获得南京信息工程大学信息与控制学院学士学位. 主要研究方向为三维点云场景感知.
E-mail: xuxiang0103@gmail.com
(**XU Xiang** Master student at the School of Automation, Nanjing University of Information Science and Technology. He received his bachelor degree from the School of Information and Control, Nanjing University of Information Science and Technology in 2018. His research interest covers 3D point cloud scene perception.)

帅 惠 南京信息工程大学博士研究生. 2018 年获得南京信息工程大学信息与控制学院硕士学位. 主要研究方向为目标检测, 3D 点云场景感知.
E-mail: huishuai13@163.com
(**SHUAI Hui** Ph.D. candidate at Nanjing University of Information Science and Technology. He received his master degree from the School of Information and Control, Nanjing University of Information Science and Technology in 2018. His research interest covers object detection and 3D point cloud scene perception.)

刘青山 南京信息工程大学计算机学院、软件学院、网络空间安全学院院长, 教授. 2003 年获得中国科学院自动化研究所博士学位. 主要研究方向为图像理解, 模式识别, 机器学习. 本文通信作者.
E-mail: qsliu@nuist.edu.cn
(**LIU Qing-Shan** Dean and professor of the School of Computer Science, Nanjing University of Information Science and Technology. He received his Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences in 2003. His research interest covers image understanding, pattern recognition and machine learning. Corresponding author of this paper.)