

集改进图卷积和多层池化的点云分类模型

周锐闯, 田 瑾, 闫丰亭, 朱天晓
(上海工程技术大学 电子电气工程学院, 上海 201620)

摘要:针对基于图卷积的点云分类模型在提取点云不同语义区域的特征信息以及高效利用聚合的高维特征方面存在的问题,本文提出了一种新的点云分类模型,该模型采用了动态自适应图卷积和多层池化相结合的方法。具体而言,本文采用了残差结构来构建更深层的卷积,以学习不同语义区域点对特征中不同层次的特征信息,从而生成动态自适应调整卷积核,针对不同的点对动态更新边的特征关系,从而提取更为精确的局部特征。同时,本文将聚合的高维特征输入到多层最大池化模块中,回收利用第一次最大池化后丢弃的特征信息进行多层最大池化,从而获取更为丰富的高维特征,提高分类模型的精度。实验结果表明,在 ModelNet40 数据集上,本文提出的分类模型的总体精度达到 93.3%,平均精度为 90.7%,明显优于目前主流的点云分类模型,并具有较强的鲁棒性。

关键词:深度学习;图卷积神经网络;多层池化;点云分类

中图分类号:TP391.4;TN958.98 **文献标识码:**A **DOI:**10.3969/j.issn.1001-5078.2024.02.005

A point cloud classification model with improved graph convolution and multilayer pooling

ZHOU Rui-chuang, TIAN Jin, YAN Feng-ting, ZHU Tian-xiao

(School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China)

Abstract: Aiming at the problems of graph convolution-based point cloud classification models in extracting feature information from different semantic regions of the point cloud and efficiently utilizing aggregated high-dimensional features, a novel point cloud classification model is proposed, which combines dynamic adaptive graph convolution with multi-layer pooling. Specifically, residual structures is employed to construct deeper convolutions and learn feature information from different semantic regions of point pairs at different levels to generate dynamically adaptive adjusted convolution kernels that update the feature relationships of different point pairs, thus extracting more accurate local features. At the same time, the aggregated high-dimensional features are input into a multi-layer max pooling module to recover the discarded feature information from the first max pooling layer and obtain richer high-dimensional features to improve the accuracy of the classification model. The experimental results show that the proposed model achieves an overall accuracy of 93.3% and an average accuracy of 90.7% on the ModelNet40 dataset, which is significantly better than the current mainstream point cloud classification models, and has strong robustness.

Keywords: deep learning; graph convolutional neural networks; multi-layer pooling; point cloud classification

基金项目:国家基金委民航联合基金重点项目(No. U2033218)资助。

作者简介:周锐闯(1997-),男,研究生,主要研究方向为计算机图形学、深度学习。

通讯作者:田瑾(1982-),女,博士,副教授,硕导,主要研究方向为大规模数值计算、计算机电磁学、机器学习。E-mail: jintian0120@foxmail.com

收稿日期:2023-04-17

1 引言

目前,三维点云数据在多个实际应用领域中具有重要的应用价值,例如自动驾驶^[1]、虚拟现实^[2]以及机器人^[3-4]等领域。这是因为三维点云数据可以较好地保留三维模型的几何信息,而三维点云的分类则成为以上实际应用中的关键步骤之一。但是,三维点云数据具有无序性、稀疏性以及高维性等特点,因此采用卷积神经网络处理点云数据是当前备受关注的研究领域,也是一个具有高难度的挑战。

近年来,三维成像技术逐渐成熟,三维点云的数据量急剧增加,手工提取点云特征的方式变得难以实现。目前针对点云分类任务的神经网络方法主要分为三类:基于体素化的方法、基于多视图的方法和基于点的方法。基于体素化的方法是将无序的点云分成一系列占用一定空间的体素(Occupancy Voxels)的方法,再利用三维卷积进行体素级别的特征学习。第一个利用体素化方式进行点云分类任务的是 Maturana 等人^[5]提出的 VoxNet,通过三维卷积神经网络提取体素数据的空间局部特征信息,但使用三维卷积造成计算空间和内存大量消耗的问题。基于多视图的方法是将三维对象按多个角度投影成二维图像,MVCNN^[6]网络模型通过将点云中提取的视图投影到二维空间,并利用二维卷积神经网络进行特征提取。这种方法通过二维卷积降低了内存和计算资源的消耗,但多角度投影只能获取外部结构信息,从而造成大量三维点云数据的丢失。基于多视图或体素化的方法都是将三维点云数据转换为其他形式,导致有价值的信息丢失,影响分类任务的精度。

基于点的方法将三维点云数据直接作为模型的输入,可以充分利用点云数据的空间几何特征,降低数据预处理过程中的信息损失。Qi 等人^[7]提出了 PointNet 网络框架,首次将原始点云数据作为输入,该模型使用 T-Net 结构解决点云数据的旋转不变性问题,通过共享的多层感知机(Multilayer Perceptron, MLP)提取每个点的特征,最后使用最大池化进行信息聚合得到全局特征。但是,该方法没有考虑点与点之间的结构关系,因此对局部特征的提取不充分。Qi 等人在 PointNet 基础上进行了改善,提出了 PointNet++^[8]模型,通过层级下采样的方式增强局部几何信息的捕获能力,但是仍无法提取到点对

间的特征关系。梁振华等人^[9]在 PoinNet 网基础上提出注意力加权特征聚合模块,改善了最大池化损失次要特征的问题,对局部特征信息的捕获能力仍有不足。Te 等人^[10]提出了正则化图卷积神经网络 RGCNN,该网络模型利用了谱图理论,可以进一步提高了局部特征的获取能力。为了能够更好地获取点对的关系特征,Wang 等人^[11]提出了一种动态卷积神经网络(Dynamic Graph CNN, DGCNN),借鉴了图卷积神经网络结构,利用共享参数的 K 近邻算法(K Nearest Neighbors, KNN)和 MLP 从中心点及中心领域上提取局部特征,将不同层次的特征连接起来,计算出点对间的局部特征,但忽略了不同领域间的特征信息。王江安等人^[12]提出结合残差网络^[13]思想,通过搭建更深的图卷积层来进行更深层局部特征的学习。田晟等人^[14]提出结合注意力机制与图卷积神经网络的点云分类模型,利用注意力机制获取更具代表性的局部特征。梁志强等人^[15]提出使用图卷积和注意力机制分别提取点云低维和高维特征,通过多个特征融合的方式来提高点云分类精度。利用图卷积结构提取局部特征的方法已经成主流,然而,使用固定参数卷积核对不同语义区域点对的特征提取能力有限,此外,目前网络都采用一次最大池化聚合特征,这会丢弃大量由复杂网络结构学习到的特征集合。

针对以上问题,本文提出了一种基于动态自适应图卷积和多层池化的三维点云分类模型。该模型首先采用 KNN 算法将输入的点云组成局部有向图,然后利用动态自适应图卷积学习不同语义区域点的特征信息,生成多样的自适应卷积核,动态更新点对间边的权重,获取更精确的局部特征。最后采用多层最大池化聚合特征,利用第一次最大池化丢弃的特征重复最大池化操作,收集更丰富的高维特征,从而提高点云分类任务精度。

2 相关工作

2.1 图卷积神经网络

传统卷积神经网络在处理固定尺寸的二维图像分类任务中取得了很好的效果,但无法直接应用于无序数据的特征提取。Scarselli 等人^[16]提出图卷积神经网络,将图结构与卷积神经网络相结合,可以有效处理无序的数据。将图卷积神经网络应用于处理无序的点云数据,利用图结构学习点对间的依赖关

系,可以提取更丰富的局部结构特征。

利用图卷积神经网络处理点云分类任务是将 n 个点 F 维的点云数据直接输入,表示为 $X = \{x_1, x_2, \dots, x_n\} \subseteq R^F$, 当 $F=3$ 时,每个点的空间三维坐标表示为: $X_i = (x_i, y_i, z_i)$ 。局部特征提取过程如图 1 所示,首先利用 K 近邻算法构建局部有向图 $G = (V, E)$, 其中 $V = \{p_i | i = 1, 2, \dots, n\}$ 和 $E = \{e_i = (e_{i1}, \dots, e_{ik}) | i = 1, 2, \dots, n\}$ 表示图结构中的中心节点和边。

然后需要提取边特征,边特征的函数定义为 $e_{ij} = h_\theta(p_i, p_{ij})$, 其中 θ 为一组可学习参数, $h_\theta(\cdot)$ 表示学习边特征的非线性函数。边函数的选择对局部特征提取有重要的影响,公式(1)为 PointNet 模型边函数的定义,只提取了每个独立点的特征,没有考虑点对间的关联;公式(2)中只对局部信息进行编码,却没有考虑原始的全局特征信息;公式(3)是 DGCNN 网络模型中边函数的定义,该函数既考虑到每个中心点构成的全局信息,又考虑了点对间的局部关系,提高了获取的局部特征信息的精度:

$$e_{ij} = h_\theta(p_i) \quad (1)$$

$$e_{ij} = h_\theta(p_j - p_i) \quad (2)$$

$$e_{ij} = h_\theta(p_i, p_j - p_i) \quad (3)$$

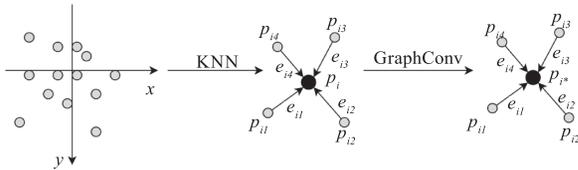


图 1 图卷积局部结构示意图

Fig. 1 Schematic diagram of the local structure of graph convolution

最后使用最大池化函数聚合每个图结构的特征信息,公式定义如(4)所示:

$$l_i = \max_{j: (i,j) \subseteq E} \text{ReLU}\{\text{Bn}(\text{GraphConv}(e_{ij}))\} \quad (4)$$

式中, Bn (Batch Normalization) 是数据归一化处理函数; ReLU (Rectified Linear Activation Function) 是非线性激活函数,经过最大池化函数后得到局部特征向量 l_i 。

2.2 残差结构

面对复杂且数量庞大的三维点云数据,构建更深层的网络结构,可以提取不同层次的信息。然而,普通网络结构中随着卷积层数的增加,会出现梯度

消失或梯度爆炸的问题。传统的解决方案是采用数据出初始化和标准化技术,但这些方法会导致网络性能的退化问题。使用残差结构搭建深层网络可以有效地解决这些问题。

残差网络是由多个残差元组成,每个残差块由两个卷积层、一个快捷连接和一个恒等映射构成,如图 2 所示。

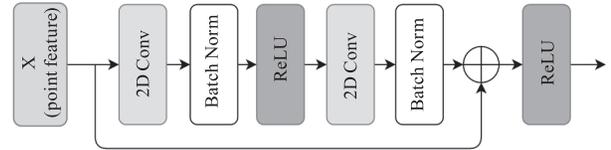


图 2 残差结构

Fig. 2 Residual structure

快捷连接通过跨越卷积层的直通路径将输入直接连接到卷积层的输出上,这种方式可以使网络从浅层的特征直接传递到深层,避免深度神经网络训练时梯度过度衰退的问题,从而更好地学习到特征。在快捷连接后,使用非常简单的恒等函数,将残差的输出直接传递到下一层的输入中,不会改变特征分布,这样能够保留输入信息,称为恒等映射,这样使得网络可以更深,同时避免网络性能的退化。通过快捷连接和恒等映射的组合,残差块可以更灵活的设计深层网络,提高网络的表达能力和性能。

3 算法描述

本文提了一种基于改进图卷积和多层池化的点云分类模型,整体网络结构如图 3 所示。该模型直接将无序的点云数据($N \times 3$)作为输入,其中 N 表示输入的点数,3 表示点云的三维坐标(x, y, z)。数据进入局部特征提取模块中的动态自适应调整卷积层 (Dynamics Adaptive Graph Conv, DAGraphConv) 后,首先利用 KNN 算法构建以 K 个节点为一组的有向图结构,然后提取图结构中每个点的特征,通过残差结构的深层卷积对点的特征进行学习,生成可适用不同语义区域点对的动态调整卷积核,并利用根据点的特征生成的卷积核处理图结构中对应对,获得更精确的特征向量,最后使用最大池化进行聚合,得到局部特征向量。本文模型的局部特征提取模块由 2 个 DAGraphConv 和 2 个 GraphConv 组成,局部特征的维度设置为 (64, 64, 128, 256)。局部特征提取模块提取到的特征会被拼接,并通过 MLP (1024)

对特征向量进行更深层的学习。随后,通过多层最大池化模块,利用前一次最大池化丢弃的部分再进行多层最大池化,更高效的利用特征信息且减少冗

余信息的产生,得到全局特征向量。最后,将得到的全局特征向量经过全连接层进行分类,获得最终的结果。

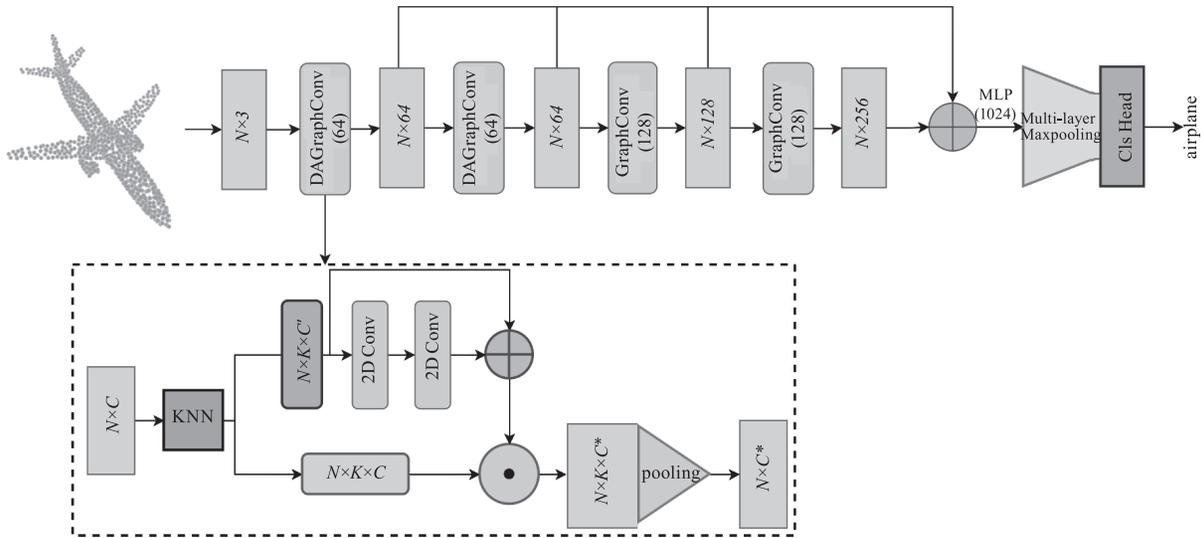


图 3 点云分类模型总体架构

Fig. 3 General architecture of the point cloud classification model

3.1 动态自适应图卷积

在图卷积神经网络中,固定参数的卷积层仅能提取点云数据中片面的几何信息,无法充分考虑组成图结构的不同语义区域点所携带的特征信息。因此,使用图卷积神经网络提取特征存在一定程度的局限性。为此,本文提出了一种动态自适应图卷积方法,该方法通过学习不同点对之间的独特关系,形成动态自适应调整卷积核,以便精确提取每一点对的特征信息。过程如图 3 所示,步骤描述如下:

首先通过 KNN 算法将无序的点云构成以 K 个点为集合的有向图结构 $G = (V, E)$ 。然后通过残差结构的卷积层学习图中点对特征关系,生成动态自适应调整卷积核,公式如下:

$$\begin{cases} f_{ij} = g(f_i, f_j - f_i) \\ f_{(y)} = \text{ReLU}\{\text{Bn}(\text{Conv}(f_{ij}))\} \\ DA_{\text{convk}} = (\text{Conv}(f_{(y)})) \oplus f_{ij} \end{cases} \quad (5)$$

式中, (f_i, f_j) 分别表示图中心点特征及邻居点特征; $g(\cdot)$ 表示特征映射函数,将第一层卷积学习到的特征信息结果归一化处理,再使用 ReLU 激活函数进行进一步优化,最终,通过将第二层卷积后的结果与初始传入的特征进行拼接,得到动态自适应调整卷积核 DA_{convk} 。如图 4 所示,通过动态自适应调整卷积核处理图中点对,更新每对独特点对间边的

权重,如式(6)所示:

$$e_{ij}^* = \text{ReLU}\{\text{Bn}(DA_{\text{convk}} \odot e_{ij})\} \quad (6)$$

式中, \odot 表示点乘,动态更新后的边特征 e_{ij}^* 可以更准确的表示不同语义部分点的特征关系。

最后通过最大池化函数聚合更新后每个图的特征信息,得到局部特征,如式(7):

$$l_i = \max_{j: (i,j) \subseteq E} DA_{\text{GraphConv}}\{e_{ij}^*\} \quad (7)$$

由于点的特征会随着卷积层深度变化,每次提取局部特征后重新使用 KNN 算法更新图结构,这样可以让信息在相似的结构之间更好的传播,加快局部语义信息的学习能力。

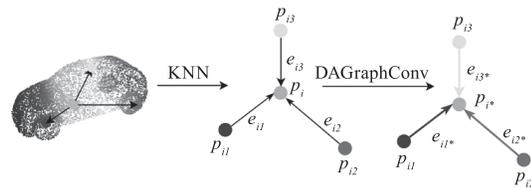


图 4 DA Graph Conv 局部结构示意图

Fig. 4 Schematic diagram of the local structure of DA Graph Conv

3.2 多层池化

目前主流点云分类网络模型采用一次最大池化聚合局部特征,以获得全局特征集合,然后,这种方式存在局部特征利用率低的问题,因为只有少部分局部局部特征被用于全局特征的生成。另一种解决

方式是将一次最大池化和平均池化得到的特征向量拼接,但这会产生冗余信息,影响分类精度。为了解决以上问题,本文提出了多层最大池化(Multi-layer Max Pooling, MMPooling)方法,该方法利用前一次最大池化丢弃的特征集合进行多层最大池化,可以更高效地利用经过复杂计算学习到的特征集合,同时降低了冗余信息的产生,提高点云分类模型的精度。具体步骤如下:

首先将每个卷积层提取的局部特征进行并联,再经过 MLP(1024) 将特征维度提升到 1024 维,得到一个整体的高维特征,公式定义如式(8):

$$f = MLP^{1024}[l_1 \oplus l_2 \oplus l_3 \oplus l_4] \quad (8)$$

式中, \oplus 为特征向量的拼接。然后将拼接的 $N \times M$ 高维特征矩阵作为输入,分别使用多层的最大池化,第一层最大池化提取 $1 \times M$ 维特征后,将剩下的 $(N-1) \times M$ 作为下一层最大池化的输入,得到的 F_1, F_2, \dots, F_n 表示每层最大池化聚合后的特征矩阵,每层聚合函数对应的分类损失函数定义如式(9):

$$L_i = \Phi(y_i, \phi(F_i)) \quad (9)$$

式中, y_i 表示真实标签值; $\phi(\cdot)$ 表示 soft-max 激活函数; $\Phi(\cdot)$ 表示交叉熵损失函数。整体的分类损失定义如式(10):

$$L_c = \sum_{i=1}^n \Phi(y_i, \phi(F_i)) \quad (10)$$

仅仅将每个池化层得到的损失串联或者相加并不能得到最优的实验结果,本文引入了修正损失函数,如图 5 所示。

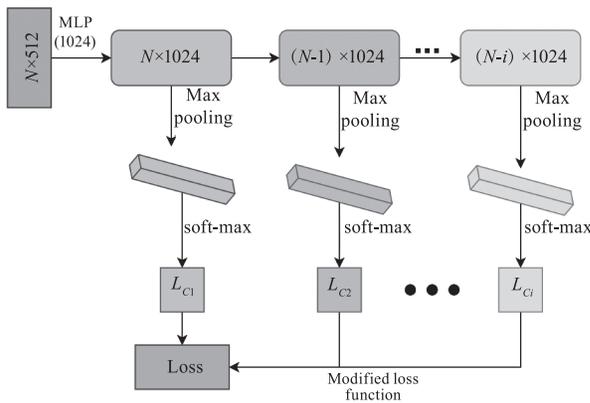


图 5 多层最大池化结构图

Fig. 5 Multi-level maximum pooling structure diagram

通过 F_2, F_3, \dots, F_n 的损失来修正 F_1 , 让 L_1 在其中保持是最小的值,维持网络的稳定,修正损失函

数定义如式(11):

$$L_{CR} = |1 - e^{L_i - (\beta_i \cdot \gamma \cdot \lambda_i)}| \quad (11)$$

$$\gamma = e^{(\sum_{i=1}^n \gamma_i \cdot \phi(F_i))} \quad (12)$$

公式(12)中 n 表示类别数量。在训练早期,收集到的特征并不是十分可靠,修正损失函数获得的收益不大,随着迭代次数增加,每一层的预测逐渐稳定,修正损失函数发挥作用,保证 $L_i > L_1$, 第一次池化得到的特征信息保持最佳利用。

最后联合修正损失函数以及分类损失函数,得到整体损失表达式:

$$Loss = \rho L_{CR} + (1 - \rho) L_c \quad (13)$$

其中 ρ 表示为可学习参数。

4 实验

4.1 实验环境和参数

本文算法的实验硬件环境为 Intel Core i5-12400F、16 GB 内存, GPU 为 NVIDIA GE-FORCE RTX 3060; 软件环境为 Linux Ubuntu 18.04 操作系统、CUDA 10.1、Pytorch 1.6、Python 3.7。

训练和测试的实验参数设置为:训练迭代次数为 250 次, Batch Size 设置为 32 或 16, Batch Size 的大小会影响显存的消耗以及模型训练的收敛速度, 本文实验中 Batch Size 会根据整体参数进行对应的调整。选择随机梯度下降法^[17]作为网络的优化器, 将动量设置为 0.9, 采用余弦退火法^[18]将学习率从 0.1 降低到 0.001。每个全连接层后都加入了 dropout 层, 比例设置为 0.5, 使用 ReLU 激活函数, 防止训练过程中出现过拟合。

经过训练的模型在测试集中进行测试得出最终的总体准确率以及平均准确率, 整体准确率 A_{oacc} (overall accuracy) 和平均准确率 A_{macc} (mean accuracy) 表示如下:

$$A_{oacc} = \frac{TN + TP}{TP + TN + FP + FN} \quad (14)$$

$$A_{macc} = \frac{\sum A_{oacc}}{TP + TN + FP + FN} \quad (15)$$

其中, TP 为真阳性, 表示实际为真, 预测也为真的样本数; FP 为假阳性, 表示实际为假但预测为真的样本数; TN 为真阴性, 表示实际为假预测也为假的样本数; FN 为假阴性, 表示实际为真但是却预测为假的样本数。

4.2 实验数据集

为了方便文中提出的点云分类模型实验数据与

其他主流模型作对比,本文选择在 ModelNet40 数据集^[18]上进行训练和测试。该数据集共有 40 个类别,含 12311 个 CAD 模型,其中有 9842 个物体用于训练,2469 个物体用于模型测试,该数据集中每个物体都有 2048 个点的三维坐标。其中部分实例展示在图 6 中,图中的飞机、钢琴和汽车等三维模型特征明显,在网络模型中较容易被区分。但长椅和椅子的模型相似度较高,需要能提取更精确特征信息的网络模型才能准确的区分。本文实验仅使用采样点的 (x, y, z) 坐标作为输入,采用随机旋转、缩放以及逐点抖动的方式进行数据增强。

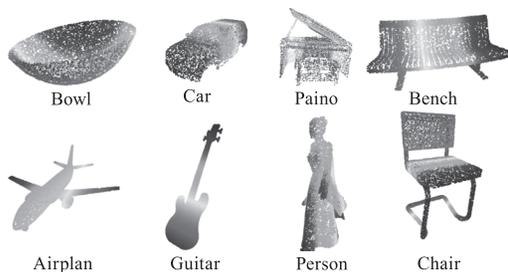


图 6 三维点云模型可视化

Fig. 6 3D point cloud model visualization

4.3 实验结果与分析

4.3.1 分类结果分析

为了验证本文提出算法的优势,在 ModelNet40 数据集上与目前主流的点云分类模型 DGCNN、PointNet++、PointCNN 等进行分类精度的对比,结果如表 1 所示。表中不同模型的输入的数据形式不同,其中 Voxels 表示使用体素作为输入,View 表示使用多视图作为输入,Points + Normal 表示使用三维坐标和法线向量作为输入,本文实验仅使用点的三维坐标信息作为输入。从实验结果可以看到,本文提出算法整体的分类精度达到 93.3%,平均分类精度为 90.4%,相比于其他点云分类模型,本文模型在分类精度上有明显提升。

对比使用多视图或体素作为输入的 MVCNN 和 VoxNet 模型,本文提出的方法直接将原始点作为输入,可以更高效的提取点云特征,减少了将数据转化为二维图像或体素形式造成的信息丢失,使整体准确率有 3.2%~8.8% 的提升。相比于同样使用点作为输入的 PointNet++ 和 PointNet 模型,本文模型不仅提取了每个单独点的特征信息,还考虑到了点与点之间的关联信息,加强了对局部特征信息的提取,在总体精度上有 1.6%~3.3% 提升。对比于

DGCNN 算法模型,本文模型考虑到不同语义部分点对差异性,使用固定参数卷积无法提取到准确的特征信息,本文中根据点对特征学习获得的自适应卷积进行特征提取,进一步增强了提取局部特征的能力,使整体精度有 0.7% 的提升。对比于增加了法向量信息作为输入的 SpiderCNN 以及融入了注意力机制的 Point Transformer 和 DTNet 模型,本文模型没有通过增加输入或采用更复杂且更耗费计算资源的特征提取方法来提高分类精度,而是采用多层最大池化更高效的利用特征信息,提高分类精度的同时不会额外增加模型复杂度,让整体精度有 0.4%~0.9% 的提升。通过对比实验表明,本文提出的模型与经典模型或当前主流的点云分类模型比较,都有着突出的表现,充分说明本文模型的实际可行性。

表 1 不同模型在 ModelNet40 数据集上的分类精度

Tab. 1 Classification accuracy of different models

on the ModelNet40 dataset

Method	Input	Aoacc/%	Amacc/%
PointNet	Point	90.0	84.3
PointNet++	Point	91.7	/
3D ShapeNets ^[19]	Voxels	84.5	77.5
MVCNN	View	90.1	/
VoxNet	Voxels	85.5	82.8
SpiderCNN ^[20]	Points + Normal	92.4	/
PointCNN ^[21]	Point	92.2	88.1
DGCNN	Point	92.6	89.8
Point Transformer ^[22]	Point	92.8	-
DTNet ^[23]	Point	92.9	90.4
Ours	Point	93.3	90.4

4.3.2 消融实验

为了验证本文提出的动态自适应图卷积及多层最大池化模块在三维点云分类任务中的有效性,构建了 4 组模型的自身消融对比实验进行验证,使用不同模块的研究结果如表 2 所示。

表中“√”表示该方法被采用,“×”表示没有采用该方法。模型 A 作为对比实验的基础模型,采用图卷积提取特征,仅使用一次最大池化聚合特征信息。模型 B 在模型 A 的基础上增加了本文提出的动态自适应图卷积进行特征提取。模型 C 在模型 A 的基础上增加了多层最大池化模块,增加卷积层提

取的局部特征的利用率。模型 D 在模型 A 的基础上同时增加了动态自适应图卷积和多层最大池化模块。通过对比实验可以得到,采用可以提取更精确局部特征的动态自适应图卷积模块后,模型的整体精度提高了 0.3%,平均精度提高了 0.8%。采用

多层最大池化后,将提取的特征更高效的利用,模型整体精度提高了 0.4%。同时加入动态自适应图卷积及多层最大池化模型,模型的整体精度提高了 0.9%,平均精度提高了 2.1%。实验结果表明,本文提出的 2 个模块均可明显提升模型分类精度。

表 2 不同模块的消融实验

Tab.2 Ablation studies about different modules

Method	GraphConv	DAGraphConv	MMPooling	Aoacc/%	Amacc/%
A	√	×	×	92.4	88.6
B	√	√	×	92.7	89.4
C	√	×	√	92.8	89.8
D	√	√	√	93.3	90.7

4.3.3 池化层数测试

为了研究多层最大池化(MMPooling)模块中使用不同层数对模型分类精度的影响,本文在 ModelNet40 数据集上分别使用(1、2、3、4)层最大池化,实验结果如表 3 所示。

表 3 最大池化层数测试

Tab.3 Maximum pooling layer test

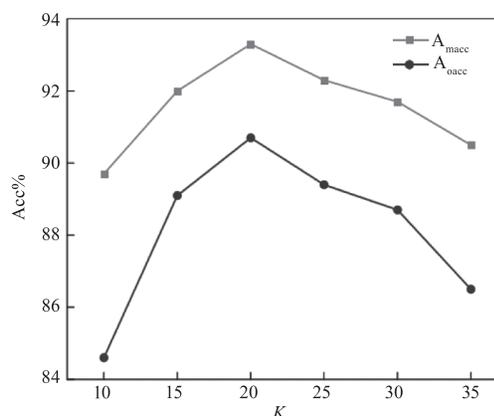
Layer	MMPooling	Aoacc/%	Amacc/%
1	×	92.6	88.7
2	√	92.8	89.5
3	√	93.3	90.7
4	√	92.6	89.0

表中层数为 1 表示未使用多层最大池化模块,仅用一次最大池化聚合特征,仅保留一组特征信息,无法充分利用经过复杂计算得到的局部结构信息。当使用最大池化层数增到 2 层,可以用于分类的特征信息增加,模型的整体精度提高了 0.2%。当采用 3 层最大池化聚合特征,进一步提高由复杂卷积计算得到的特征信息的利用率,对比仅使用一层最大池化,整体精度提高了 0.7%。但当使用的最大池化层数继续增加,会造成聚合过多冗余信息,从而干扰最后的分类结果,无法提高模型分类精度。实验结果表明,使用最大池化的层数不同会对实验结果产生影响,合适的层数可以显著提高模型性能,实验中三层效果最佳,因此本文使用三层最大池化。

4.3.4 超参数实验

本文使用 KNN 算法构建有向图结构,其中的超参数 K 值决定有向图中节点个数,为了研究 K 大小

对点云分类模型准确率的影响,本文设置了在同样参数环境下,选择不同 K 值(K 为 10、15、20、25、30、35)在 ModelNet40 数据集上进行测试,分类精度如图 7 所示。由图中的数据可以发现,当 K 值较小时,构建的有向图中节点个数少,导致无法充分提取中心点与其相邻节点的特征。但当 K 值过大时,邻居节点的数量过多会产生大量冗余信息,局部领域的几何结构被破坏,导致点云模型分类精度下降,同时还会增加网络模型的训练时间。当 $K=20$ 时,本文模型的性能达到最优,因此一个合适的 K 值对模型分类精度有至关重要的影响。

图 7 K 值对分类精度的影响Fig.7 Influence of K value on accuracy

4.3.5 模型鲁棒性实验

为了验证本文网络模型对稀疏输入和遮挡问题的鲁棒性,在 ModelNet40 数据集上采用随机依次减少 25% 的采样点(1024、768、512、256)。稀疏三维点云模型可视化如图 8 所示,图 8(a)不减少输入点数,物体的结构特征可以充分地展现出来,图中 8

(b)、(c)、(d)依次减少输入点数,物体的结构逐渐模糊,因此需要网络模型有更强提取特征信息的能力,才能在减少采样点情况下可以准确分类。

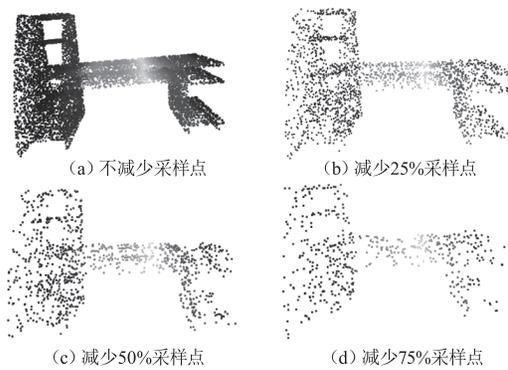


图 8 稀疏点云可视化

Fig. 8 Visualizing Sparse Point Clouds

实验中使用表 2 中模型 D 与 DGCNN 等模型进行对比实验,结果如图 9 所示。从图中可以看出,与同样使用图卷积的 DGCNN 模型比较,随着采样点数减少,本文模型的整体精度从 93.3 % 下降到 91.5 %,下降了 1.8 %,下降趋势较平缓,DGCNN 模型从 92.6 % 下降到 47.1 %,下降趋势较明显。对比针对单个点提取特征 PointNet 和 PointNet++ 网络模型,减少采样点数对获取局部特征影响较小,PointNet++ 模型整体精度从 91.7 % 下降到 89.9 %,PointNet 模型整体精度从 90.0 % 下降到 85.8 %,本文模型整体精度始终高于 PointNet 和 PointNet++ 模型,且随着采样点减少,分类精度差距逐渐增加。这说明在面对遮挡和稀疏点输入问题时,能够识别不同语义点信息和高效利用提取的局部特征可以使网络具有较强的鲁棒性。

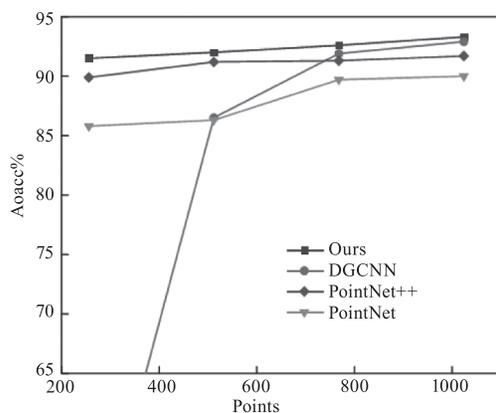


图 9 采样点密度对分类精度的影响

Fig. 9 The influence of sampling point density on classification accuracy

5 结论

为了提高点云分类模型对不同语义区域点对特

征的提取能力,本文提出了一种集改进图卷积和多层池化的点云分类模型。该模型通过学习不同点的特征,生成可动态调整点对关系的卷积核,以处理图结构中不同语义区域的点对,捕获更精确的局部特征信息。此外,该模型使用多层最大池化来聚合高维特征以得到全局特征,从而更高效地利用局部特征,同时减少冗余信息的产生,提高模型的分类准确度。在公共数据集 ModelNet40 上的实验结果表明,相较于目前经典的点云分类模型,本文模型的分类精度提高了 0.7 % ~ 3.3 %。消融实验、超参数实验以及逐渐减少采样点数实验进一步验证了本文对图卷积和池化层的改进可以提高点云分类任务的精度,并增加模型的鲁棒性。然而,更有效的卷积层和池化层会使网络模型更加复杂,模型的参数数量也会随之增加。因此,未来的研究将注重于在不降低分类精度的前提下,实现模型更轻量化的目标。

参考文献:

- [1] Li You, Ibanez-Guzman J. Lidar for autonomous driving: the principles, challenges, and trends for automotive lidar and perception systems [J]. IEEE Signal Processing Magazine, 2020, 37(4): 50-61.
- [2] Xiong Jianghao, Hsiang E L, He Ziqian, et al. Augmented reality and virtual reality displays: emerging technologies and future perspectives [J]. Light: Science and Applications, 2021, 10(1): 216.
- [3] Wang Jingyi, Chen Jialuo, Sun Youcheng, et al. RobOT: robustness-oriented testing for deep learning systems [C]// IEEE/ACM the 43rd International Conference on Software Engineering (ICSE). Madrid, ES: IEEE, 2021: 300-311.
- [4] O'shea K, Nash R. An introduction to convolutional neural networks [J]. arXiv preprint arXiv:08458, 2015.
- [5] Maturana D, Scherer S. Voxnet: a 3D convolutional neural network for real-time object recognition [C]// IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Hamburg, Germany: IEEE, 2015: 922-928.
- [6] Su Hang, 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 (ICCV). 2015: 945-953.
- [7] Qi C R, Su Hao, Mo Kaichun, et al. Pointnet: deep learning on point sets for 3D classification and segmentation [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017: 652-660.

- [8] Qi C R, Yi Li, Su Hao, et al. PointNet++ : deep hierarchical feature learning on point sets in a metric space[J]. Advances in Neural Information Processing Systems, 2017, 30:1 – 10.
- [9] Liang Zhenhua, Wang Feng. PointNet attention-weighted feature aggregation network for part segmentation[J/OL]. Computer Application Research, 2023, 40 (5) : 15771 – 1576. (in Chinese)
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/OL]. 计算机应用研究, 2023, 40 (5) : 1571 – 1576.
- [10] Te G, Hu Wei, Zheng Amin, et al. RGCNN: regularized graph CNN for point cloud segmentation[C]//Proceedings of the 26th ACM International Conference on Multimedia, 2018:746 – 754.
- [11] Wang Y, Sun Y, Liu Z, et al. Dynamic graph CNN for learning on point clouds [J]. Acm Transactions On Graphics, 2019, 38(5) :1 – 12.
- [12] Wang Jiangan, He Jiao, Pang Dawei. Point cloud classification and segmentation network based on dynamic graph convolutional network[J]. Advances in Laser and Optoelectronics, 2021, 58(12) :470 – 477. (in Chinese)
王江安, 何娇, 庞大为. 基于动态图卷积网络的点云分类和分割网络[J]. 激光与光电子学进展, 2021, 58(12) :470 – 477.
- [13] Zagoruyko S, Komodakis N. Wide residual networks[J]. arXiv Preprint arXiv:07146, 2016.
- [14] Tian Sheng, Long Anyang. A point cloud classification method based on graph convolution and multilayer feature fusion[J]. Advances in Laser and Optoelectronics, 2023, 60(4) :281 – 288. (in Chinese)
田晟, 龙安洋. 基于图卷积和多层特征融合的点云分类方法[J]. 激光与光电子学进展, 2023, 60(4) :281 – 288.
- [15] Liang Zhiqiang, Chen Chunmei, Chen Yanjie, et al. Point cloud classification and segmentation network based on fusion of high and low dimensional dual local features[J]. Laser & Infrared, 2022, 52(10) :1557 – 1564. (in Chinese)
梁志强, 陈春梅, 陈妍洁, 等. 高低维双局部特征融合的点云分类分割网络[J]. 激光与红外, 2022, 52(10) : 1557 – 1564.
- [16] Scarselli F, Gori M, Tsol A C, et al. The graph neural network model[J]. IEEE Transactions on Neural Networks, 2008, 20(1) :61 – 80.
- [17] Bottou L. Stochastic gradient descent tricks[M]. Neural Networks:Tricks of the Trade. Springer, 2012:421 – 436.
- [18] Loshchilov I, Hutter F. Sgdr: stochastic gradient descent with warm restarts [J]. arXiv Preprint arXiv: 1608. 03983, 2016.
- [19] Wu Zhirong, Song Shuran, Khosla A, et al. 3D ShapeNets: a deep representation for volumetric shapes zhirong[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition(CVPR), 2015:1912 – 1920.
- [20] Xu Yifan, Fan Tianqi, Xu Mingye, et al. SpiderCNN: deep learning on point sets with parameterized convolutional filters[C]//Proceedings of the European Conference on Computer Vision(ECCV), 2018:87 – 102.
- [21] Li Yangyan, Bu Rui, Sun Mingchao, et al. PointCNN: convolution on X-transformed points[C]//32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montreal, Canada, 2018.
- [22] Zhao Hengshuang, Jiang Li, Jia Jiaya, et al. Point transformer [C]//Proceedings of the IEEE/CVF International Conference on Computer Vision(ICCV), 2021:16259 – 16268.
- [23] Han X F, Jin Y F, Cheng H X, et al. Dual transformer for point cloud analysis[J]. IEEE Transactions on Multimedia, 2022, 25 :5638 – 5648.