

基于 PointNet++ 的逆密度点云识别与分割算法

周江伟¹, 邵洁¹, 曹盛²

(1. 上海电力大学 电子与信息工程学院, 上海 200000;

2. 中国电力工程顾问集团 华东电力设计院有限公司, 上海 200001))

摘要: 为了提高点云处理精度, 针对 PointNet++ 对不均匀分布的点云数据特征提取不完整以及忽略了部分点云特征导致分类与分割结果不佳等问题, 对算法 PointNet++ 进行了研究, 提出了基于 PointNet++ 的融合密度信息的逆密度点云识别与分割算法 D-PointNet++; 利用点云密度计算出每个点的采样概率, 根据采样概率使用多项分布进行点云采样; 通过自适应缩放分组半径进行点云分组; 采用多种池化方法混合提取点云特征并利用多头注意力机制计算出多种特征的权重, 并加权聚合得到点云的全局特征; 实验结果表明, 相较于多种参评算法, D-PointNet++ 在点云分类准确率、分割精度上均有显著提升。

关键词: 三维点云; 点云分类; 深度学习; 注意力机制; 点云分割; 激光点云

Recognition and Segmentation Algorithm for Inverse Density Point Cloud Based on PointNet++

ZHOU Jiangwei¹, SHAO Jie¹, CAO Sheng²

(1. School of Electronic and Information Engineering, Shanghai University of Electric Power,

Shanghai 200000, China; 2. East China Electric Power Design Institute Co., Ltd.,

China Power Engineering Consulting Group, Shanghai 200001, China)

Abstract: In order to improve the processing accuracy of point cloud, and to solve the incomplete feature extraction of non-uniformly distributed point cloud data and poor classification and segmentation results caused by ignoring some point cloud features, a study of the PointNet++ algorithm was made. Based on PointNet++ fusion density information, an inverse density point cloud recognition and segmentation algorithm D-PointNet++ was proposed. A point cloud density was used to calculate the sampling probability of each point, and a polynomial distribution was used to sample the point cloud according to the sampling probability. The point cloud was grouped by adaptively scaling the grouping radius. Multi-pooling methods were used to extract point cloud features, and the weights of multiple features were calculated by using multi-head attention mechanism, and the global features of the point cloud were obtained by using weighted aggregation. Experimental results show that, compared with multi-evaluation algorithms, the D-PointNet++ has a significant improvement in point cloud classification accuracy and segmentation precision.

Keywords: 3D point cloud; point cloud classification; deep learning; attention mechanism; point cloud segmentation; laser point cloud

收稿日期:2024-07-28; 修回日期:2024-09-14。

基金项目:国家电网有限公司科技项目(J202301)。

作者简介:周江伟(1997-),男,硕士研究生。

邵洁(1981-),女,博士,教授。

引用格式:周江伟,邵洁,曹盛. 基于 PointNet++ 的逆密度点云识别与分割算法[J]. 计算机测量与控制, 2025, 33(9): 334-341.

0 引言

随着三维扫描技术^[1]、激光雷达、摄影测量等技术的发展,点云数据处理在许多领域中具有重要的应用价值和前景。例如三维建模^[2]、地理信息系统、目标检测与识别、虚拟现实与增强现实^[3]、医学图像处理^[4]等。在应用范围越来越广的同时,对点云数据处理的精度和智能化^[5]程度要求也越来越高。随着深度学习的发展和激光点云处理技术的进步,目前智能建模^[6]方式中应用最广泛的是激光点云建模^[7]。激光点云建模是一种基于激光扫描技术创建模型的方法,其中三维点云的分类和语义分割是建模中非常重要的一个环节。在点云分类与语义分割领域,利用深度学习提取点云特征主要分为基于多视图的方法、基于体素化网格的方法和基于原始点云数据的方法。

基于多视图的方法:文献[8]提出多视图卷积神经网络 MVCNN,通过多个视角观测目标或场景,将信息通过卷积神经网络进行融合。提取更丰富的特征表达。文献[9]提出基于图卷积网络的多视图学习方法 View-GCN,通过建立一个图来表示多视图数据之间的关系,利用图卷积网络将不同视图的信息进行融合,提取更综合的特征表示,提高了模型的鲁棒性。然而,因需要在图结构上进行卷积运算,导致计算复杂度较高。基于体素化网格的方法:文献[10]提出神经网络模型 VoxNet,将复杂的点云数据转换为固定大小的体素网格表示。以离散的方式表示三维数据,提高复杂几何的表示能力。文献[11]提出将点云数据映射到规则三维网格中的方法 PointGrid,能够提供相对均匀的采样和规整的表示。从而减少点云数据的变化和噪声,提高处理效果。但其处理不规则分布或具有大范围变化的点云数据时会引入一些采样偏差;基于原始点云数据的方法。文献[12]提出直接处理点云数据的模型 PointNet,学习点云的空间编码,接着聚合所有点的特征得到全局的点云特征。其提取的全局特征能够很好地完成

分类任务,但局部特征提取能力较差。为了解决这个问题,文献[13]提出了 PointNet++,其核心是分层次的点集抽象层,能够有效提取局部特征和全局特征。文献[14]提出深度学习模型 DGCNN,将点云数据表示为一个动态图,并利用图卷积神经网络^[15]对点云数据进行特征提取。DGCNN 通过动态图灵活地处理不同形状和尺度的点云数据,具有一定的旋转和平移不变性。但是处理噪声较大及不完整点云数据时,性能有所下降。为了处理大规模点云数据,文献[16]提出 RandLA-Net,通过随机抽样高效地处理大规模点云数据,利用多层感知机进行特征提取和聚合,提高了算法的性能。但是随机选择局部区域可能导致一些重要的信息被忽略,对于不均匀点云存在局限性。

可以看出点云的处理速度和处理精度之间存在一定的权衡关系。在点云处理任务中,常常需要在较短的时间内处理大量的点云数据,同时又要保证处理结果的准确性。

因此本文针对 PointNet++ 中最远点采样计算复杂度高,对初始点的选择敏感,会忽略一些点云稀疏部分的信息等问题,以及点云分组对在点云数量不够情况下重复采取点云信息等问题,提出了一种复杂度更低的融合密度信息的点云抽样分组方法。同时融合多种池化方法提取点云特征,最后利用多头注意力机制计算加权聚合得到最后的全局特征,从而提高了算法对点云的分类准确率和分割精度。

1 D-PointNet++

1.1 算法结构

如图 1 所示,本文提出了一种用于三维点云数据识别分类和语义分割的算法结构。将点云数据输入到算法中,首先经过本文提出的一种计算复杂度更低的融合密度信息的点云抽样分组聚合方法进行点云采样分组,提取并聚合每一个簇的特征。经过多次分组聚合,输入点云数据的点的数量减少,每个点所携带的特征维度增

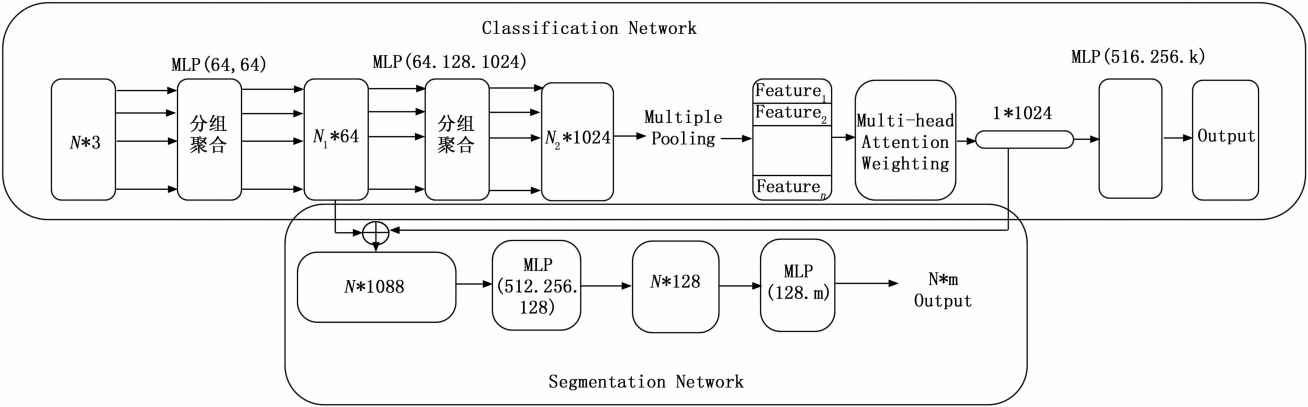


图 1 算法结构

加,接着对这些特征进行多种池化方法并进行多头注意力加权特征聚合,获得点云数据的高维全局特征。在点云分类任务中,接下来利用多层感知机进行多次特征转换最终输出分类结果。在点云分割任务中,则需要进行上采样,复制并拼接全局特征和局部特征,重复对特征的变换和拼接,最终得到每个点的最终特征,并输出每个点的语义类别的分数,从而实现点对云数据的语义分割。

1.2 融合密度信息的点云采样分组方法

常用的三维点云数据的采样方式有随机采样、均匀采样、体素网格采样和最远点采样。随机采样效率高,适用于大规模点云,但是采样点分布随机性高,影响后续特征提取。均匀采样的采样点分布相对均匀,但点云分布不是正态分布,点云越密集,经过分组后所提取出来的特征信息量越低,因此,在点云分布较复杂或不规则的情况下,所提取特征也相对受限。体素网格采样将点云划分为体素,实现对点云的稀疏化采样,但会丢失点云表面的细节。最远点采样通过选择与已采样点最远的点,实现较好的点云覆盖效果。但是需要迭代计算,计算复杂度会随着采样点数量的增加而增加。

目前常用的点云分组方法有球查询和 K 近邻查询。球查询先选择采样点作为球心,再搜索指定半径范围内的邻近点。如果邻近点数量比目标点云数量多,则保留离中心点距离更近的点,反之则复制范围内的点补齐数据。 K 近邻查询则是用于找到每个点在给定数据集中的 K 个最接近邻居点。先建立采样点数据列表,再计算其与数据集中点的距离并排序,以确定最近的 K 个邻居点。但是需要计算每个查询点与数据集中所有点的距离,因此随着数据规模的增加,计算复杂度会显著提高。而球查询在点云数量不够情况下重复采集点云信息造成特征重复等问题。

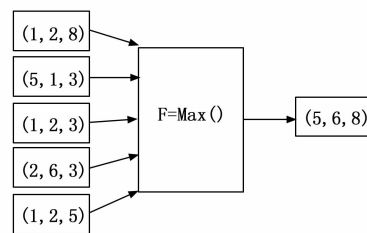
因此本文提出了融合密度信息的点云抽样方法与自适应点云分组方法,可以在较好的点云覆盖效果的情况下,降低计算的复杂度,增加抽样过程的鲁棒性。

融合密度信息的点云抽样方法利用点云数据中单个点与所有点的欧氏距离来判断点云密度,避免了最远点采样中的迭代计算,从而提高了抽样效率,同时也能很好的得到点云的密度信息。创建一个布尔掩码张量,其中大于零的元素为 True,小于等于零的元素为 False。并过滤出有效的密度值。通过点云的密度计算出每个点的采样概率,根据采样概率使用多项分布进行采样。

自适应点云分组方法是基于球查询的点云分组方法,一旦发现半径范围内的点云数量不够,则采用扩大点云半径并判断新加入的点云与中心点的距离,选择距离更优的点加入特征聚合,提高了分组特征的多样性且不会增加大量的计算过程。

1.3 注意力机制优化特征聚合方法

PointNet++使用的最大池化方法是对称操作,能够保留最显著的特征,所以对输入的平移、旋转等变换具有一定程度的不变性。最大池化方法结构如图 2 所示。



但由于这种方法选取最大特征作为池化结果,而丢弃其他特征,会导致部分细节信息的丢失。根据文献[17]中的实验可知使用非最大特征值来训练,也能得到不错的点云分类准确率和点云语义分割精度。受其启发,本文对于被最大池化丢弃的特征进行回收利用,来增加特征的利用率。

注意力机制^[18]是一种用于加权关注输入序列中不同信息的方法,对于给定的查询信息(Query)和一组键值对(Key-Value),通过计算查询与每个键之间的相似度,得到一组权重,用于加权求和,从而使算法选择性地关注更重要的信息。计算公式如下:

$$attentionweight(Q, K) = softmax\left(\frac{QK^T}{\sqrt{d}}\right) \quad (1)$$

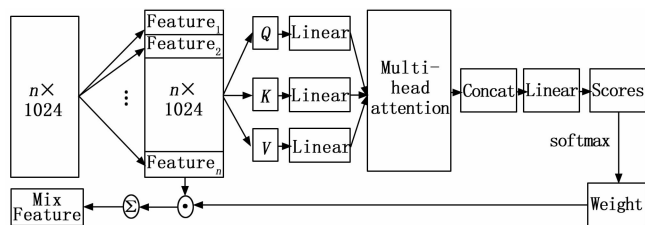
$$attention(Q, K, V) = attentionweight(Q, K) * V \quad (2)$$

Q 为 Query, K 为 Key, d 是向量的维度,作为缩放因子。缩放后的结果通过激活函数,将其转化为注意力权重,确保权重的总和为 1。

多头自注意力机制^[19]是注意力机制的变体,结构如图 3 所示。其通过多个注意力头的并行提升算法的性能。每个注意力头会学习不同的查询、键和值矩阵,并得到一个独立的自注意力输出。再将输出拼接,经过线性变换得到最终输出。公式如下:

$$head_i = attention(QW_i^q, KW_i^k, VW_i^v) \quad (3)$$

$$multihead(Q, K, V) = concat(head_1, \dots, head_h)W \quad (4)$$



注意力机制在点云处理中能够根据点的重要性和相关性分配不同的权重, 选择性地聚焦于输入中的相关部分, 有助于减少冗余信息和噪声的影响, 提高算法处理点云数据的性能。

因此为了弥补最大池化提取全局特征时忽略了很多细节特征等问题, 本文提出了注意力加权的特征聚合方法。为了兼顾点云的显著特征与全局特征的提取与聚合, 采用多次循环最大池化与平均池化相结合的方法, 提取多个高维特征, 对这些特征进行多头自注意力加权聚合, 将多个自注意力输出按通道维度进行拼接, 得到最终算法需要含有更更富点云模型信息的特征表达。

这种方法关注了输入点云的不同特征, 且可以并行计算每个头的注意力权重和输出, 从而可以表示比简单的加权平均更复杂的表示函数。提高了算法的特征提取能力。多头机制允许算法同时关注输入的不同方面, 并从多个角度捕捉输入点云数据的特征, 提高了特征的表达能力。

2 实验与结果分析

2.1 实验环境

本文所有的实验的硬件环境和配置如表 1 所示。

表 1 实验的硬件环境和配置

CPU	12th Gen Intel Core i7-12650H
GPU	8 GB (NVIDIA GeForce RTX 4060)
内存	16 GB (DDR4 3200MHz 8GB x 2)
操作系统	Windows 11
Pytorch	Pytorch 1.12.1
编译器	PyCharm Community Edition
Python	Python 3.8.16
CUDA	Cuda 11.1

将实验所需的硬件和软件环境搭建好后, 在编译器中将 D-PointNet++ 按照图 1 所示结构实现, 最后搭建好整个算法模型。

2.2 训练参数

训练参数对网络性能有着重要的影响。合适的训练参数设置可以帮助网络更好地学习数据的特征, 提高性能和收敛速度。

学习率控制了参数更新的步幅大小, 较大的学习率可能导致训练过程不稳定, 算法无法收敛, 而过小的学习率则可能导致训练过程缓慢。批次大小决定了每次参数更新时使用的样本数量。较大的批次大小可以提高训练速度, 但可能导致内存占用较大。合理选择批次大小需要考虑算法规模和硬件资源等因素。不同的优化器具有不同的优化策略和调整方式, 对网络性能也有影响。常见的优化器包括 Adam、SGD 等。选择合适的优化器有助于加速训练和改善算法性能。

本文综合考虑多个因素, 经过多次实验, 最终选择的训练参数如表 2 所示。

表 2 训练参数

训练参数	点云分类	点云分割
batch size	16	16
epoch	180	180
learning rate	0.001	0.001
optimizer	Adam	Adam
decay rate	1e-4	1e-4
sampling number	1 024	2 048
step	20	20
decay rate	0.5	0.5

每个批次中数据的数量为 16, 训练轮数为 180, 初始学习率设为 0.001, 学习率衰减步长为每 20 轮进行一次衰减, 衰减率设认为 0.5。而点云分割和点云分类是两种不同的任务, 因此它们对于输入点云的要求也不同。在点云分类任务中, 我们的目标是将整个点云归为特定的类别, 所以更注重点云的整体特征表示, 而较少的点数可以减少计算和内存开销, 同时保留主要的全局特征。综合考虑选择 1 024 个点作为点云分类任务的输入。

点云分割则需要更多的点来描述局部信息, 通常需要更全面地捕捉点云的局部几何结构和特征。通过增加输入点的数量, 可以提供更多的细节信息, 有助于更准确地进行分割。所以综合考虑选择 2 048 个点作为点云分割任务的输入。

2.3 ModelNet40 点云分类实验

ModelNet40 数据集是由斯坦福大学的 3D Vision Group 发布, 包含了 40 个物体类别, 涵盖了各种日常生活中常见的物体, 如椅子、桌子、汽车、键盘等。ModelNet40 数据集的标注有严格的规定和原则, 保证了其数据质量和实用性。为点云分类问题的研究提供了一个统一的标准和基准。

为了验证 D-PointNet++ 的有效性, 本文选取 PointNet^[12]、PointNet++^[13]、文献 [20]、AGConv^[21]、SAKS^[22]、PointMLP^[23]、Point-BERT^[24]、文献 [25] 作为参评对比算法。

本文在 ModelNet40 点云分类实验中使用了测试实例准确率 (Test Instance Accuracy) 和类别准确率 (Class Accuracy) 作为算法分类效果的评价指标。将预测结果与真实标签进行比较, 如果相同, 则认为预测正确。反之认为预测错误。最后将所有预测正确的样本数目除以总样本数, 得到测试实例准确率, 计算公式为:

$$Test\ Instance\ Accuracy = \frac{N_{correctly\ predicted}}{N_{total}} \tag{5}$$

其中: $N_{correctly\ predicted}$ 是预测正确的样本数目, N_{total} 是

总样本数。

类别准确率是先将测试数据集中的样本按照类别进行分类,统计算法在每个类别上的预测结果。接着将每个类别中预测正确的样本数目除以该类别的总样本数,得到该类别的准确率。最后将所有类别的准确率求平均,得到类别准确率,其计算公式为:

$$Class\ Accuracy = \frac{\sum_{k=1}^{class\ number} \frac{N_{correctly\ predicted\ k}}{N_{total\ k}}}{Class\ number} \quad (6)$$

其中: $Class\ number$ 是分类类别数目, $N_{total\ k}$ 是每个类的总样本数, $N_{correctly\ predicted\ k}$ 是每个类别预测正确的样本数目。

这两个准确率指标都是通过对算法在测试数据集上的预测结果和真实标签进行比较来衡量算法的分类性能。测试实例准确率关注每个样本的分类结果是否正确,而类别准确率关注整个类别的分类结果是否正确。两者相结合,可以准确衡量算法的性能。

D-PointNet++ 在 ModelNet40 点云分类任务中获得的结果如表 3 所示。

表 3 ModelNet40 数据集点云分类结果

算法	Test Instance Accuracy/%	Class Accuracy/%
PointNet	88.2	86.0
PointNet++	89.1	—
文献[20]	87.2	—
AGConv	93.4	90.7
SAKS	93.8	91.0
PointMLP	94.1	91.3
Point-BERT	93.2	—
文献[25]	93.6	91.6
D-PointNet++ 1	92.0	91.0
D-PointNet++ 2	93.5	91.7

D-PointNet++ 1 是只采用融合密度信息的点云采样分组方法, D-PointNet++ 2 是在 1 的基础上采用注意力机制优化特征聚合方法的算法。

由表 3 可知 D-PointNet++ 在测试实例准确率和类别准确率上相较于各参评算法均有显著提升。这表明融合密度信息的点云采样分组方法有效的提取了点云模型的密度信息,且分组策略更加合理,提高了特征对点云模型的表达能力。同时,注意力机制优化特征聚合方法利用次要特征与平均池化特征并利用多头自注意力加权池化处理,使算法能够更准确地捕捉点云中的关键特征,并有效提取点云的重要信息,从而提高了算法在点云分类任务的准确性。

2.4 ShapeNet 点云分割实验

ShapeNet 是一个大规模的三维形状数据库,该数据集由斯坦福大学计算机图形实验室创建和维护。其包

含了 16 类物体, 50 个物体部件类别, 并提供了丰富的标注信息, 以支持各种形状相关的任务。

为了验证 D-PointNet++ 的有效性, 本文选取 PointNet^[12]、PointNet++^[13]、文献 [20]、MH-PAEC^[26]、Point-BERT^[24] 作为参评对比算法。

本文在 ShapeNet 点云分割任务中使用准确率 (Accuracy)、类平均准确率 (Class avg accuracy)、类平均 IOU (Class avg mIOU) 和实例平均 IOU (Instance avg mIOU) 作为算法的评价指标。

$Accuracy$ 表示正确预测的部件点数与总点数之间的比例。准确率越高, 表示算法的分割性能越好。公式如下:

$$Accuracy = \frac{N_{correct}}{N_{total}} \quad (7)$$

其中: $N_{correct}$ 是预测正确的样本数目, N_{total} 是总样本数。

$Class\ avg\ accuracy$ 用来衡量算法在各个部件类别上的整体预测性能。其计算每个部件类别的准确率, 并取所有类别准确率的平均值。类平均准确率用于评估算法对不同部件类别的预测准确度。其公式如下:

$$Class\ avg\ accuracy = \frac{\sum_{k=1}^{Class\ number} \frac{N_{correct\ k}}{N_{total\ k}}}{Class\ number} \quad (8)$$

其中: $Class\ number$ 是分类类别数目, $N_{total\ k}$ 是每个类的总样本数, $N_{correct\ k}$ 是每个类别预测正确的样本数目。

$Class\ avg\ mIOU$ 用于衡量分割任务中预测区域与真实区域重叠程度。在点云部件分割中, 用于度量算法对每个部件类别的分割准确度。其计算每个类别的部件分割区域与真实标签的重叠程度, 再取所有类别的 IOU 平均值。类平均 IOU 越高, 表示算法的分割精度越好。其公式如下:

$$Class\ avg\ mIOU = \frac{\sum_{i=0}^C \frac{TP_i}{TP_i + FP_i + FN_i}}{C} \quad (9)$$

其中: C 表示类别的总数, TP_i 表示预测为类别 i 且真实标签也为类别 i 的样本数, FP_i 表示预测为类别 i 但真实标签不为类别 i 的样本数, FN_i 表示预测不为类别 i 但真实标签为类别 i 的样本数。

$Instance\ avg\ mIOU$ 衡量每个真实部件实例的分割准确度, 其计算每个真实部件实例的分割区域与预测结果的重叠程度, 然后取所有实例的 IOU 平均值。公式如下:

$$Instance\ avg\ mIOU = \frac{\sum_{j=1}^k \sum_{i=1}^k \frac{TP_{ij}}{TP_{ij} + FP_{ij} + FN_{ij}}}{N} \quad (10)$$

其中: N 表示所有实例的总数, K 表示实例所属的类别的总数, TP_{ij} 表示预测为第 j 个实例所属的类别 i 且真实标签也为该类别的样本数, FP_{ij} 表示预测为第 j 个实例所属的类别 i 但真实标签不为该类别的样本数, FN_{ij} 表示预测不为第 j 个实例所属的类别 i 但真实标签为该类别的样本数。

这些指标共同提供了评估点云部件分割任务性能的重要指标。

D-PointNet++ 在 ShapeNet 点云分割任务中获得的结果如表 4 所示。

表 4 ShapeNet 数据集点云分割结果

算法	评价指标			
	Accuracy	Class avg accuracy	Class avg mIOU	Instance avg mIOU
PointNet	89.3	82.4	80.4	81.2
PointNet++	92.4	85.6	81.8	84.3
文献[20]	93.3	—	74.0	—
MHPAEC	94.1	—	83.2	—
Point-BERT	93.9	—	84.0	—
D-PointNet++ 1	94.2	86.9	82.7	85.3
D-PointNet++ 2	94.3	87.3	83.3	85.4

根据表 4 可知 D-PointNet++ 在点云部件分割任务中相较于参评算法在 4 种点云分割评价指标上均有所提升。

本文方法在 ShapeNet 数据集上各个类别的点云分割结果如表 5 所示。

通过表 5 可知 D-PointNet++ 的 eval mIoU 在电脑、摩托、浴缸、滑板、桌子等类别的模型点云分割中取得显著提升。由此可见本文的融合密度信息的点云采样分组方法能够有效结合点云分布的疏密特征, 且不会重复采集点云特征, 同时利用注意力机制优化特征聚合方法保留了更多细化特征, 能够有效的学习到边缘点和样本比例较小的点云特征, 进而提升了算法在在点云部件分割的整体性能。

2.5 算法鲁棒性分析

算法在训练过程中的收敛速度是深度学习中用来衡量神经网络性能的一个非常重要的指标。本文采用 ModelNet40 数据集点云分类任务和 ShapeNet 数据集

云分割任务的训练过程来对 PointNet++ 和 D-PointNet++ 1 和 D-PointNet++ 2 做对比分析。

观察这 3 种算法在 ShapeNet 数据集点云分割任务训练过程中 Test Accuracy、Class Avg mIOU、Instance Avg mIOU 的变化过程。结果如图 4~6 所示。

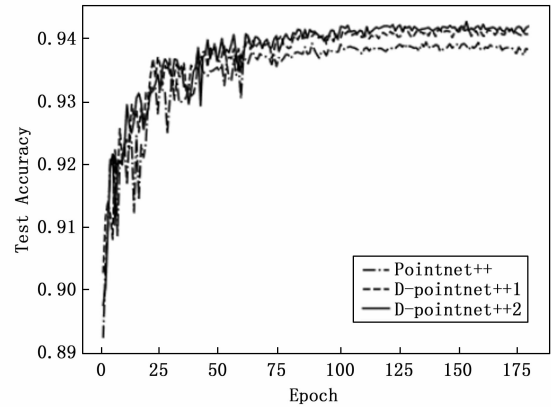


图 4 测试准确率变化曲线

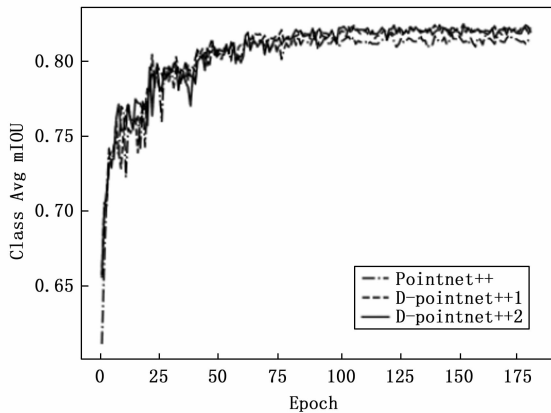


图 5 类平均 mIOU 变化曲线

在 ModelNet40 数据集点云分类任务训练过程中 Test Instance Accuracy、Class Accuracy 的变化过程如图 7~8 所示。

通过图 4~8 可知 D-PointNet++ 在 ModelNet40 数据集点云分类任务中, 训练批次 1 到 75 之间收敛速度要快于 PointNet++，在批次 76 到 180 之间的波动范围

表 5 ShapeNet 数据集各类别点云分割结果

算法	评价指标 eval mIoU															
	飞机	背包	帽子	汽车	椅子	耳机	吉他	小刀	灯	电脑	摩托	浴缸	手枪	火箭	滑板	桌子
PointNet	80.6	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
PointNet++	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	82.4
文献[20]	76.1	78.4	51.4	71.5	86.7	66.2	86.5	80.4	76.7	94.3	61.8	85.9	76.4	40.4	66.7	80.9
MHPAEC	83.6	86.7	82.8	79.9	91.3	76.6	91.2	86.5	85.8	94.7	75.7	94.3	82.9	63.9	73.6	82.4
Point-BERT	84.3	84.8	88.0	79.8	91.0	81.7	91.6	87.9	85.2	95.6	75.6	94.7	84.3	63.4	76.3	81.5
D-PointNet++ 1	82.6	81.3	88.3	79.9	90.5	75.4	91.0	86.7	84.0	95.3	71.0	94.8	82.2	62.1	75.8	82.4
D-PointNet++ 2	83.6	83.3	88.0	78.7	90.7	76.1	91.3	87.1	84.6	95.7	72.4	95.0	83.1	63.5	76.7	83.1

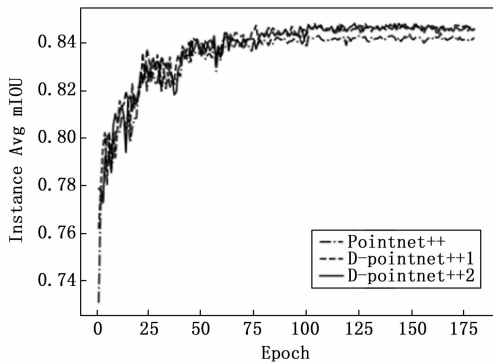


图6 实例平均 mIOU 变化曲线

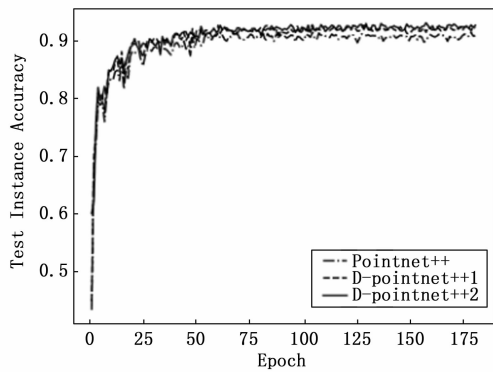


图7 实例测试准确率变化曲线

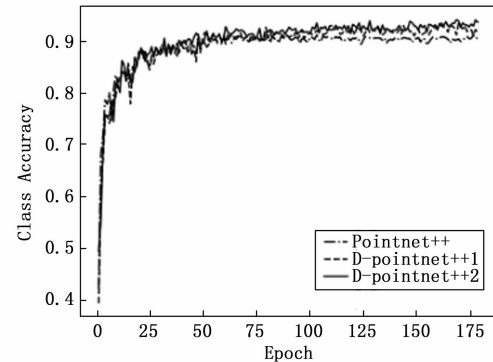


图8 类别准确率变化曲线

相比 PointNet++ 也要更小。

在 ShapeNet 数据集点云分割任务中, D-PointNet++ 在训练批次 1 到 50 之间的收敛速度也比 PointNet++ 更快, 并且在批次 51 到 180 之间的波动范围相比 PointNet++ 更小。这充分说明了 D-PointNet++ 不仅在训练初期的收敛速度更快, 而且在整个训练过程中的表现也更为稳定。因此该算法在点云识别和分割任务中具有较好的鲁棒性。

3 结束语

为解决 PointNet++ 在不均匀分布点云数据上局部特征提取不完整, 忽略了部分点云的特征信息导致点云

分类、分割精度下降等问题, 结合 PointNet++ 提出了融合密度信息的点云采样分组方法与注意力机制优化特征聚合方法。首先结合点云的密度信息并自适应缩放分组半径进行点云分组聚合, 接着采用多种池化方法提取点云特征, 最后利用多头自注意力机制加权聚合得到最后的全局特征用于点云的分类识别与分割任务。通过多种数据集实验验证了 D-PointNet++ 相较于各参评算法在多种评价指标上均有显著提升, 且分析了算法的训练过程, 证明了算法有着较好的性能和鲁棒性。近几年数字孪生和智能驾驶的发展在三维点云处理的实时性上有着越来越高的要求, 未来阶段, 还需在不影响点云处理质量的情况下大幅度提高三维点云的处理速度, 提高算法的泛化性和实时性。

参考文献:

- [1] TSAI R Y. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses [J]. IEEE Journal on Robotics & Automation, 2003, 3 (4): 323 - 344.
- [2] 郭福钟, 郑博文, 祁生文, 等. 三维地质建模技术与方法综述 [J]. 工程地质学报, 2024, 32 (3): 1143 - 1153.
- [3] 郭中远, 徐 锋, 王贵洋, 等. 基于虚拟现实的移动机器人真实环境三维建模系统 [J]. 激光与光电子学进展, 2023, 60 (2): 450 - 456.
- [4] 沈春梅, 刘 凡, 朱佳乐. 基于结构光和 CT 的背部点云配准算法 [J]. 激光与光电子学进展, 2023, 60 (22): 121 - 130.
- [5] 马庆禄, 孙 泉, 黄筱潇, 等. 基于 Transformer 的道路场景点云分类与分割方法 [J]. 激光与红外, 2024, 54 (1): 17 - 23.
- [6] 李育臣, 张之江, 曾 丹, 等. 基于毫米波雷达稀疏点云的人体行为识别方法 [J]. 计算机测量与控制, 2024, 32 (2): 198 - 205.
- [7] 曾政祥, 何晶晶, 张 婧. 融合激光点云的城市级高精度建模技术 [J]. 测绘通报, 2023 (3): 133 - 138.
- [8] SU H, MAJI S, KALOGERAKIS E, et al. Multi-view convolutional neural networks for 3d shape recognition [C] // Proceedings of the IEEE International Conference on Computer Vision, 2015: 945 - 953.
- [9] WEI X, YU R, SUN J. View-GCN: View-based graph convolutional network for 3D shape analysis [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 1850 - 1859.
- [10] ALLEN M, GIROD L, NEWTON R, et al. Voxnet: An interactive, rapidly-deployable acoustic monitoring platform [C] // 2008 International Conference on Information Processing in Sensor Networks (ipsn 2008), IEEE, 2008: 371 - 382.

- [11] LE T, DUAN Y. Pointgrid: A deep network for 3d shape understanding [C] //Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018: 9204 - 9214.
- [12] QI C R, SU H, MO K, et al. PointNet: Deep learning on point sets for 3d classification and segmentation [C] // Proceedings of the IEEE Conference on Computer Vision and Patternrecognition, 2017: 652 - 660.
- [13] 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 31 st International Conference on Neural Information Processing Systems, Long Beach, California, USA, New York: ACM, 2017: 5105 - 5114.
- [14] WANG Y, SUN Y, LIU Z, et al. Dynamic graph cnn for learning on point clouds [J]. ACM Transactions on Graphics (tog), 2019, 38 (5): 1 - 12.
- [15] XU B B, SHEN H W, CAO Q, et al. Graph convolutional networks using heatkernel for semi-supervised learning [C] //Proceedings of the 28th International Joint Conference on Artificial Intelligence. Macao, China, 2019: 1928 - 1934.
- [16] HU Q, YANG B, XIE L, et al. Randla-net: Efficient semantic segmentation of large-scalepoint clouds [C] // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 11108 - 11117.
- [17] CHEN J, KAKILLIOGLU B, REN H, et al. Why discard if you can recycle: A recycling max pooling module for 3dpoint cloud analysis [C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022: 559 - 567.
- [18] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need [C] // Proceedings of the 31st International Conference on Neural Information Processing Systems, New York: ACM, 2017: 6000 - 6010.
- [19] ZHAO H, JIANG L, JIA J, et al. Point transformer [C] //Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021: 16259 - 16268.
- [20] 梁振华, 王 丰. 面向部件分割的 PointNet 注意力加权特征聚合网络 [J]. 计算机应用研究, 2023, 40 (5): 1571 - 1576.
- [21] WEI M, WEI Z, ZHOU H, et al. AGConv: Adaptive graph convolution on 3D point clouds [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45 (8): 9374 - 9392.
- [22] CHEN C, LIU D, XU C, et al. SAKS: Sampling adaptive kernels from subspace for point cloud graph convolution [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2023, 33 (10): 6013 - 6025.
- [23] MA X, QIN C, YOU H X, et al. Rethinking network design and local geometry inpoint cloud: a simple residual MLP framework [C] //Proceedings of the Tenth International Conference on Learning Representations, USA: Ithaca, 2022: 11229 - 11244.
- [24] YU X, TANG L, RAO Y, et al. Point-bert: Pre-training 3d point cloud transformers with masked point modeling [C] //Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022: 19313 - 19322.
- [25] 陈 凯, 叶海良, 杨 冰, 等. 基于双支融合与结构采样的点云分类算法 [J]. 中国计量大学学报, 2024, 35 (2): 309 - 317.
- [26] 熊 伟, 娄政浩, 徐敏夫, 等. 集多头点注意力与边卷积的点云分类分割模型 [J]. 计算机辅助设计与图形学学报, 2025, 37 (3): 446 - 456.
- [12] YE M, ZHAO J, GUAN Q, et al. Research on eVTOL air route network planning based on improved A * algorithm [J]. Sustainability, Multidisciplinary Digital Publishing Institute, 2024, 16 (2): 561.
- [13] YANG and WEI P. Autonomous free flight operations in urban air mobility with computational guidance and collision avoidance [C] // in IEEE Transactions on Intelligent Transportation Systems, 2021, 9 (22), pp. 5962 - 5975.
- [14] 杨 磊, 陈凤翔, 陈科羽, 等. 基于多传感器的无人机避障方法研究及应用 [J]. 计算机测量与控制, 2019, 27 (1): 280 - 283.
- [15] 方祝平, 李 理, 唐 荣, 等. 融合改进 A * 和速度障碍法的无人机局部避障规划 [J]. 电光与控制, 2025, 2 (26): 1 - 8.
- [16] 郭 华, 郭小和. 改进速度障碍法的无人机局部路径规划算法 [J]. 航空学报, 2023, 44 (11): 271 - 281.
- [17] 李 樾, 韩 维, 陈清阳, 等. 基于改进的速度障碍法的有人/无人机协同系统三维实时避障方法 [J]. 西北工业大学学报, 2020, 38 (2): 309 - 318.
- [18] 马自勇, 朱星光, 马立东. 改进 A * 和 DWA 的机器人路径规划研究 [J]. 现代电子技术, 2024, 47 (20): 177 - 186.
- [19] 侯晨曦. 基于滚动速度障碍法的无人机山地航测避障路径规划研究 [J]. 计算机测量与控制, 2024, 32 (3): 232 - 238.
- [20] 高九州, 张 焯. 基于改进 A * 算法的无人机三维空间避障路径规划 [J]. 计算机测量与控制, 2023, 31 (12): 203 - 209.

(上接第 333 页)